



**HAL**  
open science

# Understanding the automatic text recognition process : model training, ground truth and prediction errors

Floriane Chiffolleau

► **To cite this version:**

Floriane Chiffolleau. Understanding the automatic text recognition process : model training, ground truth and prediction errors. Linguistics. Le Mans Université, 2024. English. NNT : 2024LEMA3002 . tel-04886481v2

**HAL Id: tel-04886481**

**<https://theses.hal.science/tel-04886481v2>**

Submitted on 30 Jan 2025

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



**Le Mans  
Université**

# THÈSE DE DOCTORAT DE

LE MANS UNIVERSITÉ

ÉCOLE DOCTORALE N° 643

*Arts, Lettres, Langues*

Spécialité : « *Langue, littératures françaises, littératures francophones* »

LABEL « *Thèse Humanités Numériques* »

Par

**Floriane CHIFFOLEAU**

**Understanding the Automatic Text Recognition Process : Model  
Training, Ground Truth and Prediction Errors**

Thèse présentée et soutenue à Paris, le 20 novembre 2024

Unité de recherche : 3L.AM/ALMAnaCH

Thèse N° : 2024LEMA3002

## Rapporteurs avant soutenance :

Ioana GALLERON Professeure à l'Université Sorbonne Nouvelle

Antoine DOUCET Professeur à La Rochelle Université

## Composition du Jury :

Président : Antoine DOUCET Professeur à La Rochelle Université

Examineurs : Elena PIERAZZO Professeure à l'Université de Tours

Jean-Philippe MAGUÉ Maître de conférences à l'ENS Lyon

Dir. de thèse : Anne BAILLOT Professeure à Le Mans Université

Co-dir. de thèse : Laurent ROMARY Directeur de recherche à l'Inria



# ACKNOWLEDGEMENT

---

Here, I would like to thank many persons and institutions, without whom I would never have been able to finish this thesis.

First, I want to express my most sincere gratitude to my supervisors, Anne Baillot and Laurent Romary. Thank you for your trust in me, first by hiring me to work on the DAHN project, then by choosing to bring me on to the projects EHRI and HarmonizingATR, and lastly, by offering me to do this PhD. I would not have done it without your quick email answers, the online meetings, the constant help, whether it was for my experiments or the redaction of my thesis. Mostly, thank you for your reinsurance, time, and time again, during this PhD, when I felt like I hadn't done anything of importance. To conclude, I want to say that I am very grateful for the many beautiful places abroad that I was able to travel to (England, Germany, Poland, Austria, Czech Republic) and for the many interesting people that I met, thanks to my involvement in the projects previously mentioned. Thank you also to the members of my *Comité Individuel de Suivi (CSI)*, Elena Pierazzo and Jean-Philippe Magué, for the meetings that we had, and the ideas and encouragements you gave me to help me continue serenely my PhD.

I want to share my appreciation for the institutions that supported me during this PhD.

Thank you to Le Mans Université, for the varied trainings I attended the (very) few times I was in Le Mans, and notably MT180, which was a very fun and formative experience. I would like especially to thank the managers of the doctoral school, that had to deal with my numerous emails and all the administrative complications of being hundreds of kilometres away.

Many thanks also go to Inria Paris, for having warmly accommodated me for four years as an engineer, then as a PhD candidate, and for the many fun events organised during my time there, such as the delicious food for the European week, the exhibition, and performance of robots, or the AGOS parties and games.

My gratitude extends especially to the ALMAnaCH team, past and present, for the great work environment, the various chats during lunch and the many interesting topics discovered and discussed.

Thank you in particular to Alix Chagué, Lydia Nishimwe, and Arij Riabi, my “PhD companions” that started at the same time as I did, and to whom I often went to, to compare experiences and advancements.

I am grateful, as well, for my office colleagues, past and present, from the former office C121 or, as we liked to call it, Clippy 121 (RIP little paper clip on our white board), for being so nice and lovely, and particularly Lydia and Hugo, who were there with me for the whole three years of the PhD.

I would like to acknowledge my DH colleagues, Alix Chagué, Thibault Clérice, Juliette Janès, and Hugo Scheithauer, for our countless exchanges on ATR, eScriptorium, TEI-XML, Oxygen or TEI Publisher, and for the great projects and papers we worked on. Additionally, I would also like to address a special thanks to Sarah Bénérière and Samuel Scalbert, my interns, turned colleagues, for giving me a great experience as a tutor, for relieving me of part of the work I had to do on the projects I was involved in during my thesis, and for our ongoing collaboration after their internship.

Most importantly, I want to express gratitude to my family, for their support during those three years, and specifically my parents and my sister.

Thank you for the multiple conversations, when I was doubting and stressing on whether I would succeed, and for the moments of rest that you offered me many times, whether it was at the beach, under the sun of Hendaye, or as a nap cushion for the adorable Griotte and Themis.

# SUMMARY

---

Acknowledgement	3
List of Acronyms	9
<b>Introduction</b>	<b>11</b>
<b>I Automatic Text Recognition: An Evolving Technique in Need of Understanding</b>	<b>21</b>
1 From OCR to ATR: Evolution of Transcription Techniques	23
2 A New Horizon Opening with the Emergence of Deep Learning Approaches	33
<b>II The (Null) Influence of the Lexicon</b>	<b>49</b>
3 The Correspondence of Paul d'Estournelles de Constant: A Thematic Corpus to Lead Our Study	51
4 Knowing the Content Before Judging its Effect	61
5 Estimating the Effect of Lexicon-Based Generated Models	77
<b>III Exploring the Impact of the Infralexical Level: The N-Gram</b>	<b>95</b>
6 Studying the N-Gram and Its Distribution Within the Dataset	97
7 Evaluating the Impact of the N-Gram Using Prediction Errors	115

<b>IV Multilingualism: An Answer to the N-Grams</b>	<b>139</b>
8 A New Dataset: Multilingual Documents from the Holocaust	141
9 Analysing How the Multilingual Model Cooperates	151
10 Testing the Efficacy of the Model by Observing Its Errors	165
<b>Conclusion</b>	<b>195</b>
<b>Annex I</b>	
<b>Tables</b>	<b>203</b>
A Tables of Results (Content Analysis)	204
B Metrics (Comparative Analysis)	206
C Tables of Results (Token Analysis)	212
D Comparison Transcription (Token Error Analysis)	219
E Tables of N-Grams (Token Error Analysis)	235
F Alphabet Analysis (Multilingual Model)	269
G Metrics (Multilingual Model)	278
H Comparison Transcription (Multilingual Model)	281
I Tables of N-Grams (Multilingual Model)	289
<b>Annex II</b>	
<b>Figures</b>	<b>311</b>
J Word Clouds (Lexicon Analysis)	312
K Word Clouds (Content Analysis)	315

---

L Diagrams (Content Analysis)	317
M Bar Charts (Token Analysis)	321
<b>Annex III</b>	
<b>Datasets</b>	<b>327</b>
N Paul d'Estournelles de Constant' Correspondence	328
O European Holocaust Research Infrastructure Editions	331
<b>Annex IV</b>	
<b>Resources</b>	<b>333</b>
P Python Scripts	334
Q Glossary	337
R Web Resources	340
<b>Annex V</b>	
<b>PhD activities</b>	<b>343</b>
S Chronology	344
T Conferences	346
U Training	350
<b>Bibliography</b>	<b>355</b>
List of Figures	373
List of Tables	377
Index	379





# LIST OF ACRONYMS

---

- ALMAnaCH** Automatic Language Modelling and Analysis & Computational Humanities. 11, 39, 42, 82
- ALTO** Analysed Layout and Text Object. 36, 38
- ASR** Automatic Speech Recognition. 27, 80, 83, 89
- ATR** Automatic Text Recognition. 15, 16, 18, 20, 23, 30, 31, 33, 34, 37–40, 42, 44, 45, 48, 54, 77, 78, 80–82, 99, 146, 176, 201
- CER** Character Error Rate. 78–81, 85–88, 91–93, 173, 174, 369
- CLI** Command-Line Interface. 34, 81, 84, 149
- CREMMA** Consortium pour la Reconnaissance d’Écriture Manuscrite des Matériaux Anciens. 28, 38, 39
- DAHN** Dispositif de soutien à l’Archivistique et aux Humanités Numériques. 11, 17, 46, 53, 198
- DH** Digital Humanities. 11, 12, 23, 33, 34, 41, 42, 45, 53
- EHRI** European Holocaust Research Infrastructure. 11, 17, 19, 41–43, 46, 143, 144, 146, 147, 149, 158–161, 165, 167, 168, 173, 197, 198, 341, 369, 375, 376
- FAIR** Findable, Accessible, Interoperable, Reusable. 13, 14
- GPU** Graphics Processing Unit. 149
- GUI** Graphical User Interface. 34, 35, 85
- HTR** Handwritten Text Recognition. 18, 19, 23, 27, 28, 36, 39, 40, 42, 44, 82, 98, 199
- IIIF** International Image Interoperability Framework. 14, 15, 38

**Inria** Institut National de Recherche en Informatique et Automatique. 11, 15, 17

**KaMI** Kraken as Model Inspector. 82–85, 89, 171

**LLM** Large Language Model. 36, 41

**MEHRI** model EHRI. 181

**MER** Match Error Recognition. 80, 83

**MGT** model Ground Truth. 112, 113, 116, 117, 120–122, 125–130, 132, 133, 135–137, 181, 192, 370

**MO** model Other. 84–86, 91–93, 112, 116, 117, 120–122, 125–131, 133, 134, 136, 369, 370, 375

**MW** model War. 84, 85, 87, 91–93, 112, 116, 117, 120–122, 125–132, 134–136, 370, 375

**MWR** model War Retrained. 84, 85, 88, 91–93, 375

**NLP** Natural Language Processing. 11, 70, 97

**OCR** Optical Character Recognition. 18, 19, 23–26, 28, 33–35, 39, 42, 47, 81, 82, 98, 199

**PAGE** Page Analysis and Ground truth Elements. 36, 38

**SER** Sentence Error Recognition. 80

**SGT** set Ground Truth. 63–67, 99, 101, 104, 107, 108, 110, 112, 369

**SO** set Other. 61, 63–65, 67–72, 74–76, 84–89, 91, 92, 99, 101, 104, 105, 107, 108, 110, 112, 116, 369, 375

**SW** set War. 63–72, 74–76, 84–89, 91, 92, 99, 101, 104–108, 110, 112, 116, 369, 375

**TEI** Text Encoding Initiative. 14, 18, 36

**VGSL** Variable-size Graph Specification Language. 31, 32, 37

**Wacc** Word Accuracy. 78, 79, 85–88, 91, 92, 173, 175, 369

**WER** Word Error Rate. 78–80, 85–88, 92, 93, 173–175, 369

# INTRODUCTION

---

In 2020, after graduating from a master’s degree in digital humanities,<sup>1</sup> I started working as an engineer at Inria Paris, in the ALMAnaCH<sup>2</sup> project-team, dedicated to Natural Language Processing (NLP) and Digital Humanities (DH). My work focused on the creation of digital scholarly editions, allowing me to learn about various tools to digitally exploit a wide diversity of documents, such as eScriptorium<sup>3</sup>, Nakala,<sup>4</sup> or TEI Publisher<sup>5</sup>. I had the opportunity to practice these skills on tangible documents. I worked on several collections of documents, from different time periods, languages, and topics. First, I worked on the DAHN project<sup>6</sup>, on two types of documents: a French, typewritten correspondence, between a French and an American diplomat, written during and about World War I, and various handwritten correspondences, in French or in German, exchanged between many Berlin intellectuals, at the crossroads between the 18th and 19th centuries. Afterwards, I was also involved on the EHRI project<sup>7</sup>. It gathered together typewritten documents, in German, French, English, Yiddish, among others, relating events from the Holocaust. I learned the functioning and extent of many tools, to digitize, transcribe, encode and publish digital scholarly editions. I also came across the limits currently present in some steps. Digital scholarly editing had been the central part of my work for over a year, which prompted an aspiration for improvement. This improvement would facilitate my work and benefit future researchers by addressing some of the challenges I faced. Consequently, my PhD research has been oriented towards easing those parts.

---

1. Master Technologies Numériques Appliquées à l’Histoire (TNAH) from the École nationale des chartes: <https://www.chartes.psl.eu/formations/masters/master-technologies-numeriques-appliquees-lhistoire>

2. Automatic Language Modelling and Analysis and Computational Humanities: <https://almanach.inria.fr/index-en.html>

3. <https://escriptorium.inria.fr/>

4. <https://nakala.fr>

5. <https://teipublisher.com/>

6. *Dispositif de soutien à l’Archivistique et aux Humanités Numériques* (Support system for Archival and Digital Humanities) : <https://github.com/FloChiff/DAHNPProject>

7. <https://www.ehri-project.eu/>

## A Workflow for Creating Digital Scholarly Editions

### What is a Digital Scholarly Edition?

Before going into further details, I need to clearly define the concept of digital scholarly edition, as it is the core of my work as a digital humanist. According to Patrick Sahle, a digital scholarly edition is

"the critical representation of historic documents [...] that are guided by a digital paradigm in their theory, method, and practice." [Driscoll and Pierazzo 2016, pp. 23–28]

The critical representation “aims at the reconstruction and reproduction of texts and as such addresses their material and visual dimension as well as their abstract and intentional dimension” [Driscoll and Pierazzo 2016, p. 25]. In this case, the material and visual dimensions refer to the physical characteristics of the documents, while the abstract and intentional dimensions focus on the content and purpose of the text. Historic documents refer to material documents (but not always to textual content) that “bridge a distance in time, a historical difference” [Driscoll and Pierazzo 2016, p. 26]. The adjective “digital” implies that the edition “cannot be given in print without significant loss of content and functionality” [Driscoll and Pierazzo 2016, p. 27]. These theoretical aspects are exemplified by numerous recent projects that showcase the diversity of digital scholarly editions in practice, be it in their structure, writing style, subject, or period of writing. Some examples of this variety include collections of manuscripts and printed documents from Gallica [Sagot et al. 2022], or parliamentary debates [Bourgeois et al. 2022], World War I correspondence [Chiffolleau and Baillet 2022], or mazarinades<sup>8</sup> [Abiven et al. 2022].

### Open Science and FAIR Principles: The Heart of the Workflow

The creation of digital scholarly editions not only involves a reflection on the content (topics, languages, etc.) and format (data structure, writing methods, etc.) of the texts that could and would be processed, but also on the workflow involved in their creation. One of the main concerns is the availability of the data produced and their capacities to be shared. Indeed, it is essential that the workflow and its components are easily reusable. This idea is at the basis of open science, towards which part of the DH research field has been oriented, since the start of the 2010s. The objective is to switch from

---

8. Pieces of satirical or burlesque verse published during the Fronde, about Cardinal Mazarin

the “publish or perish” mentality to a “knowledge sharing” perspective. According to European Commissioner for Research, Innovation, and Science Carlos Moedas,

"Open Science describes the ongoing transitions in the way research is performed, researchers collaborate, knowledge is shared, and science is organised [...]. [It] is also about making sure that science serves innovation and growth. It guarantees open access to publicly-funded research results and the possibility of knowledge sharing by providing infrastructures. Facilitating access to those data will encourage re-use of research output." [Giglia 2019, pp. 146–147]

Countries adopt open science principles in various ways. For example, in France, the open science initiative resulted in the creation of the Committee for Open Science, which was created by the French higher education and research community.<sup>9</sup> At the European level, it is embodied by the successive EU funding programs for research and innovation: Horizon 2020<sup>10</sup> (2014-2020) and Horizon Europe<sup>11</sup> (2021-2027). The mission of open science is to promote open access to all types of contents, through the creation of open source software for example, i.e. with their code and documentation accessible. Open science also goes against the concept of black box systems, that conceal their internal working, and only presents the inputs and outputs. To facilitate this knowledge sharing and in respect to the principles of open science, researchers have adopted data management frameworks like the FAIR principles (with the goal of making data Findable, Accessible, Interoperable, Reusable)<sup>12</sup>, that aim at guiding data producers and publishers to create easily reusable open data and tools [Wilkinson et al. 2016]. Therefore, researchers now engaged in the creation of workflows for digital scholarly editions are urged to comply with the FAIR principles in their projects [Galleron and Idmhand 2021] by using and promoting open source tools and standards, and by making their workflows public. The workflows are therefore useful for their creators, but also for the community as a whole. It means complete access and knowledge for the former, and universally implemented elements, with a set of rules and an extensive documentation, for the latter.

---

9. <https://www.ouvrirelscience.fr/the-committee-for-open-science/>

10. [https://research-and-innovation.ec.europa.eu/funding/funding-opportunities/funding-programmes-and-open-calls/horizon-2020\\_en](https://research-and-innovation.ec.europa.eu/funding/funding-opportunities/funding-programmes-and-open-calls/horizon-2020_en)

11. [https://research-and-innovation.ec.europa.eu/funding/funding-opportunities/funding-programmes-and-open-calls/horizon-europe\\_en](https://research-and-innovation.ec.europa.eu/funding/funding-opportunities/funding-programmes-and-open-calls/horizon-europe_en)

12. <https://www.go-fair.org/fair-principles/>

## Components of a Workflow

There are five major steps to follow in order to smoothly manage the development of the edition: digitization, segmentation, transcription, encoding, and publication. Upon the reflections inherent to the creation of a workflow, carefully considering those steps is key to properly building a digital scholarly edition. They have to be done in order, as each step's output is required for the next one. Although sequential, these steps are independent and can be outsourced or skipped if outputs are already available.

Some steps can also themselves be divided. Transcription can be a combination of text recognition and post-process correction, both of which can be done either manually or automatically.<sup>13</sup> Encoding can involve text structure tagging, named entities recognition, and critical annotation.

For each step, several open source, accessible, and reusable tools exist. Following the FAIR principles means making conscious choices regarding tools and standards that are developed and sustained thanks to the scientific community [Wilkinson et al. 2016]. For certain steps, some standards are recognized and widespread. The tool used for the digitization of a document (camera, scanner, phone) does not matter. However, for an online dissemination of a side-by-side image and its text in a digital scholarly edition, adhering to the International Image Interoperability Framework or IIIF<sup>14</sup> is the better option to abide by the FAIR principles and obtain a sustainable image repository. The framework creates a hub for the images, as they are published only once but can be reused many times, with metadata also attached to it, and an option to add and share these images in high definition, ensuring no loss of information. The framework also enables annotating the images as much as wanted.<sup>15</sup> For the encoding of text, the Text Encoding Initiative or TEI,<sup>16</sup> is widely considered as the standard for the representation of texts in digital form [Burnard 2019], through the use of the TEI Guidelines.<sup>17</sup> They recommend tags, according to the type of text worked on, its layout or specificities.

For the other steps,<sup>18</sup> no standard has been created yet, but common practices were agreed upon over time. For segmentation and transcription, semi-automation is the favoured method nowadays, i.e., manually segmenting/transcribing documents, training models

---

13. Post-OCR automatic correction is generally not recommended, as it is usually more troublesome than time-saving. See <https://harmoniseatr.hypotheses.org/226>

14. <https://iiif.io/>

15. <https://iiif.io/get-started/why-iiif/>

16. <https://tei-c.org/>

17. <https://tei-c.org/release/doc/tei-p5-doc/en/html/index.html>

18. Segmentation, Transcription, and Publication

from those data, applying them and correcting the remaining errors. The emergence of several Automatic Text Recognition (ATR) software [Jain, K. Taneja, and H. Taneja 2021] and the pooling of ground truth [Chagué, Clérice, and Romary 2021] have made automation more accessible. Regarding publication, the community has conducted a reflection on the practices these past years [Turska, Cummings, and Rahtz 2016; Pierazzo 2019]. Currently, in order to ease engineers' work, facilitate updates, and smooth the publication process, the recommended practice is not to create a website from scratch any more, to display the digital scholarly editions, but rather to use tools designed specifically to provide frameworks and interfaces for displaying documents, like TEI Publisher, Omeka,<sup>19</sup> or Edition Visual Technology (EVT).<sup>20</sup> Having standards and recommended practices help the overall work and implementing it in a workflow provides many advantages.

## The Importance of Implementing a Workflow

Workflows, as in pipelines of interoperable steps, are made to help researchers by easing their work and saving them time. Each choice in the workflow is driven by the goal of simplifying research tasks, thus choosing standards means prioritizing interoperability. The workflow intends to make the whole process run smoothly. In addition to prioritizing standards, workflows also offer flexibility, and options or elements that cover a wide range of data types can be proposed, making the workflow adaptable for a wide variety of research projects. Lastly, the semi-automation of most steps provides help for researchers, by enhancing efficiency without replacing human expertise.

## Practical Applications of a Digital Scholarly Editions Workflow

As an engineer at Inria, I worked on most of these steps with ease.

Creating IIIF links for my digitized documents was done pretty simply. We have access, in France, to the tools offered by Huma-Num, a national research infrastructure for research data in humanities and social sciences.<sup>21</sup> Among them, Nakala is useful: it is a data repository to publish, share and enhance scientific data. With the help of its API<sup>22</sup> and some Python scripts I wrote, adding the images and retrieving the IIIF links was done smoothly [Chiffolleau 2021a].

---

19. <https://omeka.org/>

20. <http://evt.labcd.unipi.it/>

21. <https://www.huma-num.fr/>

22. <https://api.nakala.fr/doc>



Regarding segmentation, the documents had no peculiar layout and followed a pretty regular structure, i.e., a title and ordered paragraphs with lines of equal length. Hence, creating an adapted model was simple and not really time-consuming [Chiffolleau 2020d].

As for the encoding, developing it for my files was helped by the existence of many examples and guidelines. They let me create a model for the structure of the XML tree [Chiffolleau 2020b]. Then, I created several Python scripts to automate part of the process, such as the encoding of the metadata and the body of text [Chiffolleau 2020c], or the encoding and referencing of the named entities [Chiffolleau 2022].

Last but not least, for the publication, my supervisors and I chose, early on, to use TEI Publisher for our work and I quickly learned to handle it [Chiffolleau 2020e]. Afterwards, I generated a functioning application to host digital scholarly editions [Chiffolleau 2021c], which was launched online at the end of 2021 [Chiffolleau 2021d].

Nonetheless, the time I saved on earlier tasks allowed me to focus more on the challenging “Transcription” step. As I mentioned in [Chiffolleau 2020a], several elements of the corpus I was working with made my work more difficult. I can notably mention the production of an efficient text recognition model. Even though I managed to generate it, the “transcription” step remained complicated, as it required a double correction. First, I applied the model. Then, I corrected the prediction semi-automatically, using Python dictionaries and regular expressions, or regex, which are “a language for specifying text search strings [...], an algebraic notation for characterizing a set of strings” [Jurafsky and Martin 2008, p. 5]. With it, I searched erroneous words in the transcription, retrieved them, added their correct version, and applied it back to the text, by using the regex to find the erroneous word and correct it. It was a pretty time-consuming task. Lastly, I had to do another correction, manually this time, since many errors were still present, despite my effort to create an efficient model [Chiffolleau 2020f].

## **Automatic Text Recognition (ATR): A Core Component of This PhD Research**

### **Challenges in Transcription**

Experimenting with and developing the workflow point out the difficulties that currently happen during the “Transcription” step. This revealed a significant gap in our understanding of how neural networks perform text recognition. Automatic Text Recognition is

a method that changed a lot in the last decades. It went from a recognition with bounding boxes and simple algorithms, to lines and regions detection, and training through neural networks. This evolution expanded the range of text recognition, on type of texts, or languages involved.

However, some questions arose from these advancements. With the recognition at line or region level, what does the model learn to recognize? How is the data interpreted and used by the neural network during the model training? Are there some specific elements that have to be in the training data to make a model more efficient?

Providing answers to those questions could help improve the training and the efficiency of the generated models. With my PhD research, I aimed to deepen the global understanding of modern text recognition methods and streamline the workflow. I then decided to dedicate my research to understand better the functioning of this step, while choosing to focus on the least understood aspect: the training process. My goal is to obtain answers about the various elements that take part in the training of a model. Thus, in the future, generating an efficient text recognition model will be much faster and involve less post-process correction.

## Datasets Used in Research Experiments

Practical experimentation on datasets is essential for understanding the operation of text recognition. Hence, constituting one or several datasets of documents to work on is fundamental. For this PhD, I used two different datasets. They were created from the documents of the projects DAHN and EHRI.<sup>23</sup> Before starting my thesis, I worked at Inria, on adapting the workflow for the creation of digital scholarly editions for each project. Regarding the DAHN project, the project aimed at

"facilitating the digitization of data extracted from the archival collections and their dissemination to the public in the form of digital documents in various formats and as an online edition." [Chiffolleau 2021b]

The project was divided between two corpora. One was completely raw and all the steps of the workflow had to be done.<sup>24</sup> The other had already been made into a digital scholarly edition, but needed an update.<sup>25</sup> As for the EHRI project, digital scholarly editions had

---

23. The projects are extensively detailed at the chapter 3 and 8, respectively

24. Paul d'Estournelles de Constant's correspondence: <https://discholed.huma-num.fr/exist/apps/discholed/index.html?collection=pec>

25. The Berlin Intellectuals: <https://discholed.huma-num.fr/exist/apps/discholed/index.html?collection=bi>

already been created<sup>26</sup>. Yet, the project was in need of options and tools to speed up the process and make it easier [Bénière, Chiffolleau, and Scheithauer 2024]. They also required solutions to homogenize the process across editions, which was done mostly with the creation of TEI specifications [Bénière, Chiffolleau, and Romary 2024]. I worked thoroughly on those corpora for several months, and became completely acquainted with them. Therefore, it made sense to use them as datasets for my thesis experiments. It was most natural with the correspondence of Paul d'Estournelles, since I segmented, transcribed, encoded and published it entirely.<sup>27</sup>

## ATR: An Overview of Key Concepts

Based on this corpora, I intend to extend the knowledge on what is called Automatic Text Recognition (ATR), which encompasses both Optical Character Recognition (OCR) and Handwritten Text Recognition (HTR).

OCR was developed for printed documents, e.g., text produced using a printing process, with a standardized format that can be made into consistent and identical copies. In printed documents, the fonts used are typically made of distinct characters. They are separated from one another by at least a slight space.<sup>28</sup> This makes character recognition possible, as every character can be recognized clearly.

While OCR is used for printed texts, HTR is meant for handwritten documents, i.e., text written by hand with a pen. Consequently, the recognition can be harder because handwriting is unique to each person. Moreover, many write in cursive. It means that the characters are not separated any more, and longer sequences then have to be recognized. Those sequences could even appear several times in the text, though they would not always look the same. Handwriting can change even in a same document. The quality of the document can also vary, and the type of ink, pen, or paper used by the writer can make the recognition harder.

In addition to OCR and HTR, a less alluded, third category is typewritten text. A typewritten document refers to a document created using a typewriter. It uses a uniform typeface (`monospaced` usually), on a paper of various size and quality. No option for correction are available once the ink is transferred onto the paper.<sup>29</sup> For its uniformity of

---

26. <https://www.ehri-project.eu/ehri-online-editions>

27. For more details on this work, see my blogposts: <https://digitalintellectuals.hypotheses.org/category/dh-projects/dahn>

28. <https://www.britannica.com/topic/printing-publishing>

29. <https://www.britannica.com/technology/typewriter>

text and font used, typewritten text could be processed through OCR. But elements such as crossed-out parts of the texts, overlapping characters, or some faded characters due to too little pressure on the key makes the recognition with HTR and adapting models more difficult.

## **Typewritten Documents: An Ideal Resource For This PhD**

Therefore, typewritten text occupies a middle ground between OCR and HTR techniques, incorporating elements of both. I chose it as my only type of study for my research. This is both a choice and a requirement. The entirety of the correspondence of Paul d'Estournelles de Constant was written on a typewriter. The majority of the documents of the EHRI Online editions were either written with or transcribed by a typewriter. While focusing solely on typewritten text might seem limiting, it actually provides significant benefits. The uniformity of writing makes the text easier to read and process. It is of help to me, as I am working on the text, and I then do not have to struggle to decipher the characters. It is also of help to the software that will process the source as training data, since the homogeneity of text ease its recognition and learning.

Typewritten text could, furthermore, be considered as a neutral ground for my research. As presented priorly, typewritten text has both the easiness of OCR sources and peculiarities of HTR sources. Testing my hypotheses on such text provides a preliminary basis for further research. It could afterwards be transposed to more tricky sources.

## **Problem Statement and Outline of the Thesis**

I took into consideration the lack of understanding of neural network recognition and the ignorance around the exact implication of the training data. Combined with my thorough knowledge on two extensive datasets made of typewritten documents, I decided to explore more deeply the ground truth, i.e., the matching transcription of an image's text, used as training data for the production of a model. I also chose to explore the raw prediction, i.e. without any post-processing correction, made by the model, and notably its errors, to understand better what affects the accuracy of the model's recognition. With my research, I am trying to answer the following questions: What does the recognition model learn from the content of the ground truth? In what ways does the content of the training data have an impact on the recognition accuracy and results of the model?

To address these questions, my research is organized into four main parts. In the first part, I provide an overview over the state-of-the-art of the various topics relevant to my thesis work. I put a strong focus on the evolution of ATR techniques (Chapter 1) and a presentation of the current trends in ATR (Chapter 2). In the second part, I delve into my first hypothesis, which consisted in evaluating the impact of the lexicon in the training and recognition of the model. After presenting the main corpus for the experiments (Chapter 3), I focus on the methods I used and the results I obtained while observing the content of my ground truth (Chapter 4). Then, I present a comparative analysis on the models I trained (Chapter 5). Following the realization that my hypothesis was wrong, I tried some of those experiments at a different level. This level is the n-gram. First, I render the various ways I observed the n-grams of my ground truth (Chapter 6). I proceed with a thorough presentation of the main experiment I conducted on those n-grams (Chapter 7). Finally, as the results of this experiment were not completely conclusive, I decided to expand my research by working on a new corpus. It brought multilingualism as a new parameter, in order to see the impact of language in what I already observed. Once the new dataset is introduced (Chapter 8), I observe its content and how multilingualism plays into it (Chapter 9). Finally, I focus on the results produced by the model trained from this new dataset (Chapter 10), before concluding on the various progress I made and the additional work that could be done (Conclusion).

## PART I

---

# AUTOMATIC TEXT RECOGNITION: AN EVOLVING TECHNIQUE IN NEED OF UNDERSTANDING



# FROM OCR TO ATR: EVOLUTION OF TRANSCRIPTION TECHNIQUES

---

This thesis aims at expanding the knowledge on model training with neural networks for Automatic Text Recognition (ATR). Model training with neural networks is the technique mostly used nowadays for ATR in the DH community. At first, the domain was limited to Optical Character Recognition (OCR), and the work was dedicated to printed texts, and done with algorithms. ATR was then extended to Handwritten Text Recognition (HTR). The diversity of studied texts and their peculiarities prompted the creation of new techniques that had to perform on handwritten texts; those were oriented towards deep learning. Finally, considering the various texts and methods used, the general term of Automatic Text Recognition (ATR) was adopted. It designates all kinds of recognition done, while using deep learning techniques. In this first chapter, I present in detail the history of this evolution, and I also explain the various recognition methods used historically and in modern times.

## 1.1 OCR: Working with Bounding Boxes and Characters

### 1.1.1 The Development of OCR and Its Purposes

Optical Character Recognition is a process that consists in recognizing printed characters from scanned documents or images. It is done after the document is produced, rather than in real time, i.e. off-line, to obtain information that should be readable both to humans and to a machine [Eikvil 1993, p. 7]. It significantly developed during the 1950s. In the middle of this decade, the very first commercial OCR system was made available, and its purpose was to input sales' report into a computer [Assefi 2016]. Since its initial development, OCR has expanded into a multitude of tasks in various domains, such as



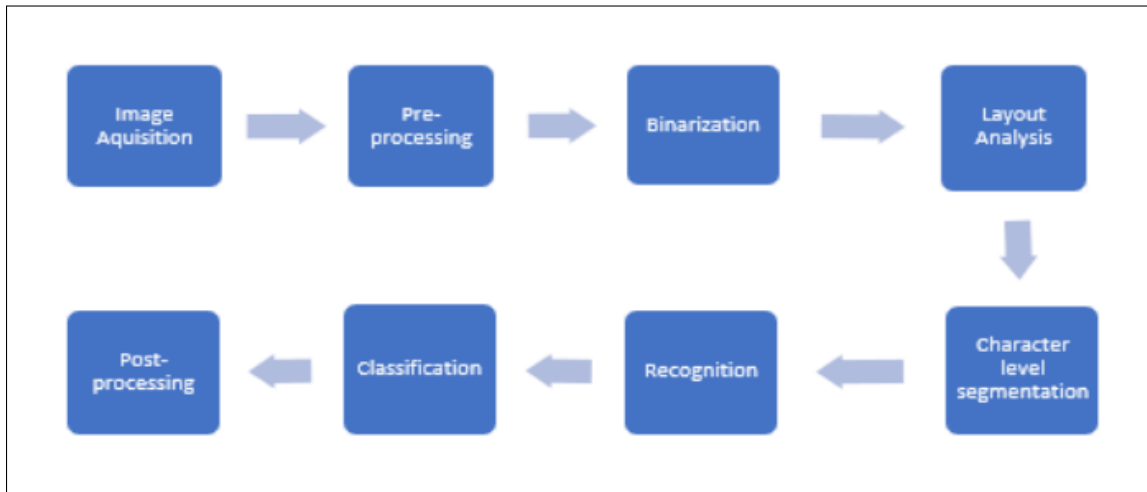


Figure 1.1: Steps of the OCR process

healthcare, finance, insurance, and education [Assefi 2016]. Examples of such applications include recognizing licence plate, extracting image text from natural scene, rectifying the text retrieved from camera captured images [C. Patel, A. Patel, and D. Patel 2012]. Other uses include processing cheques in banks, recognizing barcodes, or extracting information from scanned documents, including printed forms, bank statements, invoices, passport documents, mails, or else [Jain, K. Taneja, and H. Taneja 2021]. For example, [Assefi 2016] detail the use of OCR in healthcare for the digitization of patients forms, in order to easily create databases, extract information and retrieve medical data. It can be of great help with vulnerable patients, such as those with cancer or HIV, by streamlining medical data processing.

As can be observed in the various examples mentioned priorly, despite improvements in accuracy and functionality, the underlying method of OCR has remained largely unchanged.

### 1.1.2 How Does OCR Operate?

The OCR process follows the steps outlined in Figure 1.1. It is a schema created by [Jain, K. Taneja, and H. Taneja 2021] in their paper. While the names and order of the steps can vary across studies, the general process is as such: image retrieval, preprocessing, segmentation, recognition, and post-processing.

The first step is image acquisition. It simply consists in the downloading of an image from an online source, capturing it using a camera, or scanning it [Jain, K. Taneja, and

H. Taneja 2021].

Afterwards, preprocessing can be applied to enhance image and appearance, to increase the system accuracy, which is particularly useful when dealing with noisy images. This preprocessing can take the form of smoothing, which eliminates gaps and holes in the characters, and can also reduce the width of the line. Normalization can also be applied. It standardizes the size, slant, and rotation of the character, as it can hinder the recognition [Eikvil 1993]. Binarization can also be done as part of this preprocessing or as a unique step if preprocessing is not necessary. It consists in changing the rendition of the image into a binary one with only two classes of pixels, black and white, and separates the content from the background, as only the content is useful for the OCR process [Jain, K. Taneja, and H. Taneja 2021].

Once done, the following step is segmentation. It is done in several steps, too: layout analysis and character-level segmentation. The objective is to locate the regions where data have been printed, differentiating it from figure, graphics, or else. Then, the segmentation is done on lines, words, and character. The idea is to isolate each from one another [Eikvil 1993; Jain, K. Taneja, and H. Taneja 2021]. Those isolated parts are called bounding boxes. They have the shape of a rectangle that defines the spatial extent of the element.<sup>1</sup> After isolating individual characters during segmentation, feature extraction identifies key characteristics. [Eikvil 1993] describes it as a way to “capture the essential characteristics of the symbols”, as well as “one of the most difficult problems of pattern recognition”. They expand the subject from the pages 14 to 18, while mentioning elements of feature extraction. It can entail distribution of points,<sup>2</sup> transformations and series expansions,<sup>3</sup> and structural analysis.<sup>4</sup> The main goal is to check the character in all its components, which could be its skeleton, its curves, the distribution of its points, i.e. the way the black and white points are allocated in the pattern. Once all of this information is gathered, the system can deduce and determine what it recognizes, based on the database created during training.

The next step is called classification, which aims to “identify each character and assigning to it the correct character class”. This signifies outputting the character as it is supposed to be [Eikvil 1993; Jain, K. Taneja, and H. Taneja 2021].

---

1. Bounding box, glossary of HarmonisingATR: <https://harmoniseatr.hypotheses.org/glossary-atr#BoundingBoxID>

2. Feature extraction based on the statistical distribution of points

3. Reduction of the dimensionality of the feature vector

4. Extraction of features that describe the geometric and topological structures of a symbol.

Lastly, some post-processing can be applied to refine the output. Post-processing can enhance the classification results using various language models and dictionaries [Jain, K. Taneja, and H. Taneja 2021]. As presented by [Eikvil 1993], since the recognition is done character by character, it does afterwards grouping<sup>5</sup> and create words. They are then verified, using the context, to ensure the validity, followed by error-detection and correction. An example of how that could be working would be this:

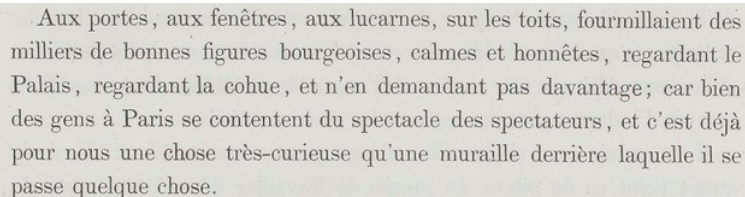
"In the English language the probability of a “k” appearing after an “h” in a word is zero, and if such a combination is detected an error is assumed."  
[Eikvil 1993, p. 21]

Although the OCR process is a clearly established and outlined system, its limitations in handling certain types of text prompted a change of direction, towards more advanced recognition methods.

## 1.2 ATR: The Emergence of Recognition via Neural Networks

### 1.2.1 The Need for a New Recognition Method

As mentioned by [Eikvil 1993], OCR was restricted to constrained writing, where texts follow strict formatting rules allowing high recognition accuracy, such as in Figure 1.2.



Aux portes, aux fenêtres, aux lucarnes, sur les toits, fourmillaient des milliers de bonnes figures bourgeoises, calmes et honnêtes, regardant le Palais, regardant la cohue, et n'en demandant pas davantage; car bien des gens à Paris se contentent du spectacle des spectateurs, et c'est déjà pour nous une chose très-curieuse qu'une muraille derrière laquelle il se passe quelque chose.

Figure 1.2: Printed edition of *Notre-Dame de Paris* by Victor Hugo

However, interest in recognizing unconstrained writing has steadily increased. It includes notably cursive text, where the recognition of individual characters would not be possible, as words are usually written by linking characters with one another, *such as this*, meaning that there is no individuality any more. It is an area where OCR constantly

5. Association of individual symbols that belong to the same string with each other, making up words and numbers. [Eikvil 1993, p. 20]

struggles [Romero, Serrano, et al. 2011]. As presented by the same authors, HTR can be compared to the field of Automatic Speech Recognition. It would be like trying to recognize “continuous speech in a significantly degraded audio file”, although the input is a handwritten line rather than an audio file. Consequently, a new method for the recognition of texts was needed.

Moreover, unconstrained texts often exist in vast collections, representing huge quantities of texts, such as “hundreds of terabytes worth of digital image data” [Romero, Serrano, et al. 2011], making manual transcription impractical, as it would be extremely tedious and time-consuming [Romero, Toselli, et al. 2016]. Libraries, museums, and archives have been preserving digitally and providing access to their various and numerous collections of handwritten historical documents. Yet, no attempt was made to transcribe those documents, and “provide historians and other researchers new ways of indexing, consulting and querying them” [Romero, Serrano, et al. 2011; Romero, Toselli, et al. 2016]. Nevertheless, those collections offer valuable insights into varied topics, such as mathematics, medicine, or religion. They also supply information on ancient everyday activities. From that, they retain the “evolving memory of our societies” [Romero, Toselli, et al. 2016].

For example, I can mention some projects that embody those ideas through different times and spaces. Hours - Recognition, Analysis, Editions, or HORAE, is a project that works on handwritten prayers books owned by rich people in the late Middle Ages. It studies the religious practices during this period, through the exploration of books of hours [Boillet et al. 2019].

The project *LECTure Automatique de REPertoires*, or *LECTAUREP*, examines notary registries from the Central timetable of Paris notaries of the French National Archives. They were written by thousands of hands from 1803 to 1940 [Chagué, Terriel, and Romary 2020].

*Foucault Fiches de lecture*, or *FFL*, is a project dedicated to the study of Michel Foucault’s reading notes. They are handwritten texts written for 30 years of his life [Massot, Sforzini, and Ventresque 2019].

The project TIME US, specialized on work, remuneration, textiles, and home (17th-20th century) is constituted of more than ten thousands printed and handwritten texts. They originate from various collections, such as the Minutes of the Paris Industrial Tribunal, the Violations at the Lyon Arts and Crafts Police, the Lyon Prefecture police reports, or the reports of the hearings of the Council of Industrial tribunal of Lyon published in the workers’ press [Chagué, Le Fournier, et al. 2019].

Lastly, MARITEM (“*Manuscrit du Roi, Paris, BnF fr. 844. Image, Texte, Musique*” ; 2019-2022) is a project working on six hundred and two musical pieces from several musical and linguistic traditions. The pieces vary from songs of *trouvères* and troubadours, motets, instrumental works, or some religious pieces and in languages like old French, old Occitan, old French Occitan and Latin [Mariotti 2020].

In addition to the temporal and geographic diversity of these historical archives, they also present a wide range of writing style and structure, prompting the need for a new recognition method. Among the specificities that make the recognition extremely complicated for OCR, few examples can be mentioned. There are texts separated into columns and with characters that are so alike that it is really difficult to set them apart, as in Figure 1.3 (CREMMA project [Chagué 2021a]). There are lines of text with characters not used any more, as well as uneven spacing, like in Figure 1.4 (Données HTR incunables du 15e siècle [Pinche, Gabay, et al. n.d.]).

Along those difficulties, it is also possible to mention typewritten text with several fading characters, like in Figure 1.5 (Dataset Tapus Corpus [Chagué 2021b]).

In addition to those types, the documents studied by some of the new projects also include handwritten texts, which present a whole new side of recognition. It is demonstrated by the Latin medieval manuscript of Figure 1.6 (CREMMA project [Chagué 2021a]), or the notary public directory of Figure 1.7 (LECTAUREP project [Chagué, Terriel, and Romary 2020]).

With such quantity of varied collections and unique texts, it was important to find the adequate method for the recognition. This is why the solution found was the use of neural networks, as it has the major advantage of having a very adaptative nature [Eikvil 1993, p. 19].

### 1.2.2 A Solution Found in the Use of Neural Networks

Towards the end of the 2000s, researchers in text recognition decided to turn their attention to the field of artificial intelligence and its subdomains, following their evolution and adaptation to tasks such as automatic text recognition [Naiemi, Ghods, and Khalesi 2022]. Artificial intelligence is a field that can be defined as “the effort to automate intellectual tasks normally performed by humans” [Chollet 2017, p. 4].

One of its subfields is machine learning. It aims at adapting certain humans’ tasks to computers, through training, and not programming. A machine-learning model learns to recognize and render elements seen during its training [Chollet 2017, pp. 5–6]. The learn-

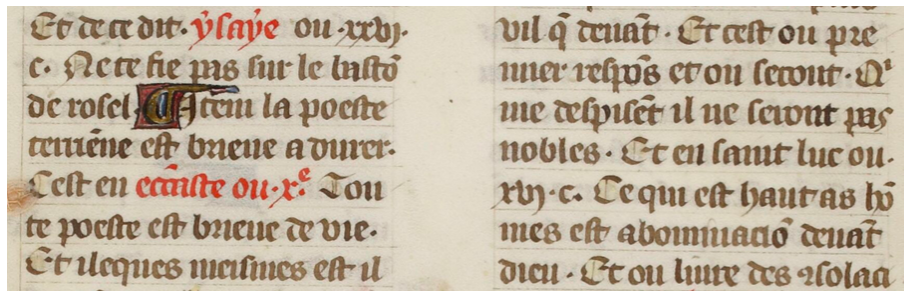


Figure 1.3: Excerpt from a medieval manuscript from the 14th century

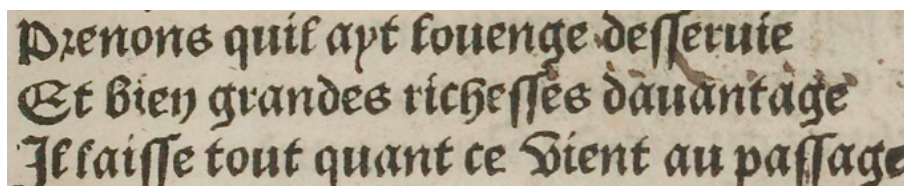


Figure 1.4: Excerpt from an incunabulum from the 15th century

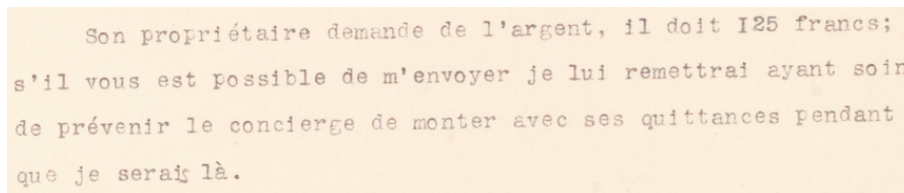


Figure 1.5: Excerpt from a letter from Auguste Delâtre to Edouard Foley

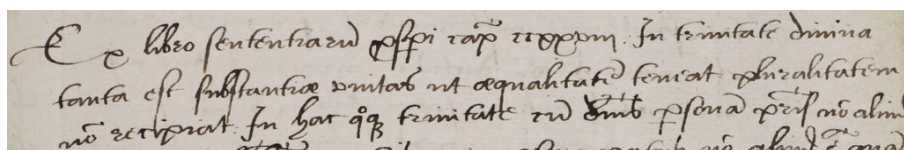


Figure 1.6: Excerpt from a Latin medieval manuscript from the 16th century

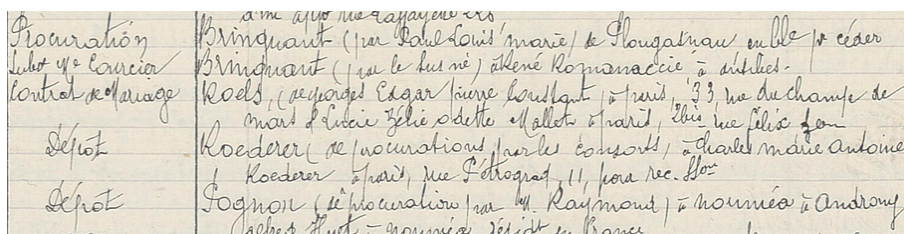


Figure 1.7: Excerpt from a Parisian notary's registries of deeds

ing process is guided by input data, expected outputs, and a performance metric that evaluates the algorithm.

Within machine learning, various approaches can be found, such as the opposites: shallow and deep learning. To get from one to the other, the training has to be made more in-depth, as shallow learning involves fewer layers and simpler models, while deep learning adds more layers of work to process more complex patterns. In deep learning, neural networks are structured in “literal layers stacked on top of each other” [Chollet 2017, p. 8]. It is the technique used for ATR. The method takes its origin from the ambition to simulate the mechanism of learning in biological organisms. It is done through artificial neural networks, containing computation units designated as neurons [Aggarwal 2023, p. 1]. [Haykin and Haykin 2009] defines a neural network, in page 2, as:

"A massively parallel distributed processor made up of simple processing units that have a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects:

1. Knowledge is acquired by the network from its environment through a learning process.
2. Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge"

Many parameters are involved into mimicking the human brain, during a neural network training. Neural networks learn progressively through varied patterns, rather than a monotonous process, with the help of layers and weights' variation. The network is made of input data, expected outputs,<sup>6</sup> a loss function, and an optimizer. The loss function measures the performance of the network. It compares the prediction of the network to the expected output. Then, it produces a score estimating the performance of the network [Chollet 2017, p. 10]. The optimizer uses the result produced by the loss function to update the model's weights. By adjusting the parameters during training, the model's performance improves, reducing the loss score. [Chollet 2017, pp. 11, 58]. With that, it will initiate a training loop. It will repeat many times, until the loss score decreases, and the outputs are as close as the targets. It is explained by [Chollet 2017] and demonstrated in their schema of a neural network from page 11, rendered in Figure 1.8.

While neural network can be highly effective, they can have issues such as overfitting. It designates a situation where a neural network ends up memorizing data rather than just learning patterns. It can cause trouble, especially if noise is present in the training data.

---

6. Those come from training data

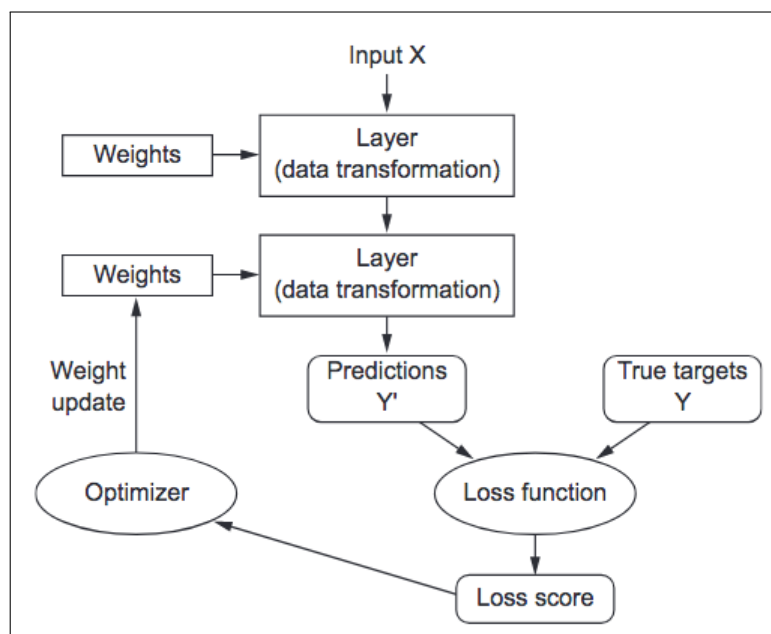


Figure 1.8: Schema of a neural network architecture

In this case, the network “loses the ability to generalize between similar input–output patterns” [Haykin and Haykin 2009, p. 164]. As a text recognition model is supposed to be able to render what it has learned based on its training data, but also to adapt it to new texts, overfitting would be inopportune. To address it, two methods are commonly used. They are part of regularization, a process created to stabilize solutions by incorporating prior knowledge and imposing additional constraints [Haykin and Haykin 2009, p. 315]. Those two methods are early stopping and dropout. Early stopping implies to hold out a portion of the training data as a validation set. The model errors on the validation are monitored. At a certain point, when the errors are less in the training data than in the validation set, the model is considered to be about to overfit. Then, the training is terminated [Aggarwal 2023, p. 188]. This is a solution rather outside the training procedure, contrary to dropout. This implies dropping out units, hidden or visible, randomly, in the neural network. It is then temporarily removed from the network, input and output connections included. It prevents the units from co-adapting in excess and forces the network to rely on multiple neurons, making it more robust [Srivastava et al. 2014]. Many ATR systems allows the implementation of those elements of regularization. In addition to regularization techniques, ATR systems allows adding or removing layers from the neural networks, through the Variable-size Graph Specification Language (VGSL) network



specification, enabling the specification of different network architectures, to suit specific tasks.<sup>7</sup>

Following those notions of neural network process, a model training works as rendered in Figure 1.9. An input is given to the network. It is typically a couple of images and transcription. The former is accepted in various formats (JPG, PNG, TIFF) according to the software. The latter is allowed in TXT or XML format. It goes then into a cycle. All the training data elements are processed until the software can give a progress report. Each time this cycle finishes, it is referred to as an epoch [Aggarwal 2023]. During the cycle, the training loop will train, test and validate its progress. The training data, the input of the beginning, will go through the loop, but not all at the same time. Most of it is learned during the training step. Then, the model produced will be tested on it. Lastly, the model is checked on a portion of it, called the validation set. It was created by removing it before the beginning of the training, to verify the recognition's skills of the model on unseen data [Aggarwal 2023, p. 169]. Once it has done enough epochs to consider that it will not progress enough, it produces an output. This is the model that will be subsequently used for the text recognition.

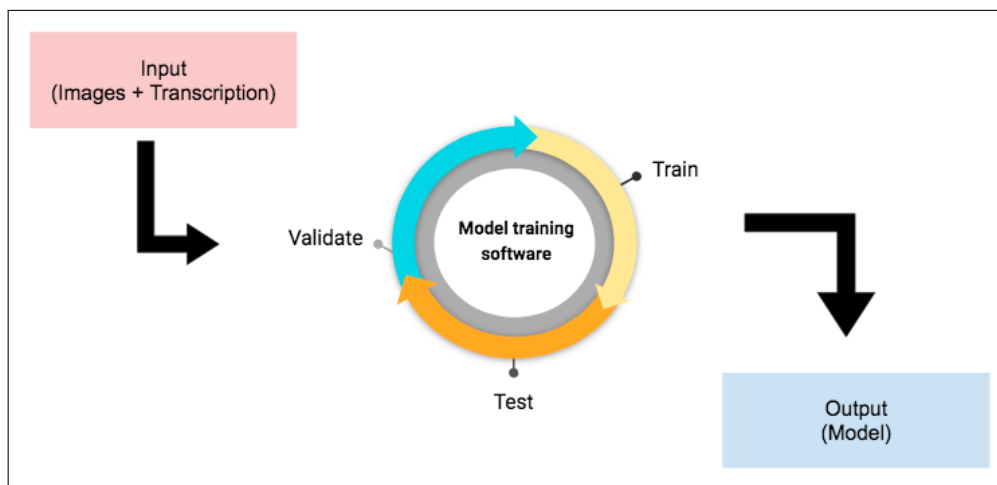


Figure 1.9: Schema of a model training

7. VGSL network specification: <https://kraken.re/main/vgsl.html>

# A NEW HORIZON OPENING WITH THE EMERGENCE OF DEEP LEARNING APPROACHES

---

The evolution of Optical Character Recognition (OCR) to Automatic Text Recognition (ATR) and of the techniques attached to it led to plenty of innovations and research. Those new methods and features spurred the development of new tools. This raised interrogations about the various fields encompassed by ATR. To address these developments, this chapter details extensively the various ATR software that were created to adapt to those innovations. It also highlights diverse studies conducted by the Digital Humanities (DH) community to broaden the knowledge on the domains that contribute to effective text recognition.

## 2.1 With Great Changes Come Great Software

### 2.1.1 From Proprietary to Open Source, from Generic Algorithms to Neural Networks, from Printed to Handwritten

For a few decades now, OCR has been effectively performed, thanks to a proliferation of software for this task, that also offer choices to the user, especially, when considering that they do not propose the same features, pricing models, or accuracy depending on the input. In their article written in 2021, [Jain, K. Taneja, and H. Taneja 2021] present a comparative study of OCR toolsets. They explicitly present the three distinctive categories of software that exist: proprietary, online, and open source.

The first category, proprietary, usually referenced commercially produced software, where the internal algorithms and functioning are not disclosed. Among the leading proprietary OCR software in this category, there are, for example, [ABBYY FineReader](#) and

**Transym**. They have been the subject of an evaluation in [Assefi 2016]. Their OCR services were tested on a dataset of more than one thousand images, and on the field of healthcare informatics. Both tools are very good at information retrieval from digital images. They are also equipped with a Graphical User Interface (GUI). The former is available online, while the latter needs to be downloaded [Assefi 2016; Jain, K. Taneja, and H. Taneja 2021]. ABBYY FineReader is particularly skilled at layout analysis and OCR. Transym is able to read “blurred, obscure, and even broken characters” [Jain, K. Taneja, and H. Taneja 2021, pp. 2–3].

Moving on to the online category, it is possible to mention **Google Docs OCR**. It is a cloud document storage working with Google Drive [Jain, K. Taneja, and H. Taneja 2021, p. 4], that can convert different types of image data into editable text data [Assefi 2016, p. 737]. Contrary to ABBYY FineReader and Transym, this OCR service is free, like it is the case for most online service [Jain, K. Taneja, and H. Taneja 2021, p. 7].

In contrary to proprietary systems, the third category, open source, is probably the most prominent, because it fits in an era where the keyword is becoming open science and the researchers are prompted to share their data and their results. Although it doesn’t have any GUI and work only on Command-Line Interface (CLI), one of the more extensively tested tools is **Tesseract**. It was used first in its version 3, then in version 4. It offers many interesting features, such as multilingual support for more than a hundred languages [Jain, K. Taneja, and H. Taneja 2021, p. 6]. Its architecture and the way it operates is thoroughly presented in [Smith 2007] and [C. Patel, A. Patel, and D. Patel 2012]. It was also the object of comparison. First, it was tested against the proprietary software ABBYY FineReader in [Heliński, Kmiecik, and Parkola 2012]. Then, it was tested against Transym in [C. Patel, A. Patel, and D. Patel 2012]. It was also tested against many other software in [Jain, K. Taneja, and H. Taneja 2021, pp. 8–9]. In the majority of those experiments, and faced with texts presenting peculiarities, Tesseract seemed to always be the better software, and it is also notable for its continuous evolution. It became a software working with a recognition engine based on neural networks in its version 4 [Jain, K. Taneja, and H. Taneja 2021, p. 3]. While Tesseract seems to be a performant open source tool, it has some difficulties when it comes to historical material [Reul et al. 2019, p. 12], which can present peculiar layout, fonts, or text structure. However, this capability is crucial for ATR in Digital Humanities.

Two other tools are worth mentioning from the open source category: **OCRopus** and **Calamari**. OCRopus, presented in its first version in [Breuel 2008], has evolved since, and

it is now in a third version, working with deep learning techniques [Reul et al. 2019, pp. 12–13]. Calamari is an OCR engine, usually integrated into OCR workflow. The only task it can do is recognizing text from text line images [Jain, K. Taneja, and H. Taneja 2021, p. 4]. For example, it is the case for OCR4all presented in [Reul et al. 2019, pp. 7–8]. Experiments done by [Reul et al. 2019] and [Jain, K. Taneja, and H. Taneja 2021] demonstrated that those two software have rather good result with historical material, like early printed books from the 15th century.

One key benefit of the shift from generic algorithms to neural networks is the introduction of the training data option. It means that, while an OCR software user was limited to the fonts integrated in the recognition engine, now they have the possibility to adapt the engine to their data. This is an option provided by the three open source software mentioned. However, they are lacking an element: a GUI.

GUI can be essential for researchers in humanities that are not familiar with computer systems. As mentioned previously, it is present with the proprietary systems, but those are not ideal in the field of humanities where open science is the preference.

Therefore, in order to efficiently and easily work on various historical documents, it is essential to find a software that integrates three key elements: (1) It has the ability to create data on a GUI, (2) can train a model from it via neural networks, and (3) work can be done on printed, typewritten or handwritten texts.

### 2.1.2 Transkribus vs eScriptorium: The Battle of the HTR Software

At the beginning of my PhD research, only two software stood out as leading options due to their combination of the three key elements mentioned above: [Transkribus](#) and [eScriptorium](#).

Transkribus is a platform created in 2013 as part of the tranScriptorium project [Sánchez et al. 2014]. It has been continued, since January 2016, in the READ project, later becoming [READ COOP](#) [Kahle et al. 2017].

Transkribus has many features: “AI text recognition”, “custom AI training”, “field & table recognition”, “powerful text editor”, and “publishing & search tool”.<sup>1</sup> Transkribus is able to work on printed, typewritten and handwritten documents with the same ease and accuracy. It can process large datasets of any size.<sup>2</sup> This is helped by the presence of

---

1. <https://www.transkribus.org/#features>

2. <https://www.transkribus.org/ai-text-recognition>

many public AI models (more than 150).

More recently, transformer-based Handwritten Text Recognition (HTR) models were also created. They are trained like Large Language Model (LLM)<sup>3</sup> with way more data than a standard recognition model. This makes them effective when working with datasets containing multiple authors, and they are also efficient with handwritten styles never seen before [Ströbel, Clematide, Volk, and Hodel 2022].

Although many models are available, it is also possible to train custom AI models on Transkribus, that helps create training data faster, as well as collaborating and sharing your work to not having to do it alone.

Once the model is trained, various tools are proposed to evaluate the quality of the model.<sup>4</sup> Moreover, for the transcription, external help can be requested, whether it is with the collaborative features or with the transcription editor for crowdsourcing. It is also possible to produce digital scholarly editions by transcribing, editing and annotating the corpus.<sup>5</sup> In terms of segmentation, Transkribus has been trained to enhance information extraction.<sup>6</sup> It is feasible thanks to an advanced field and table recognition, with the help of LLM. Those models are able to combine field and table recognition: they recognize tables and lists in various formats, and they also help find semantic information<sup>7</sup> when the layout structure is insufficient.<sup>8</sup> Regarding output options, Transkribus allows exporting in PAGE<sup>9</sup> and ALTO XML,<sup>10</sup> in TXT and DOCX, in PDF (with some layers<sup>11</sup>) and also with a TEI encoding.<sup>12</sup>

Statistically, Transkribus is more than 200k HTR AI models trained, +50M of pages processed and +150 free AI HTR models.<sup>13</sup>

---

3. Large language models are the result of language models submitted to a pretraining, which means "learning knowledge about language and the world from vast amounts of text. Large language models exhibit remarkable performance on all sorts of natural language tasks [...]. They have been especially transformative for tasks where we need to produce text, like summarization, machine translation, question answering, or chatbots." [Jurafsky and Martin 2008, p. 214]

4. <https://www.transkribus.org/ai-training>

5. <https://www.transkribus.org/transcribing>

6. Information extraction (IE) consists in turning "the unstructured information embedded in texts into structured data, for example for populating a relational database to enable further processing." [Jurafsky and Martin 2008, p. 415]

7. Information about the linguistic meaning of the text

8. <https://www.transkribus.org/information-extraction>

9. [http://www.primaresearch.org/publications/ICPR2010\\_Pletschacher\\_PAGE](http://www.primaresearch.org/publications/ICPR2010_Pletschacher_PAGE)

10. <https://www.loc.gov/standards/alto/>

11. It is possible to choose how to include the image and the transcribed text in the PDF file, with searchable text possibility

12. <https://help.transkribus.org/downloading>

13. <https://www.transkribus.org/>

eScriptorium is a platform created in November 2018. It is designed to facilitate Automatic Text Recognition (ATR) in a user-friendly environment, using the software Kraken. It has been under development since 2017 [Kiessling et al. 2019]. Kraken has been created by Benjamin Kiessling, as part of the Scripta PSL programme at the PSL University [Kiessling et al. 2019].

Developed from a fork of the OCRopus software [Breuel 2008; Reul et al. 2019], it has many features such as recognition, layout analysis and script detection [Kiessling 2019]. It also offers the possibility to train models,<sup>14</sup> by using the parameter “ketos” in the command line.

Moreover, for the recognition training, the tool proposes many options to choose. With them, we can obtain a model as adapted to our need as possible. Users can pick:

- the output name;
- the number of epochs for the training;
- the partition ratio between training and validation;
- the format type of the training files (XML, binary, etc.).

The software also allows, when fine-tuning a model, to determine the type of codec<sup>15</sup> desired for the model. Lastly, it is also conceivable to manipulate the VGSL specification<sup>16</sup> of the neural network that will be used for the training, by modifying the layers.<sup>17</sup>

With Kraken as its core, eScriptorium integrates these functionalities into a graphical interface that offers additional tools for managing documents, metadata, and transcription. With it, it is possible to create projects and documents, to which tags can be added to classify them. In documents, which are basically folders of images, metadata (information about the content), and ontologies<sup>18</sup> (the structure and relationships between concepts)

---

14. <https://kraken.re/main/ketos.html>

15. Stands for “coder-decoder” or “compressor-decompressor”. This is the process of encoding, by converting input text or images into a format suitable for processing, then decoding, by interpreting the processed data to generate human-readable text.

16. Specification of different network architectures for image processing purposes using a short definition string, that consists of an input block, one or more layers, and an output block.

17. <https://kraken.re/main/ketos.html#recognition-training>

18. “Computational ontologies are a means to formally model the structure of a system, i.e., the relevant entities and relations that emerge from its observation, and which are useful to our purposes [...] The backbone of an ontology consists of a generalization/specialization hierarchy of concepts, i.e., a taxonomy.” [Staab and Studer 2009, p. 2]

for the folder can be chosen before adding any document [Kiessling et al. 2019]. The platform offers the possibility to add images in various formats (IIIF, PDF, JPG, etc.). Once it is done, automatic binarisation, manual or automatic segmentation and/or transcription can be carried out. The segmentation and transcription can be modified afterwards:

- The masks, zones, and polygons can be modified;
- The lines order can be rearranged;
- The prediction can be modified to correct the possible errors.

Once done, the output can be exported in various formats (ALTO, PAGE, TEXT). It can be downloaded with or without the images and with some informative metadata. The models trained on eScriptorium can be exported after training. It is also possible to use public models accessible on the instance.<sup>19</sup>

eScriptorium has several “instances”. It is possible to download a desktop version or to use some online versions available. One of them, generated for the Consortium pour la Reconnaissance d’Écriture Manuscrite des Matériaux Anciens (CREMMA), launched about a year ago. It represents, statistically, according to the managers of the instance, around 1500 models, segmentation or text recognition, trained or uploaded, +410k images treated by the system and 25 different scripts worked on.<sup>20</sup>

Although Transkribus and eScriptorium differ in some aspects, particularly in their approaches to ATR, they share several useful features. Both platforms provide a virtual keyboard that simplifies the inclusion of special characters, a particularly helpful feature for transcribing historical or non-standardized texts. I can also mention the side-by-side transcription view, that allows you to easily spot the part of the image currently transcribed.

Despite these shared features, one key difference lies in how models are handled. While models created on Transkribus are usually kept private, eScriptorium models are sometimes openly shared, such as through platforms like Zenodo.<sup>21</sup> To the contrary, there is a shift regarding the ground truth produced to generate those models. HTR-United<sup>22</sup> provides a catalogue of training dataset’s metadata where this ground truth can be accessed.

---

19. <https://escriptorium.readthedocs.io/>

20. <https://cremmacall.sciencescall.org/>

21. [https://zenodo.org/communities/ocr\\_models/](https://zenodo.org/communities/ocr_models/)

22. <https://htr-united.github.io/index.html>

When we browse the registry of datasets,<sup>23</sup> there are mostly training data made by “eScriptorium + Kraken”, but there are also many cases where Transkribus was used.

In terms of usage statistics, Transkribus far surpasses eScriptorium (+150k versus +500 (CREMMA only)). [Nockels et al. 2022] also retrieved, for a “systematic review of Transkribus in published research”, 381 publications between 2015 and 2020, with 140 papers with direct mentions of Transkribus. This demonstrates a wide diffusion of Transkribus, with no country borders, and there are as well many example cases of use of the software for various projects.

Ultimately, despite those numbers and observations, I chose eScriptorium due to its direct connection to my work at the ALMAnaCH team, where several colleagues are involved in its development and maintenance. Moreover, two colleagues and I worked on creating an exhaustive documentation on eScriptorium,<sup>24</sup> that presents its features, explains how it works and gives some useful tips, obtained from our experience with the tool [Chagué, Chiffolleau, and Scheithauer 2024].

## 2.2 New Techniques, Emerging Concerns, Innovative Solutions

### 2.2.1 Novel Approaches to Segmentation

The rise of deep learning approaches and the growing number of transcription projects have expanded the scope of Automatic Text Recognition, also bringing new challenges in segmenting complex document layouts. Outside the varied writing styles presented in corpora, unusual or highly complex layouts also appeared. These complexities necessitated the development of advanced segmentation techniques.

In terms of OCR and printed documents, segmentation was relatively straightforward, as it typically involved recognizing sequential straight lines of text, as demonstrated in Figure 1.2. On the contrary, HTR presented unique challenges:

- Documents with double columns, as shown in Figures 1.3 and 1.7;
- Text written in various directions and places in the page;
- Pages with header, footnotes, or other peculiar elements.

---

23. <https://htr-united.github.io/catalog.html>

24. <https://escriptorium.readthedocs.io/>



As a result, developing innovative segmentation techniques became essential to ensure that no information was lost during the transcription process

[Tensmeyer and Wigington 2019] and [Coquenet, Chatelain, and Paquet 2021] wrote about the idea of changing the level of segmentation. They are moving away from line-level, the level used mainly for the training data for ATR models produced with neural networks. They are engaging towards paragraph-level and even whole documents. Yet, their approach and applications differ.

[Tensmeyer and Wigington 2019] focused on documents that have already been transcribed, at page-level without any layout information, but could be used as ground truth. It is a topic that will be developed in the following subsection. As they presented it, many projects have been transcribing their documents without caring about the line formatting. It means that the text is just page-level transcription. Their goal was to make those ground truth still usable, despite the lack of formatting. Indeed, large quantity of text already transcribed is always welcomed to generate adequate model, especially, if it is enabling the development of an HTR model that would propose an alignment that would predict the line breaks and make those training data subsequently exploitable. The authors also emphasize that “training without line breaks would reduce the cost of future manual annotation for HTR, as annotators would not need spend time on preserving the text formatting” [Tensmeyer and Wigington 2019]. The idea here would be to prevent some researchers from doing unnecessary work.

[Coquenet, Chatelain, and Paquet 2021] aimed at submitting new types of segmentation for single column documents, as they encountered some difficulties with the current methods of segmentation. Afterwards, the idea was to expand it to even more complex layout documents. Although the methods they suggest functioned rather well with the documents they use, they admitted that there are some limits to the possible utilization of the tools they present.

While the previous methods focus on improving segmentation for traditional text-based models, more recent approaches are turning to object detection techniques, such as [Romanello and Najem-Meyer 2022] and [Clérice 2023], and even more specifically to YOLOv5, instead of the pixel-classification based polygonization done in the papers mentioned before.

[Romanello and Najem-Meyer 2022] worked on documents with complex layout, historical commentaries. They required a new kind of page layout analysis. In their paper, they explored two approaches, textual or visual:

- One uses transformers, a deep learning architecture generally used for LLM.
- The other uses object detection, a “computer vision task that deals with detecting instances of visual objects of a certain class (such as humans, animals, or cars) in digital images” [Zou et al. 2023, p. 1].
- There are also some hybrids of both.

Their experiment show notable accuracy gains, especially for historical document layout recognition and YOLOv5.

[Clérice 2023] intended to improve the layout analysis for small datasets. Notably, it is wanted for Kraken, as it has some difficulties in that area. In order to do so, they developed the package YALTai (You Actually Look Twice at it), which is meant to combine YOLOv5 with Kraken, in order to replace its segmentation method. Indeed, it is largely beaten by YALTai in their experiments. However, some challenges are still there, as YALTai seems to be limited to rectangle boxes, and does not appear to work well with skewed documents.

Both [Romanello and Najem-Meyer 2022] and [Clérice 2023] use the SegmOnto controlled vocabulary for layout analysis, introduced in 2021 [Gabay and Pinche 2021]. The objective is to annotate the common material features of the document, while facilitating interoperability between DH projects. To do so, two levels of description are used: “zones (main text, notes, figure, damage, seal. . .) and lines (default, musical, interlinear, rubric, drop capital)” [Gabay and Pinche 2021].

In the context of the COLaF (*Corpus et Outils pour les Langues de France*) project,<sup>25</sup> [Clérice et al. 2024] developed the LADaS (Layout Analysis Dataset with SegmOnto) dataset and associated annotation guidelines by adapting the SegmOnto controlled vocabulary.<sup>26</sup> The objective of LADaS is to have a diachronic dataset with as many different layouts as possible [Clérice et al. 2024]. Among other subsets, LADaS has received an input from the DataCatalogue project [Scheithauer, Bénére, and Romary 2024].<sup>27</sup> The adaptation of SegmOnto allowed them to train YOLO object-detection models for LADaS and DataCatalogue with homogenized annotations for both datasets [Clérice et al. 2024]. Following the rules presented by [Gabay and Pinche 2021] and [Clérice et al. 2024] for the annotation of SegmOnto, the controlled vocabulary was also applied to some collections of the EHRI Online Editions that are used for some experiments of this thesis. The goal is to

---

25. <https://github.com/DEFI-COLaF>

26. <https://github.com/DEFI-COLaF/LADaS>

27. <https://github.com/DataCatalogue/datacat-object-detection-dataset>

create a semi-automatic layout analysis for the collections, where the user will verify after an initial automatic phase, as it was presented in [Chiffolleau and Scheithauer 2024]. Done by another DH member of the ALMAnaCH's team, Hugo Scheithauer, with [Roboflow](#), an example of controlled vocabulary for an EHRI image is shown in Figure 2.1.

### 2.2.2 Creating Efficient and Sufficient Ground Truth: An Enigma

The ground truth designates the exact match of the transcription of an image's text. It is used as data to develop a recognition model trained on those transcriptions. With OCR, various models were available, and training adapted ones was not possible. Therefore, the ground truth was not a topic heavily discussed and questioned. Among the reflections brought up with the increasing use of Automatic Text Recognition with neural networks, and the wide variety of texts to exploit, an emphasis was put on developing proper ground truth. They had to be in enough quantity and quality to produce an efficient model. In order to do so, there are several criteria and parameters to consider:

- the quantity of data;
- the language(s) of the documents;
- the style of writing;
- the structure of the source.

There are still some questions on whether the neural networks learn from the paragraphs, the lines of texts, the words, and/or the characters, in sequences or one by one. Then, determining the right quantity remains an issue. Those are some of the ideas covered by [Gabay, Clérice, and Reul 2020], in their article presenting the creation of ground truth for French prints of the 17th century. Focusing on corpus building, the article questioned the difference between quantity and quality. They also explored the choices to make when creating ground truth. Similarly, this quantity question was also mentioned by [Gatos et al. 2014]. Despite having transcribed many manuscripts, themselves, or even by using crowdsourcing to gain time, the ground truth was still not enough to generate sufficient HTR results. [Ströbel, Clematide, and Volk 2020] also reflected on this idea of quantity. They asked clearly the question: "How much data do you need?". They proceeded to various tests with the ground truth created, since they wanted to see the efficiency and accuracy with the amount provided for the experiment.

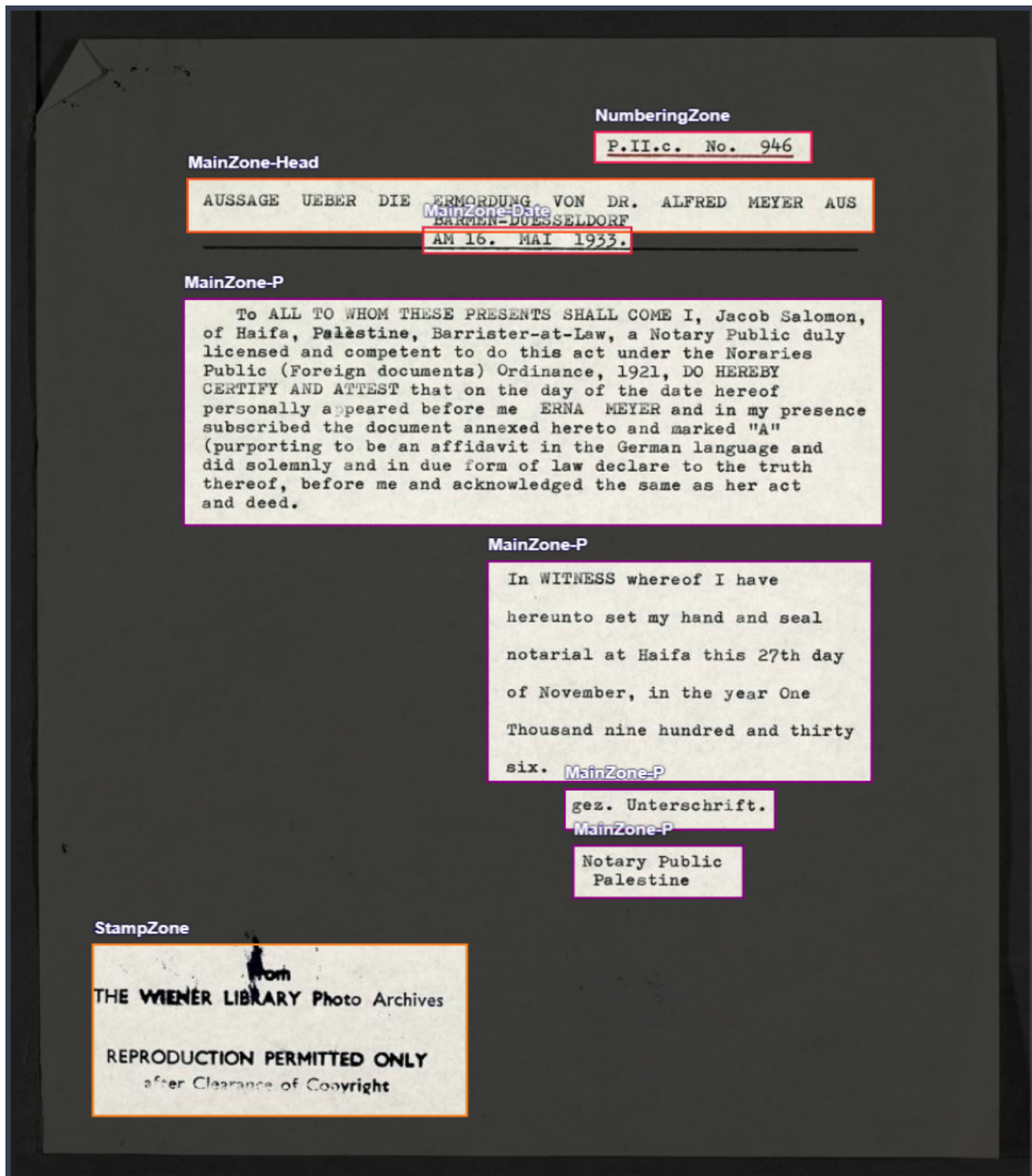


Figure 2.1: An EHR document annotated at region-level with the SegmOnto vocabulary

Another reflection that can be pondered for the ground truth is that of the variety of texts, wondering what to do when the texts of work are substantially different from one another. The question has also been worked on by [Gabay, Clérice, and Reul 2020]. It had texts from the same century, but with pretty heterogenous structure, as well as differences in fonts. However, their experiment gave promising results despite the heterogeneity, and provided valuable insights. In [Springmann et al. 2018], the idea of sources variety was also mentioned. They presented many corpora of diverse centuries and styles. Afterwards, they are used as subsets to produce HTR models.

Although the question of quantity and heterogeneity of sources remain an issue, all those articles agreed on one element: having transcription rules is essential. In order to not complicate the training, it is important, according to [Springmann et al. 2018], that “the same glyph must have the same transcription, even if the glyph has different context dependent meanings.”. Respecting the diplomatic transcription, as in the transcription that reproduces the structure of the images, is also crucial. Following those rules made it easier to reuse and share the ground truth.

To alleviate issues and difficulties raised by having to select and create ground truth, and to help the ATR community, Alix Chagué and Thibault Clérice launched [HTR-United](#). It is an organisation to share ground truth [Chagué, Clérice, and Romary 2021], through a catalogue of metadata on training datasets, as well as the toolbox to help with the datasets [Chagué and Clérice 2023].

The toolbox is made of four tools:

"HTRuc (controls the validity of the htr-unique.yml files and helps build the main catalogue file); HTRVX (controls the validity of the XML files (including to ontologies like SegmOnto and presence of empty elements)); HumGenerator (computes metrics (files, regions, lines, char[acter]s), creates nice badges to display them, updates htr-unique.yml); ChocoMufin (controls char[acter]s in a dataset, converts char[acter]s according to a conversion table)." [Chagué and Clérice 2023]

The catalogue contain numerous training datasets. It is provided with a description of their content, a link to their data, as the catalogue only references the datasets, and their authors. It also mainly procure metadata about the datasets:

- languages;
- scripts;

- period of writing;
- style of writing;
- number of hands that were involved in writing the original source;
- tool used to transcribe;
- the volume of the dataset (files, regions, lines, and characters).

The catalogue also offers the possibility to filter it by one or several of those options.<sup>28</sup> Creating ground truth is often a long and time-consuming task, but it is necessary. Using already available ground truth, and having access to much information about them, already allows the members of the ATR community to ease their work. Moreover, this organisation is perfectly aligned with the direction of open science promulgated by the DH community. They freely and openly share ground truth for text recognition. This is why the datasets that will be mentioned in this thesis have already been shared in HTR-United, as can be seen with Figure 2.2.

HTR-United represents a more-than-welcome helping hand in the creation of ground truth and the generation of efficient models in this new era of ATR and neural networks. Several researchers also tried some texts and experiments to resolve parts of the enigma. There are still many questions that remain about what to do to obtain ground truth that will, right away, produce a sufficiently accurate text recognition model. My experiments aim to address these challenges and accelerate model training.

### 2.2.3 Post-ATR Correction and Prediction Errors: What Does It Entail?

Automatic text correction is not a perfect technique, and, even with a model with 95-98% of accuracy, a prediction will most likely have few or many errors. If the transcription is intended to be used as ground truth for digital scholarly editions, it is then necessary to correct it.

This task has several names, such as post-ATR correction, post-OCR processing or OCR post-correction. It always signifies the same idea: correcting the prediction errors after the application of a text recognition model. As it is a task that could be really time-consuming, researchers have been working on easing these techniques.

---

28. HTR-United catalogue: <https://htr-united.github.io/catalog.html>

**DAHN Corpus**

DAHN  
1914 - 1924

[Link](#) Data repository

---

Language fra

Script Latn

Script Type only-typed

Hands less-than-11

Volume 475 849 characters

Volume 547 files

Volume 12 539 lines

Volume 527 pages

Volume 547 regions

Known characters (NFD) 92

License CC-BY 4.0

Software eScriptorium + Kraken

OCR ground Truth dataset based on French 20th typewritten letters

**Authors:** Chiffolleau, Floriane

Complete record

# Tweet

(a) DAHN

**EHRI Dataset**

European Holocaust Research Infrastructure  
1936 - 1958

[Link](#) Data repository

---

Language eng

Language ces

Language deu

Language slk

Language hun

Language pol

Language dan

Script Latn

Script Type only-typed

Hands unknown

Volume 252 files

Volume 540 645 characters

Volume 9 203 lines

License CC-BY 4.0

Software Unknown [Automatically filled]

Multilingual dataset from various corpus of the EHRI project

**Authors:** Floriane, Chiffolleau and Sarah, Beniere and Michal, Frankl and Wolfgang, Schellenbacher and Zoltán, Vági and Gábor, Kádár and Magdalena, Sedlická and Miriam, Schulz and Christine, Schmidt and Jessica, Green and Martina, Ravagnan and Daniela, Bartáková and Judith, Levin and Daphna, Sehayek and Michał, Czajka and Marta, Wojas and Dagmara, Chelstowska and Winfried, Garscha and Claudia, Kuretsidis-Haider

Complete record

# Tweet

(b) EHRI

Figure 2.2: Entries of the PhD datasets in the HTR-United's catalog

First, manual approaches were proposed. Then, (semi-)automatic approaches were submitted. It was even the assignment of competitions, such as during the conferences ICDAR2017 [Chiron et al. 2017] and ICDAR2019 [Rigaud et al. 2019], and for the workshop ALTA2017 [Mollá-Aliod and Cassidy 2017]. Among the researchers that presented their work on the topic, [Dannélls and Persson 2020] and [Lin and Ledolter 2021], respectively worked on Swedish texts and U.S. Congress debates. They exposed similar techniques, focused on error detection and correction, and used candidates and Levenshtein distance,<sup>29</sup> with the objectives to not be bound to a single software and to try to not be too time-consuming.

In order to be effective with the post-OCR correction and to choose the right method, it is also important to be able to identify the type of mistakes rendered by the prediction. [T.-T.-H. Nguyen et al. 2019] worked on fixing this shortage, since “to this date, few analyses were done to uncover common characteristics of OCR errors, and they all have been on a coarse level”. They distinguished five main types of analysis for the errors:

- Edit operations involve an observation of the substitutions, deletions, or insertions that could have been done;
- Length effects are targeted both at word and token lengths;
- Erroneous character positions identify the place in the work of the misspelling (first/middle/last), which can give some interesting details;
- Real-word vs. non-word errors differentiates errors that are valid in the dictionary but not in the context of the sentence, and errors unrecognized by a dictionary;
- Word boundary focus on the errors created by the addition or deletion of a space.

Regarding the identification and analysis of those types of errors, [T.-T.-H. Nguyen et al. 2019] consider it an “initial step to further analyses or towards more efficient and robust post-OCR techniques”. The same authors expanded their work in [T. T. H. Nguyen et al. 2021]. They brought up the disadvantages of noisy output due to prediction, and also mentioned difficulties for tasks of natural language processing or information retrieval. In order to remedy that, they exposed the various techniques that are used for post-OCR correction, which can be manual or (semi-)automatic. They mentioned several methods:

- using error models;

---

29. For more details, see subsection 5.1.1



- doing lexical approaches;
- exploiting topic-based, statistical, or neural network-based language models.

Furthermore, they referred to various approaches, like isolated-word or context-dependant. Finally, they also introduced metrics to evaluate the error detection, and toolkits to proceed to the detection and the correction. They provided, as well, guidelines on what to use and when. Focusing on a similar topic, my thesis' experiments dedicate a large part to observing and understanding prediction errors. The objective is to be able to correct them, not afterwards, like with the post-OCR correction, but before they even occur, by improving the training data.

This little overview, of the various topics and elements observable in Automatic Text Recognition and its evolution, cleared some leads. They deserve to be explored to understand better how this new recognition method operates.

To do this, I centred my research around the study of the ground truth used as training data and the prediction errors rendered after the application of the model generated from it.

In the first place, my experiment focused on the lexicon content that constitutes the training data.

## PART II

---

# THE (NULL) INFLUENCE OF THE LEXICON



# THE CORRESPONDENCE OF PAUL D'ESTOURNELLES DE CONSTANT: A THEMATIC CORPUS TO LEAD OUR STUDY

---

Properly conducting an experiment requires a fitting corpus, in terms of quantity or quality, to work with. The corpus also needs features that allow for multifaceted analyses, from multiple analytical perspectives. To address the requirement for a rich corpus, the majority of my work was done on the correspondence of Paul d'Estournelles de Constant, a source I have been exploring and analysing in depth for a few years now. Therefore, I considered it to be an ideal material for the experiments in my thesis. This chapter is meant to explain the context and essence of this corpus. It is followed by detailing elements, exposing the ways in which the dataset has been meticulously studied.

## 3.1 History and Presentation of the Corpus

### 3.1.1 Historical Background

In 1914, the assassination of Archduke Franz Ferdinand ignited a conflict between two major alliances, the Allies and the Central Powers. This conflict escalated into the Great War, later called World War I, an opposition that involved many countries worldwide. Though the war spanned many regions, one of the primary battlefields was located in France. This war was one of the deadliest in history, largely due to the industrialization of warfare, which introduced new technologies and mass-produced weapons, and to the four long years that it lasted [Audoin-Rouzeau and Becker 2013].

While the war raged on, there were also strong movements for peace and diplomacy,

and a group was actively engaged in the promotion of diplomatic relations between countries. This was demonstrated by the organisations of peace conferences, such as The Hague Conventions of 1899 and 1907, the creation of the Permanent Court of Arbitration in 1899 or of the Carnegie Endowment for International Peace in 1910 [Estournelles de Constant and Butler 2018, pp. 13–15]. Among them, two individuals were the main actors of the correspondence that is used as my dataset: Paul d'Estournelles de Constant and Nicholas Murray Butler.

### 3.1.2 The Main Characters

Paul d'Estournelles de Constant was a French diplomat born in 1852. He was elected to represent Mamers (Sarthe, Pays de la Loire) then La Flèche (Sarthe, Pays de la Loire) to the Chamber of Deputies, and later on, was elected to represent the Sarthe to the Senate. Throughout his time as a representative, he worked for peace and conciliation, and in recognition of his efforts, he was awarded the Nobel Peace Prize in 1909. He also facilitated the extension of the Carnegie Endowment for International Peace in France and Europe, and became president of the European bureau in 1911 [Estournelles de Constant and Butler 2018, pp. 15–16].

Similarly, Nicholas Murray Butler was a prominent figure in diplomacy and peace efforts. American diplomat and politician, he also was an educational administrator, notably as the president of Columbia University, from 1902 to 1945. He was a member of the Republican Party, where he unsuccessfully sought the Republican presidential nomination in 1916, 1920, and 1928.<sup>1</sup> He was also deeply committed to peace, in particular as the president of the American branch of the Association for International Conciliation, founded by d'Estournelles, and significantly in his commitment to the Carnegie Endowment for International Peace [Estournelles de Constant and Butler 2018, p. 15].

### 3.1.3 The Correspondence

As they shared a common goal of promoting peace and often met in diplomatic settings, a friendship developed between them after their meeting in 1902, during d'Estournelles first trip to the United States [Estournelles de Constant and Butler 2018, p. 16], for the establishment of the federal bureau of the *Alliance Française* [Barcelo and Réau 1995, p. 208]. Because they lived on different continents, their friendship evolved into an

---

1. First, for other candidates such as Elihu Root, and then for himself

exchange of letters, casual at first, then more formal and organized when World War I began [Estournelles de Constant and Butler 2018, pp. 18–19]. From this point on, Paul d'Estournelles de Constant began to systematically write and preserve his correspondence with Nicholas Murray Butler. He numbered the letters, created copies, and maintained detailed inventories of the letters he sent. The original goal of this correspondence was to relate the war to Butler, and in addition, d'Estournelles also wanted to give details of it to the Americans. The objective was to convince them to join the war effort, as presented in [Estournelles de Constant and Butler 2018]. This correspondence could have stopped with the end of the war in 1918, or even in 1919 after the signing of the Treaty of Versailles. However, their correspondence continued until 1924, the year of d'Estournelles' death. One of the last letter is dated of March 1924,<sup>2</sup> which was approximately two months before d'Estournelles' death. Although the war was over after 1919, it remained the topics of some letters, as d'Estournelles recounted the aftermath of the end of the war. Over the course of the war and afterwards, d'Estournelles wrote a total of 1,500 letters to Nicholas Murray Butler.

This correspondence was the focus of [Estournelles de Constant and Butler 2018], a book providing both sides of it. By contrast, my dataset is only made of the letters wrote by d'Estournelles. The paper version of the dataset, as well as the entire collection of the correspondence, is available at the Archives Départementales de la Sarthe<sup>3</sup> under the identifier 12J<sup>4</sup>. It is divided in several boxes, as the correspondence is pretty extensive. It was donated by the daughter of d'Estournelles in 1957, and was classified<sup>5</sup> afterwards by the two archivists that handled the collection, Henri Bouillier de Branche then Gérard Naud.<sup>6</sup>

### 3.1.4 The Project

In 2019, the DAHN (*Dispositif de soutien à l'Archivistique et aux Humanités Numériques* (Support system for Archival and Digital Humanities)) project was launched. One of its goals was to create a digital scholarly edition with Paul d'Estournelles de Constant'

---

2. Letter number 1498 from Paul d'Estournelles de Constant to Nicholas Murray Butler (March 10, 1924): <https://nakala.fr/10.34847/nkl.1a39yyz0>

3. <https://archives.sarthe.fr/>

4. <https://archives.sarthe.fr/chercher/les-inventaires/acces-aux-inventaires>

5. Etat des archives privées (séries F et J)/Fonds privés entrés à partir de 1957 (série J)/Fonds d'Estournelles de Constant (12J)

6. Both were directors of the Archives départementales de la Sarthe when they proceed to the classification.

correspondence, which marked my initial involvement in the project. A key task was obtaining a machine-readable version of the correspondence, which required transcription. From the outset, and given the size of the corpus, Automatic Text Recognition (ATR) was chosen as the most efficient method for transcription. This required the creation of ground truth of part of the corpus, a necessary step before training a model that could work effectively on the rest of it. To create ground truth, it is necessary to do manual transcription and/or correction. It ensures that the data is free from errors and avoids confusing the model during its training (for example, by providing two versions of the same character). The final model<sup>7</sup> was obtained after two or three attempts. It resulted in adding new content to the ground truth every time. The composition of the various batches of ground truth used to produce this model is described in Annex N.

### 3.1.5 Structure of the Document

As already mentioned in Introduction, this dataset is made of typescript documents. The use of a typewriter ensures that the letters have straight and regular lines, and, as previously mentioned, this regularity facilitated the segmentation process. The dataset and ground truth are exclusively in French.<sup>8</sup> Beyond the language, the dataset also possesses some distinctive traits, visual or structural, related to the correspondence. Paul d'Estournelles de Constant was consistent and meticulous in his writing. He always followed the same pattern. First, all letters were written on his official Senator stationery, meaning that the first page consistently contains a “*SÉNAT*” (Senate) letterhead. Then, the opener always starts the same way:

- a letter numbering (“*LETTRE N°XX*” (Letter noXX))
- a dateline (“*XX, le XX XX 19XX.*”)
- a title (“*XX*”)
- a salute (“*Mon cher Butler,*” (My dear Butler)).

At the bottom of this first page, an address to Nicholas Murray Butler is always written. It is typically the same: “*à Monsieur le Président N. Murray BUTLER, NEW-YORK*” (To Mister the President N. Murray BUTLER, NEW-YORK). Subsequent pages vary

---

7. <https://zenodo.org/records/10556673>

8. On rare occasions, d'Estournelles wrote letters in English, but these were not included.

in length, but typically include page numbering at the top centre. Finally, on the last page of the letter, a closing greeting is included: “*Votre affectueusement dévoué,*” (Yours affectionately devoted). It is followed by d'Estournelles' signature, often manually added. In the end, the address to Butler can, once again, be found. In some occasions, after that, a postscript could be added, written via typewriter or by hand, and an indication of annex to the letter can also be mentioned. Figure 3.1 displays a letter containing some of these elements.

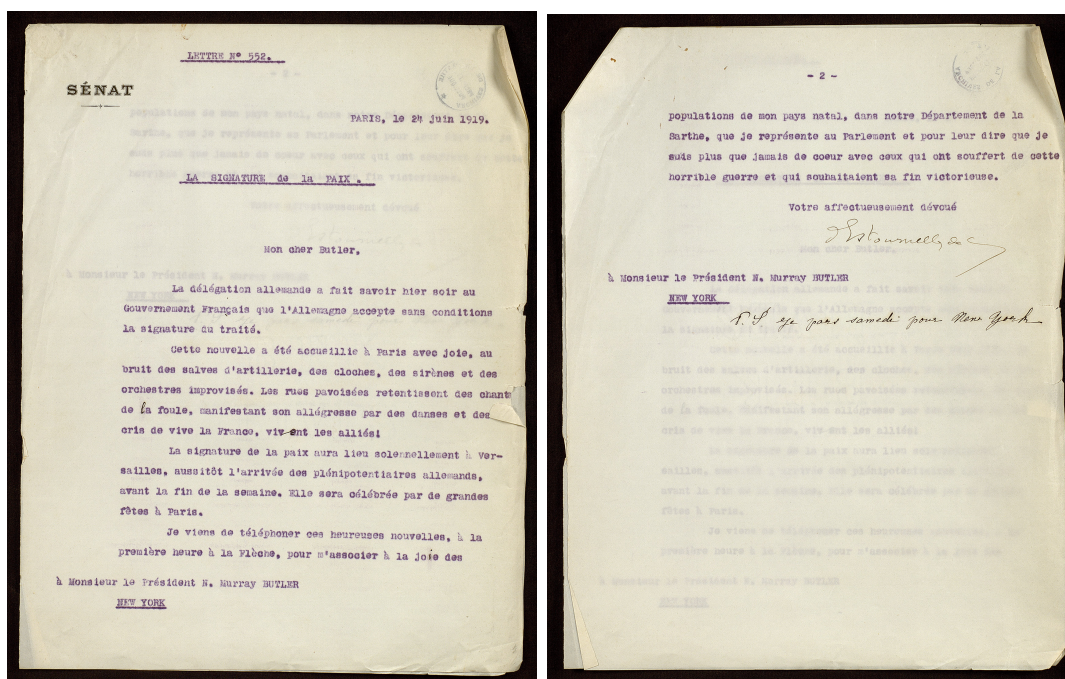


Figure 3.1: Letter no552 – June 24th, 1919

## 3.2 A Rich and Diverse Corpus

### 3.2.1 A Large Source

Paul d'Estournelles de Constant wrote his correspondence for almost ten years (July 1914 to March 1924). In addition to spanning nearly a decade, the letters vary greatly in length. While some letters are as short as two or three pages, as shown in Figure 3.1, others can be exceptionally long. For instance, [Letter 418](#), dated August 27 to September 27, 1918, is 45-pages long, or [Letter 797](#), dated March 18, 1921, is 64-pages long. There are also numerous letters ranging from 10 to 30 pages in length.



### 3.2.2 An Eclectic Collection

The extensive nature of this corpus ensures that the dataset provides a rich and varied material for analysis. The corpus also covers a wide range of topics and subjects, which will be valuable for my experiments. The variety in topics is evident from letters such as 418 and 797, which cover contrasting subjects. Letter 418 is war-oriented, and talks about the contribution of the United States during and after the war. Letter 797 is more socialite-oriented, and recounts the events that happened at a reception held by Marie Curie, where d'Estournelles was invited.

As said previously, the distinctive traits of d'Estournelles' letters include a title, there to summarize the topic(s) that he will address in the letter. A review of the titles from the 1,500 letters reveals the collection's thematic diversity. D'Estournelles had many unique roles during and after the war: state representative, politician, diplomat, father, friend, etc. The correspondence's thematic diversity covers the majority of those roles, as further illustrated by examples from the corpus. For example, he discussed family matters, like in [Letter 682](#) ("*Mademoiselle Sarah Butler*" [Miss Sarah Butler]), [Letter 753](#) ("*Fiançailles de mon fils*" [My son's engagement]) or [Letter 850](#) ("*Ma plus jeune fille premier prix du conservatoire de musique*" [My youngest daughter, first prize at the music Conservatory]). He addressed questions of justice and trials, such as in [Letter 406](#) ("*Le procès Malvy*" [The Malvy trial]) or Letters [630](#), [652](#) and [653](#) ("*Le procès Caillaux*" [The Caillaux trial]). He mentioned political topics, like in [Letter 197](#) ("*L'élection présidentielle*" [The presidential election]) or [Letter 768](#) ("*Les élections sénatoriales*" [The senatorial elections]). He frequently mentioned the *Association de Conciliation Internationale*, and its situation in numerous countries, such as in [Letter 151](#) ("*La conciliation en Espagne*" [The conciliation in Spain]), [Letter 183](#) ("*La conciliation en Chine*" [The conciliation in China]) or [Letter 815](#) ("*Le représentant de la conciliation au Pérou*" [The conciliation's representant in Peru]). He recounts, on numerous occasions, something that happened in his constituency, like in [Letter 519](#) ("*Dans la Sarthe. Pour que la guerre ne recommence pas*" [In the Sarthe. So that war does not start again]), [Letter 569](#) ("*Mon tour de la Sarthe*" [My trip of the Sarthe]) or [Letter 1398](#) ("*La périphérie du département de la Sarthe*" [The periphery of the Sarthe's department]). Finally, many letters focus on the war but address various aspects and themes. There are mentions of the battlefield in [Letter 144](#) ("*Travaux des soldats sur le front et des blessés*" [Soldiers' work on the front and the wounded]) or [Letter 225](#) ("*Ma visite au front italien*" [My trip to the Italian front]). The intervention of the United States is raised in [Letter 190](#) ("*L'effort charitable des États-Unis*" [The charitable effort

of the United States]) or [Letter 675](#) (“*L’aide financière des États-Unis*” [The financial help of the United States]). There are talks of the consequences of the war in [Letter 152](#) (“*Les orphelins de guerre*” [The war orphans]), [Letter 173](#) (“*Les bienfaits de la guerre ??*” [The benefits of war??]), [Letter 782](#) (“*L’avenir des mutilés de guerre*” [The future of the war disabled]) or [Letter 1094](#) (“*Le bilan de la guerre*” [The toll of the war]).

Therefore, this long and varied list of letters from the correspondence convinced me to consider the corpus as an adequate source for the dataset. It could also offer the diversity of lexicon needed for the experiment. This would help me to best prove or disprove my theory.

### 3.3 Choosing the Experiment’s Test Set

#### 3.3.1 A Few Elements to Consider

To analyse the lexicon, I need to create test sets with distinct vocabularies, which allows me to conduct a thorough lexicon analysis. The first task, then, is to identify the main themes around which the test sets will be organized. Those topics should be as distinct from each other as possible, to have a clear separation between the two thematics.

The final model, used for text recognition and already mentioned, was developed based on the ground truth I created. They were based on the first part of the correspondence, that includes letters from number 1 in July 1914 to number 606 in January 1920.<sup>9</sup> No specific topics were chosen when creating the ground truth, but they will most likely refers to war, as it was written during the heat of battle. The dataset I am now analysing presents a different situation: I am working with the letters written between January 1920 (606) and March 1924 (1500), long after the end of the war then, which should bring new topics to the discussion.

#### 3.3.2 Which Topics to Select?

For the test sets, I selected two subjects: “War” and “Other”. Since the war had been over for several months before the dataset period began, I focused on letters specifically dedicated to the topic, because it suggested that d’Estournelles intentionally focused on war-related topics. It would then strengthen the possibility of having a dedicated lexicon

---

9. This part of the correspondence is available at [https://discholed.huma-num.fr/exist/apps/discholed/index\\_pec.html?collection=pec/corpus](https://discholed.huma-num.fr/exist/apps/discholed/index_pec.html?collection=pec/corpus)

in these pages. In contrast, as no specific topics emerged from the remaining letters, I labelled this category as 'Other', encompassing all topics where war is not mentioned. This content can be pretty diversified, such as political affairs, family business or the situation of France after the war. This approach resulted in unequal test sets. Each set has nine letters, with about two to five pages, except on two occasions for the set Other (SO). Those letters are 17- and 38-pages long. This explains the inequality: the set Other has 76 pages, while the set War (SW) only has 31. In the tables 3.1 and 3.2, information is given on the letters chosen for each set. It mentions their number, page count, date, and topics.

### 3.3.3 Production of Models

To conduct the subsequent experiments on those test sets, I developed text recognition models, by using the specific texts mentioned priorly and the eScriptorium instance.<sup>10</sup> Three models were generated: Model Other, Model War and Model War Retrained. Model Other trained for 28 epochs,<sup>11</sup> and resulted in a model achieving 98.2% of accuracy. Model War trained for 20 epochs, and produced a model of 97.3% of accuracy. Although the percentage difference is not that big, I was concerned that the limited ground truth might not be sufficient for an effective model. To address this concern, I created Model War Retrained. This model was trained on the War dataset and fine-tuned<sup>12</sup> based on the existing Model War. The model trained for 13 epochs, and achieved 97.7% accuracy, which represents a slight improvement over the original Model War but is still below the accuracy of Model Other.

### 3.3.4 Sampling the Test Set

For the rest of the lexicon experiment, I also needed to extract a smaller, more focused test set from within the larger test set. This approach aimed to enable a detailed analysis of a sample from the dataset. From each set, five pages were chosen with much attention. I aimed to evaluate the model's performance, across various scenarios. I selected certain pages based on their unique characteristics. For example, some pages contain a high frequency of uppercase letters, which can challenge the model if it has not been trained with

---

10. See subsection 2.1.2 for details

11. An epoch is a cycle during which training, testing and validating will be done to progress into the production of an accurate model

12. Fine-tuning is a technique that adapts a pre-trained Machine Learning model for a specific task.

Document	Number of pages	Date	Topics
Letter 617	3	1920-02-04	Treaties and war actions
Letter 678	2	1920-06-03	Trials and war
Letter 844	5	1921-06-18	War memorials and ceremonies
Letter 927	4	1921-12-09	Military occupation
Letter 948	4	1921-12-20	War memorials and ceremonies
Letter 957	4	1921-12-25	Armaments of war
Letter 1000	5	1922-01-23	Summary of the war correspondence
Letter 1364	2	1923-07-30	German sentiment
Letter 1367	2	1922-08-03	Post-war Germany

Table 3.1: Composition of the set War

Document	Number of pages	Date	Topics
Letter 607	38	1920-01-12/13	Elections
Letter 722	2	1920-11-18	Colleague life
Letter 753	3	1920-12-23	Family marriage
Letter 846	3	1921-06-20	Diplomacy and theater
Letter 1029	3	1922-02-20	Carnegie project
Letter 1103	2	1922-06-12	Family relations
Letter 1170	4	1922-10-21	Religion and populations
Letter 1217	4	1922-12-14	Cost of post-war life
Letter 1358	17	1923-07-18	Work obligations

Table 3.2: Composition of the set Other

sufficient uppercase examples. Others feature narrow lines of text, potentially affecting the model's ability to recognize text accurately. Table 3.3 lists the selected pages and details the unique features of each. Those pages have been used on experiments that will be presented in this part, as well as the next one.

Document	Set	Specificities
Letter 607 Page 3	Other	None
Letter 607 Page 17	Other	Narrow lines of text
Letter 678 Page 1	War	None
Letter 722 Page 1	Other	None
Letter 844 Page 1	War	Narrow lines of text
Letter 948 Page 1	War	None
Letter 1000 Page 3	War	None
Letter 1170 Page 3	Other	None
Letter 1358 Page 4	Other	Uppercases letters
Letter 1367 Page 1	War	Narrow lines of text

Table 3.3: Composition of the test set

# KNOWING THE CONTENT BEFORE JUDGING ITS EFFECT

---

To advance the knowledge on how training works in neural networks, I first hypothesized that the lexicon is part of the production of an accurate text recognition model. However, proving or disproving this theory requires a comprehensive understanding of the dataset used. Correlating a bad recognition accuracy with a lack of the right lexicon, without knowing what the correct lexicon should be, is pointless. Therefore, this chapter covers the in-depth analysis I have done on the correspondence of Paul d’Estournelles de Constant to understand its composition and distribution. It explores what would make it a good fit, or not, for a lexicon-based model.

## 4.1 Learning More About the Topics in the Dataset

### 4.1.1 What Sources to Explore, and How, to Obtain More Knowledge?

To accompany my first experiment aiming at testing the effect of lexicon-based models, I decided to further explore the training data, to clearly understand its lexical composition. In this analysis, I work with three sets from the Paul d’Estournelles de Constant’ corpus. There is a set of texts about war, another is a set of texts about other topics, and the last one is a set of texts used as ground truth for the corpus’ model.

The composition of the first two was already presented in section 3.3, but this analysis didn’t start with their final form. During the first part of the analysis, the letters 607 and 1358 from Table 3.2, and 844 and 927 from Table 3.1, were not part of the dataset. However, as the results I obtained were not sufficient to be able to deduce anything, which I will delve into in the following subsections, I added these four new letters, two for each set, and the new texts added to the set Other are notably longer. Then, I can get more

substantial data and more conclusive results.

Those letters were chosen after a thorough reading, which allowed me to classify the letters into two distinct groups based on their themes. Analysing in-depth their lexicon would have two effects:

- I could verify my assessment, by checking if the vocabulary is really war-oriented or the opposite;
- I could observe precisely the difference, if there is any, between the two sets.

For the last set of text, the ground truth, I barely mentioned anything in the previous section, and only mentioned its chronological boundaries (July 1914-January 1920). This set is pretty consequent, as it is made of about 400 pages of images segmented and transcribed in their entirety. Additionally, 100 pages, including specific elements, such as uppercase letters and digits, were added to enhance the ground truth. All told, the ground truth contains about 12,500 lines and 475,000 characters. During the creation of the ground truth, the focus was on quantity: I transcribed a certain amount of documents, trained a model, applied it on new documents, and transcribed a bit more if the prediction was still too bad. Consequently, I have no knowledge of its thematic composition. This is why it is one of the subject of this analysis. The model developed from those ground truth works rather well on the documents of the rest of the corpus. Therefore, exploring its lexicon would allow me to check if it is due to a large quantity, a diversified vocabulary or some other reasons.

I have the three clearly established sets I want to work with, but a bit of cleaning is necessary to keep only the useful information for this analysis.

First, the texts were taken from their transcription output, which was prepared for encoding the letters. For this reason, they contain symbols used for transcription rules, such as markers for handwritten or crossed-out sections, which need to be removed.<sup>1</sup>

Secondly, the punctuation and digits present in the texts also have to be removed, as they are inconsequent in a vocabulary analysis.

In addition to vocabulary variety, I am also interested in word frequency, and I want to assess how frequently, or infrequently, they appear in the set. Therefore, it is essential that every word be in lowercase. For example, I do not want to have different occurrence's

---

1. For example, while I did the transcription for d'Estournelles dataset, I put two £ on each side between a sequence of handwritten words, and one € between a crossed out sequence of characters, words, or lines.

counts for “*Président*” and “*président*” (president).

Then, my sets need to only include valuable words, as these are the words that should provide meaningful insights into their lexicon. As a result, I decided to remove from the sets all the stop words, which are “high-frequency terms [that] carry little semantic weight” [Jurafsky and Martin 2008, p. 298]. Additionally, their commonality and high frequency could clutter the word occurrence lists. There is no exhaustive list of stop words that exists, but it is possible to easily find some extensive list on the Internet, to which I can also add some items if necessary. It was the conduct I followed by using a website that shared pre-established lists of stop words by language.<sup>2</sup> The list was subsequently added to a script that performs all the cleaning steps, detailed in Annex P.1.1.<sup>3</sup>

With my newly cleaned sets, I can now proceed to my lexicon analysis and produce the outputs I wanted.

#### 4.1.2 A Dual Style of Results

For this analysis, my outputs are twofold, with one being the direct visualisation of the other.

First, as mentioned previously, my goal was to obtain information on the composition and distribution of vocabulary within the sets. In order to do that, I decided to create frequency lists. They are defined as such:

"In computational linguistics, a frequency list is a sorted list of words together with their frequency, where frequency here usually means the number of occurrences in a given corpus, from which the rank can be derived as the position in the list."<sup>4</sup>

The frequency lists will provide three key insights into my sets: (1) I will have a list of all the words in my sets, (2) their number of occurrences will be next to it, and (3) it will rank the words from most to least frequent. The top-ranking words should indicate the dominant themes within each set. Thereby, it would verify, for the set War (SW) and set Other (SO), whether their lexicon aligns with the themes they are expected to address. For the set Ground Truth (SGT), it would enlighten me about the vocabulary variety that might appear in the set.

To generate these frequency lists, I added additional lines of codes to Script P.1.1, which

---

2. <https://countwordsfree.com/stopwords>

3. The list of stop words can be found at the line 21 of the script.

4. [https://en.wikipedia.org/wiki/Word\\_list](https://en.wikipedia.org/wiki/Word_list)



counted the occurrences of each word of the sets, compiled this data into a dictionary, ranking words by frequency from highest to lowest, and then outputted it into a CSV file.<sup>5</sup> One column is for the word, and one is for the occurrences. Therefore, the information provided are clearly observable. I generated several frequency lists for my analysis: the set Ground Truth has only one [frequency list](#), but it is not the case for the sets War and Other. As explained in the previous subsection, the analysis for those sets were made in two parts. First, I only had [43 pages](#) in the test. Then I had [107 pages](#). Consequently, each set has two batches of frequency lists, and, in each batch, there are three lists:

- One has the occurrences of the set Other;
- One has the occurrences of the set War;
- The last one has the combined occurrences of the two sets.

The data from these lists enabled the second part of the results: the visualisation. It is displayed by a word cloud and is defined as follows:

"A word cloud is a visual representation of text data [...] and the importance of each tag is shown with font size or colour."<sup>6</sup>

All the word clouds, shown in Annex J, followed the same pattern:

- The word clouds use three colours: black, blue, and red.
- The words in the cloud are oriented in multiple directions;
- The more omnipresent words are the biggest, and they pop up in the centre of the clouds.

The sets range in size from 2,600 to 38,700 words. However, this difference in set size, while apparent via the number of occurrences in the frequency lists, is not immediately noticeable in the word clouds. This is illustrated in Figures 4.1 and 4.2.

In both cases, the first ranked word is “*guerre*” (war). While appearing in roughly the same size in both word clouds, they have a large difference in occurrences: 481 occurrences in the set Ground Truth and 38 in the set War.

---

5. A CSV or Comma-Separated Value file is a text file format that uses commas to separate values, and newlines to separate records, to stores tabular data (numbers and text) in plain text, where each line of the file typically represents one data record [Shafranovich 2005].

6. [https://en.wikipedia.org/wiki/Tag\\_cloud](https://en.wikipedia.org/wiki/Tag_cloud)

To compare, for example, the word “*président*” (president) demonstrates this point. In Figure 4.2, it seems to be slightly smaller than the first ranked word, but not by much, while it is ranked 5th, with 23 occurrences in the frequency list. By contrast, in Figure 4.1, it is half the size of the top ranked word, but, in the frequency list, it is ranked 4th, and occurred 273 times. The size difference is really due to the gap between the two entries. There is only a difference of 15 times for the set War, while it is 208 in the set Ground Truth.

Understanding the impact of frequency distribution on word size is critical for the continuation of my analysis. This emphasizes the importance of drawing conclusions from both the frequency lists and the word clouds.

### 4.1.3 A Greater Understanding of the Composition of the Sets

My data provided by the frequency lists was combined with the visualisation rendered by the word clouds.

From that, I observe that the top four most frequent words, regardless of the set’s size or source, are roughly the same: “*guerre*” (war), “*butler*” (Butler), “*président*” (president), and “*paris*” (Paris). This is clearly visible in the word clouds, where these words stand out significantly. In addition, “*paix*” (peace) also appears consistently within the top ten across all sets. Similarly, the next ten most frequent words are also quite consistent across the sets.

In contrast, the word clouds display many inconsequential words, in small font at the periphery, which are less noticeable in the frequency lists where they appear simply as words with one occurrence.

In the frequency list, the major difference between the small and the big source test is from adding longer letters. It led to the disappearance, from the top of the list, of recurring words from d’Estournelles greetings: “*affectueusement*” (affectionately) and “*dévoué*” (devoted). Interestingly, while they are no longer at the top of the frequency list after adding longer letters, they still stand out in the word clouds. The same can be said for the other elements from the greetings (the address, the salute). The small source test provides limited insight, and their top occurrences are pretty low, with highest frequencies of 24 and 22. It can also be observed with the word clouds, since there are very few words that stand out, and it is the multitude of little words in the outline that are mostly striking.

Meanwhile, the large source test provides valuable insights and allows for comparisons. When comparing the set Other and set War, the bigger size of the set Other can easily be



pointed out in the frequency list: they have respectively 3,523 and 1,635 words, and both sets have some similarities. Both sets use the word “*guerre*” (war) with similar frequency. However, the distinction in lexicon content becomes clear, particularly in the word clouds, since words stand out in the middle. The set War has mostly a lexicon focused on combat and diplomacy, while the set Other’s lexicon revolves around political and societal stuff. This dominance of political terms could be explained by the larger size of the set Other, as observed in both the frequency list and word cloud.

This comparison also brings to light the need for reassessing the stop words. I used a pre-established list, but I did not consider that the recurrent elements from d’Estournelles could hinder my results. As a result, if I reprocess these sets, I would remove phrases such as “*à Monsieur le Président Nicholas Murray Butler*” (To the Sir President Nicholas Murray Butler) or “*Affectueusement dévoué*” (Affectionately devoted), since it adds no value to my lexicon analysis.

As for the ground truth, no specific lexicon transpired from both the frequency list and the word cloud. The vocabulary seems to be a mix of military and society lexicon, which seems to highlight that, although it was not deliberate, the ground truth is rather varied in quality content and its performance is rather due to its quantity.

Therefore, if the lexicon significantly affects the model’s recognition abilities, the ground truth would have to be remade more thoroughly, and I will need to carefully curate its thematic content.

This is also telling about the first part of the correspondence. It was written right at the time of the war, but lexicon about war and fighting do not seem to be the main topics. It means that d’Estournelles were already pretty varied in the subjects he addressed to Murray Butler.

## 4.2 Exploring the Distribution of the Test Sets

### 4.2.1 A Continuing Experiment, but with a Different Focus

In the previous experiment, I was interested in the general composition of each set. In this one, my focus is on the distribution of the content. I intend to find patterns or reasons that would explain afterwards why one model is working better than another, in relation to the lexicon it is made of. Therefore, this experiment focuses solely on set War and set Other, since, as previously explained, the set Ground Truth concentrates on quantity

rather than thematic content, making it less suitable for this experiment. My goal, in this part, is to determine if a model trained on lexically themed data performs differently based on the lexicon it is built from.

The frequency lists and word clouds of the lexicon analysis demonstrated that the set War and set Other have numerous words in common. Consequently, it is essential to thoroughly explore their composition and distribution. I need to ensure that aside from a few commonalities, the sets are thematically distinct. In order to do that, this experiment concentrates on both content and quantity. Again, I am interested in both the difference in size between the sets and the occurrence frequency of the same words.

As in the previous experiment, the sets first underwent a similar cleaning process:

- I removed punctuations and digits;
- I lowercased all text;
- An additional cleaning was done. The strings of texts are separated by lines, then by spaces. As a result, some words were cut off at the beginning or end of a line, and some were also wrongly written. Since I wanted to prevent this noise from complicating the experiments' results and observations, it was removed, with the help of Script P.2.5, with which I used the sets and a French dictionary.

After cleaning the texts, the next step is to focus on the core of the experiment, which involves revealing the commonalities and differences between the sets. This is why the stop words have been kept for this experiment. In this experiment, I aim to assess both the intensity of their presence and their distribution across the sets. I am not interested in the sets as individual whole entities, but as two elements for a comparison, and I want to understand them more. This will allow me to better understand the results of the model predictions in the upcoming comparative analysis.

To properly conduct this experiment, three Python<sup>7</sup> components—one function and two methods—were essential in my scripts.

The function `counting()`, detailed in Annex P, creates a dictionary of word occurrences from a text. The input text, originally a string of characters, is first split into a list of words based on spaces. The function, then, iterates through each item in the list, adds it to a dictionary, and counts its occurrences.

The two methods, `difference()` and `intersection()`, are used in Script P.2.2. The

---

7. Python is the primary programming language used in this PhD

methods take, at minimum, two lists of items as arguments, and they find respectively unique and common elements of each list.

Since these methods require lists with unique elements as arguments, I also applied the method `set()` to it, converting them into ensembles with no duplicate. Concretely, I have two given ensembles A and B made of single items:

- `A.difference(B)` returns an ensemble containing all items from set A, except those found in both A and B.
- `B.difference(A)` returns an ensemble containing all items from set B, except those found in both A and B.
- `A.intersection(B)` returns an ensemble with the items found in both A and B.

Ultimately, those three ensembles will have absolutely no item in common.

Then, I used the dictionaries of occurrences and the ensembles of unique items jointly. I obtained dictionaries of occurrences for the unique and common items between the set War and set Other, and I can also visualize them with word clouds, as I did before.

## 4.2.2 The Test Sets' Content in Its Many Shapes

This experiment aims to gain a deeper understanding of the contents of my sets. To adequately ensure that, I decided to explore it in various forms.

First, the ensembles, lists, and dictionaries I mentioned in the previous subsection were acquired in two kinds: tokens and lemmas.

The tokenization, defined as “the task of segmenting running text into words” [Jurafsky and Martin 2008, p. 19], was used in the previous lexicon analysis, as tokens are the standard form derived directly from the text.

The lemmatization, which involves “determining that two words have the same root, despite their surface differences” [Jurafsky and Martin 2008, p. 24], will provide additional insights. Its execution can be defined as such:

"The most sophisticated methods for lemmatization involve complete morphological parsing of the word. Morphology is the study of the way words are built up from smaller meaning-bearing units called morphemes. Two broad classes of morphemes can be distinguished: stems—the central morpheme of the word, supplying the main meaning—and affixes—adding “additional” meanings of various kinds". [Jurafsky and Martin 2008, p. 24]

This experiment focuses on the content of each set, particularly the unique lexicon of each. Analysing tokens will provide useful information, but examining lemmas will reveal deeper insights into the core vocabulary.

Lemmatization reduces the content to its basic morphemes.<sup>8</sup> Since the sets are in French, tokenization to lemmatization involves significant changes due to grammatical aspects of the language.

This language involves genders and plurals in all kind of grammatical categories (noun, adjective, determinant). For example, the following token list (“*le*”, “*la*”, “*les*”, “*tout*”, “*toute*”, “*toutes*”, “*tous*”, “*grand*”, “*grande*”, “*grandes*”, “*guerre*”, “*guerres*”)<sup>9</sup> is lemmatized to (“*le*”, “*tout*”, “*grand*”, “*guerre*”).<sup>10</sup>

The French language also encompasses various and numerous conjugations according to the pronoun and time. Therefore, the following token list (“*vais*”, “*irai*”, “*allaient*”, “*allez*”, “*vont*”)<sup>11</sup> is lemmatized to only (“*aller*”).<sup>12</sup>

To obtain those two forms, I used *spaCy*, a Python library for Natural Language Processing (NLP), made of linguistic data and algorithms that are able to process natural language texts. It functions with pretrained models available in many languages among which English, German, Italian, Dutch [Vasiliev 2020, p. xvi].

Amidst the operations proposed by *spaCy* are the tokenization and lemmatization. After loading a language model, it will parse the text put as an argument, and will output it, either in the token or the lemma form [Vasiliev 2020, pp. 18–19]. The *spaCy* library and the lines of code to obtain the tokens and the lemmas were added in Script P.2.4, and it intervenes after the cleaning of the sets.

The results acquired were then put through the `counting()` function. A Python dictionary *output*, of each set’s tokens and lemmas and their occurrences, was produced, and I then generated word clouds for the six lists created:

- common tokens;
- tokens unique to the set War;
- tokens unique to the set Other;
- common lemmas;
- lemmas unique to the set War;
- lemmas unique to the set Other.

---

8. A morpheme is “the smallest unit of meaning in a word.” [Denham and Lobeck 2013, p. 143]

9. Translation: the, the, the, all, all, all, all, big, big, big, war, wars

10. Translation: the, all, big, war

11. Translation: goes, will go, went, went, go

12. Translation: go

They are available in Annex K, and in Annex A with Tables A.1 and A.3. They render with numbers the distribution of the sets before and after Script P.2.2, which produced the differences and commonalities. An example of both sets' common elements, in their tokens and lemmas forms, is delivered with Figure 4.3.

Following that, I decided to further explore my sets by doing part-of-speech tagging. The part-of-speech designates the nature of the word in a given sequence, and can help understand sentence structure and meaning [Jurafsky and Martin 2008, p. 162]. Part-of-speech tagging, which involves “assigning a part-of-speech to each word in a text” [Jurafsky and Martin 2008, p. 165], is also one of the operations done by the library spaCy [Vasiliev 2020, pp. 21–22]. As the main element used in Script P.2.3, it provides the part-of-speech category for the [ten lists](#) created from the sets:

- common tokens;
- tokens of the set War;
- tokens of the set Other
- tokens unique to the set War;
- tokens unique to the set Other;
- common lemmas;
- lemmas of the set War;
- lemmas of the set Other;
- lemmas unique to the set War;
- lemmas unique to the set Other.

According to [Jurafsky and Martin 2008], there are seventeen part-of-speech classes, thoroughly detailed and explained in pages 163 to 165. My analysis only included fourteen of them, and they are described in Table 4.2, which is a slightly modified version of the table in [Jurafsky and Martin 2008, p. 163].

The distribution of the part-of-speech classes in set War and set Other are available in two ways:

- First, I created Tables A.5 to A.11, included in Annex A. The columns rendered either numbers from the sets as whole, or with their unique and common elements, and the rows are the classes in alphabetical order;
- Then, I generated pie charts, shown in Annex L. It displayed the classes' distribution, with both forms next to each other each time. A different colour is used for each slice, and the class label and its percentage are provided as well, as illustrated in Figure 4.4.





Tag	Description	Example
ADJ	Adjective: noun modifiers describing properties	red, young, awesome
ADP	Adposition (Preposition/Postposition): marks a noun's spatial, temporal, or other relation	in, on, by, under
ADV	Adverb: verb modifiers of time, place, manner	very, slowly, home, yesterday
AUX	Auxiliary: helping verb marking tense, aspect, mood, etc.	can, may, should, are
CCONJ	Coordinating Conjunction: joins two phrases/clauses	and, or, but
DET	Determiner: marks noun phrase properties	a, an, the, this
NOUN	Words for persons, places, things, etc.	algorithm, cat, mango, beauty
NUM	Numeral	one, two, 2026, 11:00, hundred
PRON	Pronoun: a shorthand for referring to an entity or event	she, who, I, others
PROPN	Proper noun: name of a person, organization, place, etc.	Regina, IBM, Colorado
PUNCT	Punctuation	; , ()
SCONJ	Subordinating Conjunction: joins a main clause with a subordinate clause such as a sentential complement	whether, because
VERB	Words for actions and processes	draw, provide, go
X	Other	asdf, qwfg

Table 4.2: 14 part-of-speech classes of the sets

The combination of tokens and lemmas forms, words and part-of-speech classes, and common and unique elements, should provide plenty of data for me to draw some conclusions about the sets, and could help me for the subsequent comparative analysis.

### 4.2.3 What Does the Content Tell Us?

An observation of the content by their numbers attests clearly that the two sets have a big size difference. Whether it is in tokens or lemmas, the set Other counts the double of items of the set War.

That difference remains noticeable with the separation between common and unique tokens. The common elements represent about half of the set War, but only a quarter of the set Other.

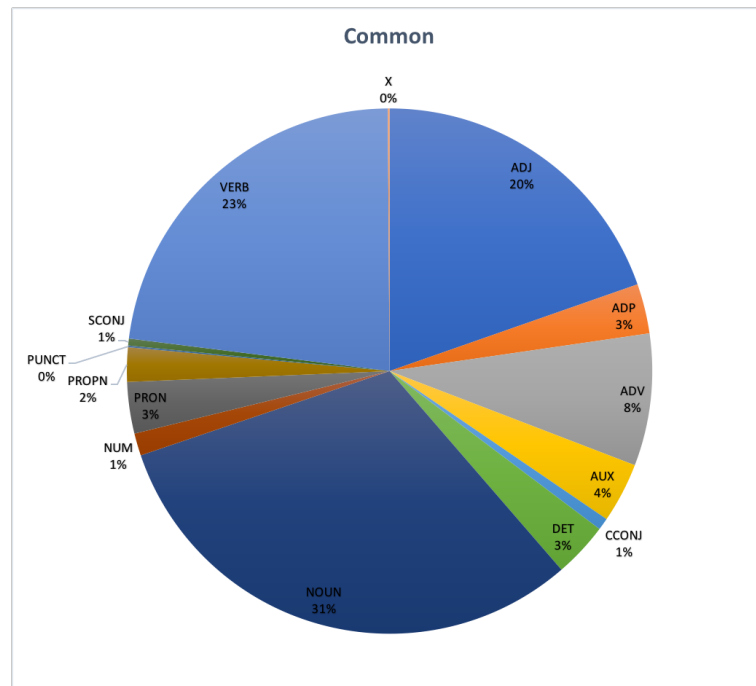
The difference in size is also pretty apparent when the tokens are transformed into lemmas. The set War loses about 20% of its content. It is 25% for the set Other. As for the common items, the diminution is only of 5%, which could be explained by the disappearance of plurals and conjugation.

It can be proven with the observations of the pie charts and word clouds. The common items are mostly made of conjunction, adverbs, determinants, and auxiliaries. The last two are the elements that draw the most attention in the word clouds. The other main elements of this list is, once again, the greetings' formula from d'Estournelles. They can be seen popping out in the word cloud.

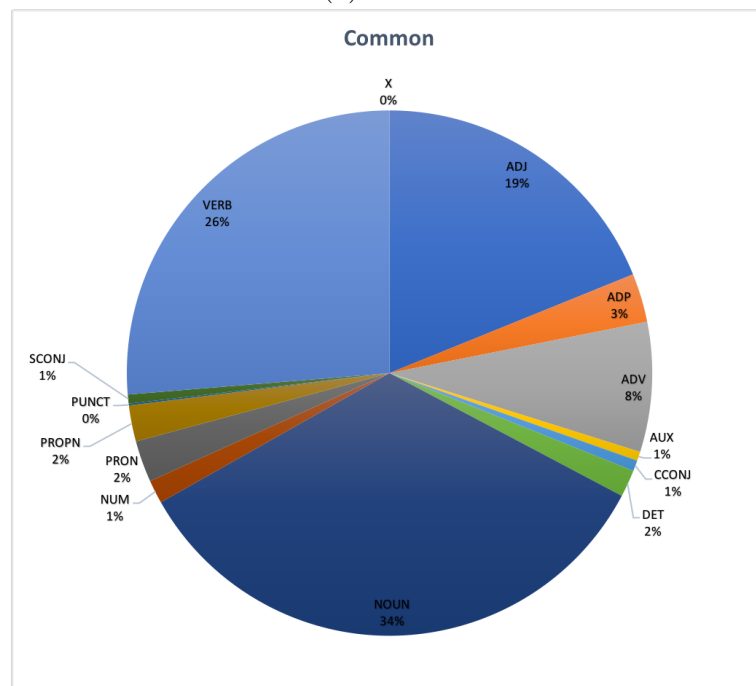
While focusing only on the pie charts of the part-of-speech tagging, the division from whole sets to common and unique elements bring very different results.

The full sets are rather similar in their division. There is usually only one point of difference, but the same can't be said with the partitioned sets.

First, when observing the split of the tokens, the unique elements from the set Other contain way more adjectives and verbs than the other two, 32 and 27%. The unique elements from the set War have 29 and 22%. The common elements have 23 and 20%. In contrast, it encloses fewer nouns (29%), compared to the other two partitioned sets (31% for both). Then, focusing on the division of the lemmas, new numbers appear. The distribution of the verbs and nouns is rather equivalent to that of the tokens. There are 26% for the unique elements in the set Other, and 34-35% for the other two. The presence of adjectives have lowered with the lemmas. There are 19% for the common elements, and 22 and 25% for the unique elements of the sets Other and War, respectively. The numbers are not always very high, but those distributions of the content allow me to hypothesise that



(a) Tokens



(b) Lemmas

Figure 4.4: Part-of-speech division of common elements between the set Other and set War

the sets Other and War contain much separate but consistent vocabularies. It is enough to verify if the lexicon is indeed involved in the recognition's abilities of the model.

Finally, when concentrating on the word clouds of the partitioned sets, the scale of results is the most noticeable element. The common elements are mostly made of items in huge fonts in the centre. The unique elements of the set Other has some large words, but of about half the size of the common elements, and they appear to be surrounded by numerous unique terms. The unique elements of the set War, in contrast, contain very few prominent words, which are barely noticeable among the numerous unique terms. Although the word clouds for the unique elements are less discernible than those of the common elements, they confirm the findings from the earlier lexicon analysis. Their vocabularies are clearly focused on the respective themes for which they were chosen:

- The set War is made of many military terms, including “*désarmement*” (disarmament), “superdreadnought”, “*colonie*” (colony), “*amiral*” (admiral), “*marine*” (navy), and “*capitaine*” (captain);
- The set Other primarily contains political lexicon, with terms such as “*chambre*” (chamber), “*républicain*” (republican), “*assemblée*” (assembly), “*électeur*” (constituent), and “*canton*” (township).

This supports the hypothesis formed after observing the pie charts. This confirms that the two sets are distinct enough to develop lexicon-based models, and will help me verify my hypothesis on the impact of the lexicon in the recognition.

# ESTIMATING THE EFFECT OF LEXICON-BASED GENERATED MODELS

---

Determining the reasons behind the efficiency of a text recognition model is not an easy task. It relies on many elements and data, and notably metrics, i.e. a the measure of distances between elements. Those metrics are numerous and focused on specific elements from a source. They require advance calculations, meticulous inquiries and precise observations. They also involve fully understanding how to interpret them compared to the model. Fortunately, progress were made on providing data and results on the efficiency of a model through the use of metrics, while the research field of text recognition was progressing. In this chapter, I am exposing the extent of the metrics used in Automatic Text Recognition (ATR) and the available tools to calculate it. Then, the results obtained about my lexicon-based models will be detailed, as well as the reasons why the lexicon was, in fact, not a viable source for the recognition.

## 5.1 How to Evaluate the Accuracy of a Model

### 5.1.1 A Variety of Metrics

To determine how well a model is doing, comparison and error analysis are the key elements. The comparison is done between two types of documents: a ground truth, i.e., the matching textual transcription of an image, and a prediction, i.e., the result of the application of a text recognition model on an image. In this case, the ground truth and the prediction are produced from the same image. The comparison is then done between the text the model should have outputted and the prediction it actually outputted. Then, an analysis can be conducted on the errors that occurred during the prediction. They designate the occurrences where the model encountered difficulties. From this, it is possible to gain knowledge on the recognition abilities of the model. More accurately, it informs

about the recognition inabilities of the model, knowledge that can be subsequently used to improve it.

Evaluating the performance of a model is possible with the use of the Levenshtein distance, named after Vladimir Levenshtein. In 1965, he defined the metric, which takes into account “the minimum edit distance between two strings, which is defined as the minimum number of editing operations needed to transform one string into another” [Jurafsky and Martin 2008, p. 25]. It produces “the simplest weighting factor in which each of the three operations has a cost of 1”, with the assumption “that the substitution of a letter for itself, for example, t for t, has zero cost” [Jurafsky and Martin 2008, p. 26].

The Levenshtein distance takes into account three elements, illustrated in Figure 5.1:

- Insertion, i.e., the addition of a character or sign where it should not exist;
- Deletion, i.e., the removal of a character or sign where it should appear;
- Substitution, i.e., the change of a character or sign by another at the exact same place in the content.

A substitution involves changing one character for another, while insertions and deletions are distinct operations. This distinction is important because every insertion, deletion, or substitution count for 1 in the Levenshtein distance, as shown in Figure 5.2. Therefore, it is essential to properly differentiate each case.

From what the Levenshtein distance entails, obtaining metrics more telling than the mere number is possible. Error rate is the most used metric, because it gives information on the amount of errors made by the model applied, on various levels. The main metrics used in ATR are Character Error Rate (CER) and Word Error Rate (WER).

The WER, or Word Error Rate, is directly related to the Word Accuracy (Wacc). They are two sides of the same coin, since their sum equals 1, as seen in Figure 5.3. This metric is “based on how much the word string returned by the recognizer (the hypothesized word string) differs from a reference transcription” [Jurafsky and Martin 2008, p. 352].

In a lower level, there is the Character Error Rate or CER. Its calculation is the same as the WER, except that “word” is replaced by “character”. The term “character”, here, has to be considered as “a letter, number, or other mark or sign used in writing or printing, or the space one of these takes”.<sup>1</sup> It implies that the equation’s numbers are way different than they were with the WER equation.

---

1. <https://dictionary.cambridge.org/dictionary/english/character>

**Example of an insertion: “beauty” → “beaurty**  
**Example of a deletion: “beauty” → “beaty”**  
**Example of a substitution: “beauty” → “bearty”**

Figure 5.1: Examples of Levenshtein distance of 1

**I + D + S = 3 : “extraordinary” → “ektraodinnary”**  
**I + S + S + D = 4 : “extraordinary” → “ekxtraonbinay”**  
**I + D + D + S + S = 5 : “extraordinary” → “exaordimorys”**

Figure 5.2: Examples of insertions (I), deletions (D), substitutions (S) and their Levenshtein distance

$$WER = \frac{\text{Word substitution}(s) + \text{Word deletion}(s) + \text{Word insertion}(s)}{\text{Number of words in the reference}}$$

$$WER = = \frac{\text{Word substitution}(s) + \text{Word deletion}(s) + \text{Word insertion}(s)}{\text{Word substitution}(s) + \text{Word deletion}(s) + \text{Correct word}(s)}$$

$$Wacc = 1 - WER$$

Figure 5.3: Equations for the WER and Wacc

GT = “The big **equation** is **pretty complicated**, for my **simple brain**.”  
 P = “The big **equations** is **prety connplicated**, for my **simplebrain?**”  
 $WER = (4S + 1D) \div 10 = 5/10 = 50\%$   
 $CER = (2S + 2D + 2I) \div 60 = 6/60 = 10\%$

Figure 5.4: Example of a WER and CER calculation (GT = Ground Truth; P = Prediction)



Studying those metrics and their difference with one another offers the advantages to verify if the model is terrible at recognizing or if it only has minor difficulties. Figure 5.4 exemplifies that perfectly. The high WER (50%) suggest a terrible recognition. However, the CER at only 10% prove that the error are few, but distributed in half the words of the example.

According to the kind of research done and results sought, they are also other metrics that can be used:

- The Sentence Error Recognition (SER) measures “the percentage of sentences with at least one word error”[Jurafsky and Martin 2008, p. 353]
- The Match Error Recognition (MER) measure the “probability of a given match being incorrect”[Morris, Maier, and Green 2004].

However, they are mostly used in Automatic Speech Recognition (ASR), a task aiming at mapping any waveform into its appropriate string of characters. It is used in many domains, such as communicating with smart home appliances, personal assistants, or cellphones, producing general transcription for audio or video text, or helping in the interactions between computers and humans with some disability [Jurafsky and Martin 2008, pp. 3337–338]. ASR offers more thorough metrics, nonetheless not as useful for ATR. The Hamming distance can also be used, in the exceptional case where the two strings (GT and P) are of the same length. When it happens, the Hamming distance “between two binary vectors is the number of coordinates in which the two vectors differ” [MacKay 2002, p. 206]. For the example in Figure 5.4:

- With the addition of an *s* at “equation” in the prediction, the position of each character changes afterwards, until the omission of the second *t* at “pretty”;
- The count’s difference, here, is 8;
- Then, with the change of *m* into *nn* in “complicated”, the positions are again modified, until the space’s omission in “simple brain”;
- At this point, there are 25 different positions;
- Finally, with the last character changed from a dot to a question mark, there is one more different position;

- In total, there are 34 different positions ( $25 + 8 + 1$ ), which makes the Hamming distance.

Despite its interesting nature, the condition to obtain this metric – having the exact same length between the ground truth and the prediction – is too rarely happening to consider it to evaluate the accuracy of the model.

### 5.1.2 Open Source Tools for Evaluation

For the examples provided with Figure 5.4, the Levenshtein distance and the error rates were metrics not very difficult to calculate, as the strings were small, with only ten words and sixty characters. However, when we evaluate on a text with 2000 characters or more, doing it by hand becomes extremely complicated, because manual evaluation can be both time-consuming and prone to miscalculation.

For this purpose, tools for the evaluation have been created, and there are cases where the evaluation has been directly implemented into the software for ATR. It is the case for Transkribus and Kraken.<sup>2</sup>

As seen previously, Transkribus offers the possibility to “evaluate the performance of your models”, thanks to “a number of tools that help you understand the error rates of your models and how it compares to other models, so you can find the right model for your documents”.<sup>3</sup>

As for Kraken, the evaluation is done with the Command-Line Interface (CLI), i.e. the software mechanism used to interact via lines of code with the computer. It uses the parameter 'ketos',<sup>4</sup> with the option test and not train. All that is needed is to provide a model and evaluation files and to execute the command. It produces a report that “contains character accuracy measured per script and a detailed list of confusions”.<sup>5</sup>

There are also various tools that work on evaluating the accuracy of models. Notwithstanding, they have some features specific to a project, which explains why several exist. For example, there is dinglehopper,<sup>6</sup> created by the Qurator project for “transparent GT-based evaluation of OCR quality using CER/WER”.

---

2. Those two software have been introduced and detailed in subsection 2.1.2

3. <https://www.transkribus.org/ai-training>

4. ketos is a parameter in Kraken used to train model. See <https://kraken.re/main/training.html#dataset-compilation>

5. <https://kraken.re/main/ketos.html#recognition-testing>

6. <https://github.com/qurator-spk/dinglehopper>

ocrevalUAtion,<sup>7</sup> was created by the IMPACT project, “for comparison between GT and OCR results as well as between different OCR results in most common formats” [Neudecker et al. 2021].

More recently, in 2023, a pretty complete tool, CERberus,<sup>8</sup> “guardian against character errors” [Haverals 2023], has been created. It can be run in a web browser, and it offers numerous features, which, ultimately, give many statistics (character, block, confusion) [Haverals 2023].

### 5.1.3 My Choice: Kami App

CERberus is a useful, heavily features and easy-to-use tool. However, by the time, it was out, I was already working with an effective tool that generated all the information I needed: KaMI app.<sup>9</sup>

KaMI stands for “Kraken as Model Inspector”. It was developed by Lucas Terriel from 2020 [Terriel 2021]. My choice was made for a similar reason as the one that made me use eScriptorium as my HTR tool. KaMI was elaborated at ALMANaCH by one of my colleagues at the time. I was already working on ATR at the time, and I also served as a beta-tester for the tool while it was in development. As the various features proposed by the tool completely satisfied the needs I had for it, I continued to use it for the multiple comparative analysis that I did afterwards.

Before starting the comparison and after inputting the reference and the prediction, it is possible to choose some preprocessing options. This means deciding to ignore some elements of the text, according to what we want to observe in terms of accuracy. It is possible to ignore:

- the digits;
- the text case, i.e., the tool will not point out an error if an uppercase has been recognized as a lowercase;
- the punctuation;
- the diacritical signs.<sup>10</sup>

---

7. <https://github.com/impactcentre/ocrevalUAtion>

8. <https://github.com/WHaverals/CERberus>

9. <https://github.com/KaMI-tools-project/KaMI-app>

10. Mark above, through, or below letters used in many orthographies to remedy the shortcomings of the ordinary Latin alphabet. For more details, see subsection 8.2.3 and 10.1.1

This is useful, notably if some weaknesses of the model are already known, but aren't relevant in the case of what is to be observed with this comparative analysis. For example, deciding to ignore the digits when working with a model that had surely not been training on digits, but only on characters, is the best option.

It is also possible to combine some options. For example, stripping the text of its digits, diacritics, punctuation and upper case is possible, leaving the text bare. It gives the opportunity to see how the model did with only the characters. In case one or several options are chosen, the metrics will give the result of each case:

- default, i.e. the evaluation of the prediction as it was submitted;
- ignore 1, i.e. the evaluation of the prediction minus the evaluation of the option 1 selected;
- ignore 2, i.e. the evaluation of the prediction minus the evaluation of the option 2 selected;
- ignore combined; i.e. evaluation of the prediction minus the combinations of the ignored options

KaMI offers the metrics I have mentioned earlier, and even some used in Automatic Speech Recognition (ASR). The Levenshtein distance is given at two levels: the characters and the words.

Besides the metrics, the table of statistics provides the numbers of hits, insertions, deletions, and substitutions.

Lastly, it renders the total of characters in the reference and in the prediction, and this whole table is also downloadable in a CSV format.

Moreover, in addition to the metrics, KaMI offers a visualisation of the differences, with a “versus text”.

It presents a side-by-side display of the reference, the comparison, and the prediction, with colours in the comparison: green for exact match, blue for insertion, and red for deletion and substitution.

Excluding the metrics used in ASR (Match Error Recognition, Character Information Lost, Character Information Preserve) and the Hamming distance, all the metrics, parameters, and features have been used during the various comparative analysis that I did and will present in subsequent chapters.

## 5.2 Comparing Two Different Lexicon-Based Models

### 5.2.1 How Was the Comparative Analysis Conducted?

After having thoroughly observed the composition of the set War (SW) and set Other (SO), I applied the produced models and observed the obtained predictions. I wanted to see if the models were efficient regarding their topics.

For this comparative analysis, three models were used: model War (MW), model Other (MO), and model War Retrained (MWR).<sup>11</sup> At first, I only intended to conduct the experiment with the first two models, each trained with the set of the same name. However, the small size of the set War and the first results I obtained convinced me to try something else.

I created a new model, fine-tuned from the already existing model War, and trained once again with the data from the set War, with the objective to double the input of the model, to have it at an equivalent size to the model Other. However, doing that includes the major risk of overfitting,<sup>12</sup> because, while the model might have learned to recognize some elements more accurately, it is also possible that it just memorized the training data.

Evaluating the performance of those models is possible through the observations of the data produced from a two-steps experiment.

First, I am doing a comparative analysis on the result of the model's application on the set as a whole.

Then, as presented in subsection 3.3.4, I gathered some specific pages from each set, with their own peculiarities. They allowed me to effectively test the models, and observe the results page by page. Since I used eScriptorium to create my training data, as well as to train my models, the two sets were already available on the interface. I only needed to apply the three models to the whole set to obtain my predictions.

For the subsequent steps of the experiment, I used KaMI. In addition to the application presented in the subsection 5.1.3, it can be used with its `library` version, in the CLI. The application limits the number of characters that can be inputted as reference/prediction, with a cap at 7,000 characters. Exploiting the library allows me to not be bounded to this limit, so that I can generate metrics for the recognition of the models for each set. Indeed, they are made of about 39,000 (set War) and 100,000 characters (set Other), which is significantly above the application's cap. Table 5.1 renders the outputted metrics.

---

11. They have been previously presented in the subsection 3.3.3

12. See section 1.2 for explanation and details about it

	SW/MW	SW/MO	SO/MO	SO/MW	SO/MWR	SW/MWR
Levenshtein distance (char)	372	770	451	4090	3768	197
Levenshtein distance (words)	322	540	346	2734	2512	174
Word Error Rate (WER in %)	4,92	8,25	2,08	16,48	15.15	2.66
Char. Error Rate (CER in %)	0,95	1,97	0,45	4,08	3.76	0.51
Word Accuracy (Wacc in %)	95,08	91,74	97,91	83,51	84.85	97.34
Hits	38658	38299	99797	96425	96697	38819
Substitutions	279	557	274	3301	3031	144
Deletions	67	148	118	463	461	41
Insertions	26	65	59	327	276	12
Length (reference)	39004	39004	100189	100189	100189	39004
Length (prediction)	38963	38921	100130	100053	100004	38975

Table 5.1: Metrics for the models applied to the sets  
(S = Set; M = Model; O = Other; W = War; R = Retrained)

Next, I used the Graphical User Interface (GUI) version of KaMI to obtain results page by page, as each page has less than 7,000 characters. They are available in Tables 5.2 to 5.7. They are divided in two, then three:

- First, there are tables for the pages from the set Other and for the pages from the set War;
- Then, there is a table for the application of the model Other, the model War, and the model War Retrained.<sup>13</sup>

<sup>13</sup>. For a different view of the metrics, Annex B renders them page by page, with the results of the three models in the same table for each page.

	607-3	607-17	722-1	1170-3	1358-4
Levenshtein Distance (Char.)	5	10	9	11	7
Levenshtein Distance (Words)	2	8	6	7	6
Word Error Rate (WER in %)	0.803	1.941	3.947	2.661	8.955
Char. Error Rate (CER in %)	0.316	0.413	1.032	0.684	1.369
Word Accuracy (Wacc in %)	99.196	98.058	96.052	97.338	91.044
Hits	1576	2414	866	1597	504
Substitutions	0	3	5	7	4
Deletions	4	2	1	3	3
Insertions	1	5	3	1	0
Total char. in reference	1580	2419	872	1607	511
Total char. in prediction	1577	2422	874	1605	508

Table 5.2: Metrics for the model Other applied to the set Other

	678-1	844-1	948-1	1000-3	1367-1
Levenshtein Distance (Char.)	33	40	59	6	50
Levenshtein Distance (Words)	23	32	35	4	38
Word Error Rate (WER in %)	13.294	10.774	19.125	1.403	13.818
Char. Error Rate (CER in %)	3.116	2.312	5.296	0.357	3.086
Word Accuracy (Wacc in %)	86.705	89.225	80.874	98.596	86.181
Hits	1028	1700	1059	1671	1574
Substitutions	24	25	45	6	41
Deletions	7	5	10	0	5
Insertions	2	10	4	0	4
Total char. in reference	1059	1730	1114	1677	1620
Total char. in prediction	1054	1735	1108	1677	1619

Table 5.3: Metrics for the model Other applied to the set War

	607-3	607-17	722-1	1170-3	1358-4
Levenshtein Distance (Char.)	25	90	65	56	127
Levenshtein Distance (Words)	22	71	37	44	50
Word Error Rate (WER in %)	8.835	17.233	24.342	16.73	74.626
Char. Error Rate (CER in %)	1.582	3.72	7.454	3.484	24.853
Word Accuracy (Wacc in %)	91.164	82.766	75.657	83.269	25.373
Hits	1557	2334	811	1553	386
Substitutions	18	66	61	43	116
Deletions	5	19	0	11	9
Insertions	2	5	4	2	2
Total char. in reference	1580	2419	872	1607	511
Total char. in prediction	1577	2405	876	1598	504

Table 5.4: Metrics for the model War applied to the set Other

	678-1	844-1	948-1	1000-3	1367-1
Levenshtein Distance (Char.)	11	21	27	8	19
Levenshtein Distance (Words)	9	20	20	8	18
Word Error Rate (WER in %)	5.202	6.734	10.928	2.807	6.545
Char. Error Rate (CER in %)	1.038	1.213	2.423	0.477	1.172
Word Accuracy (Wacc in %)	94.797	93.265	89.071	97.192	93.454
Hits	1048	1709	1089	1669	1603
Substitutions	10	17	16	7	14
Deletions	1	4	9	1	3
Insertions	0	0	2	0	2
Total char. in reference	1059	1730	1114	1677	1620
Total char. in prediction	1058	1726	1107	1676	1619

Table 5.5: Metrics for the model War applied to the set War



	607-3	607-17	722-1	1170-3	1358-4
Levenshtein Distance (Char.)	19	84	61	55	121
Levenshtein Distance (Words)	16	62	37	46	43
Word Error Rate (WER in %)	6.451	15.012	24.342	17.49	64.179
Char. Error Rate (CER in %)	1.204	3.466	6.995	3.422	23.679
Word Accuracy (Wacc in %)	93.548	84.987	75.657	82.509	35.82
Hits	1558	2344	814	1553	393
Substitutions	18	63	54	43	109
Deletions	1	16	4	11	9
Insertions	0	5	3	1	3
Total char. in reference	1577	2423	872	1607	511
Total char. in prediction	1576	2412	871	1597	505

Table 5.6: Metrics for the model War Retrained applied to the set Other

	678-1	844-1	948-1	1000-3	1367-1
Levenshtein Distance (Char.)	8	6	9	9	13
Levenshtein Distance (Words)	7	7	8	8	13
Word Error Rate (WER in %)	4.046	2.356	4.371	2.807	4.727
Char. Error Rate (CER in %)	0.755	0.346	0.807	0.536	0.802
Word Accuracy (Wacc in %)	95.953	97.643	95.628	97.192	95.272
Hits	1051	1724	1105	1668	1607
Substitutions	7	4	7	9	11
Deletions	1	2	2	0	2
Insertions	0	0	0	0	0
Total char. in reference	1059	1730	1114	1677	1620
Total char. in prediction	1058	1728	1112	1677	1618

Table 5.7: Metrics for the model War Retrained applied to the set War

### 5.2.2 WER, CER, Insertions, Deletions, Substitutions, “Versus Text”: a Plethora of Statistics and Errors Comparisons at hand

Not all metrics proposed by the tool have been used. The various metrics related to Automatic Speech Recognition were removed, because I considered they would not provide any valuable information for this experiment.

For instance, the Hamming distance was removed, because except for page 1000-3 in Tables 5.3 and 5.7, the total number of characters between the reference and the prediction always differed.

Additionally, none of the “ignore” options (digits, punctuations, text case) of KaMI App has been selected, since, for this comparative analysis, knowing how well or bad it does for this specific elements of text is imperative. The French language is a language of diacritics,<sup>14</sup> and the model, no matter which set produced it, should be trained to recognize most diacritics. A similar observation can be made about punctuation in French, so it should be trained for that too. Moreover, some pages were chosen specifically for this. The page 1358-4 is made of a list of people’s names, with their last name entirely written in uppercases, to test its recognition’s skills on it. I also picked, in a few cases, the first page of the letter. It would inevitably include digits and uppercases, with the regular structure of letter numbering, date, and title, of Paul d’Estournelles de Constant’ correspondence.

Lastly, while the metrics provide valuable information, a visualisation of the errors is essential, in order to really understand where the model had problems, notably for the specific pages. Consequently, each time I produced the metrics, I also observed and studied the “versus text”,<sup>15</sup> whether it was for the images of the set Other or of the set War. The goal is to see clearly the differences between reference and prediction. An example can be observed in Figure 5.5.

It allows me to determine, after the first conclusions from the metrics, if the errors come from the lexicon, or something else.

---

14. For an extensive definition of diacritics, see subsections 8.2.3 and 10.1.1.

15. Display of the insertions, deletions, and substitutions between reference and prediction

<p>- 4 - M.M. Président Nicholas Murray BUTLER Sénateur d'ESTOURNELLES de CONSTANT Baron ADELWARD Recteur Paul APPELL Professeur J.T. SHOTWELL SANGRO de OLANO Professeur J. REDLICH Hellmuth von GERLACH Professeur FOERSTER Professeur Charles GIDE EFREMOFF Député Justin GODART Sénateur LA FONTAINE Professeur Henri LICHTENBERGER LEJEUNE, Représentant de M. Albert THOMAS Professeur NIPPOLD Pierre JAUDON Colonel CONVERSE Paul d'ESTOURNELLES de CONSTANT Professeur Th. RUYSSSEN Professeur J.-J. PRUDHOMMEAUX DANDIEU</p>	<p>- 4 - M.M. Président Nicholas Murray BUTLER Sénateur d'ESTOURNELLES de CONSTANT Baron ADELWARD Recteur Paul APPELL Professeur J.T. SuHOTVWELL SANGRO de OLANO Professeur J. REDLICH Hellmuth von GERLACH Professeur FOERSTER Professeur Charles GIDE EFREMOFF Député Justin GODART Sénateur LA FONTATINE Professeur Henri LICHTENBERGER LEJEUNE, Représentant de M. Albert THOMAS Professeur NIPPOLD Pierre JAUDON Colonel CONVERSE Paul d'ESTOURNELLES de CONSTANT Professeur Th. RUYSSSEN Professeur J.-J. PRUDHOMMEAUX DANDIEU</p>	<p>- 4 - M.M. Président Nicholas Murray BUTLER Sénateur d'ESTOURNELES de CONSTANT Baron ADELWARD Recteur Paul APPELL Professeur J.T. SuOTVELL SANGRO de OLANO Professeur J. REDLICH Hellmuth von GERLACH Professeur FOERSTER Professeur Charles GIDE EFREMOFF Député Justin GODART Sénateur LA FONTATNE Professeur Henri LICHTENBERGER LEJEUE, Représentant de M. Albert THOMAS Professeur NIPPOLD Pierre JAUDON Colonel CONVERSE Paul d'ESTOURNELLES de CONSTANT Professeur Th. RUYSSSEN Professeur J.-J. PUDHOMMEAUX DANDIEU</p>
---	--	--

## (a) Model Other

<p>- 4 - M.M. Président Nicholas Murray BUTLER Sénateur d'ESTOURNELLES de CONSTANT Baron ADELWARD Recteur Paul APPELL Professeur J.T. SHOTWELL SANGRO de OLANO Professeur J. REDLICH Hellmuth von GERLACH Professeur FOERSTER Professeur Charles GIDE EFREMOFF Député Justin GODART Sénateur LA FONTAINE Professeur Henri LICHTENBERGER LEJEUNE, Représentant de M. Albert THOMAS Professeur NIPPOLD Pierre JAUDON Colonel CONVERSE Paul d'ESTOURNELLES de CONSTANT Professeur Th. RUYSSSEN Professeur J.-J. PRUDHOMMEAUX DANDIEU</p>	<p>- 34 - SM.NM. Président Nicholas Murray BUTLER sSénateur d' REsSTdOURNELLES de CoONaSITAN:T MBaron ADELsYWARD BRecteur Paul ApPLPNEILT Professeur 3J.7T. SHCOTNWPEILIL SANGGhRO de COILA44NOO pProfesseur J. BRETDLIOCNIH MHellmuth von GERLACNH pProfesseur FOFERCSTREER pProfesseur Charles GILDRE NEEFNREMOTFTF pDéputé DJustin dGoOpDART sSénateur LaA FoONLTAIRNRE FProfesseur BHenri LIOCNTENRBERdGER LETJEUENE, BReprésentant de M. Alhbert TLHaOcMnAsS Professeur NIPPGLRD Pierre TJANUDON Colonel CoONvVERSE Paul d'EsSfToOuURNETLILES de CoORNsSTANT FProfesseur ITh. BRUTYSsREN Professeur 3J.-3J. PRpURDOHNOMUMREAUX BDANRDIESU</p>	<p>- 3 - S.N. Président Nicholas Muray BUTLER sénatour d' RsfduRLIES de CoNaIAN: Maron ADELsYARD Becteur Paul ApLNIT Professeur 3.7. SHCTNPIL SANCh0 de CIA440 professeur . BETLION Mellmuth von GERLACN prfesseur FOFRCTRR professeur Charles GILR NENEMOTT péputé Dustin dopAR. sénateur La FoNLAIIR Frofesseur Benri LIONTERERdER LETEUEE, Beprésentant de M. Alhert Lacns Professeur NIPPGLR Pierre TANDON Colonel CoNVERSE Paul d'EstouRETIES de CoRstANT Frofesseur lh. BUTSsRN Professeur 3.-3. PRpRONMURAU BANRIES</p>
---	--	--

## (b) Model War

Figure 5.5: Versus text for the page 1358-4

### 5.2.3 A Lack of Influence of the Lexicon in Prediction Errors

In Table 5.1, the most striking thing is the Levenshtein distance in characters. The gap, between the model with the set it trained on, and the model trained with the other set, is really high. For the set War, the number has more than doubled, and for the set Other, it has been multiplied by almost ten. I can observe, as well, between the length of the reference and the prediction, that all predictions are missing characters compared to the reference (41 SW/MW; 83 SW/MO; 59 SO/MO; 136 SO/MW). Frequent deletions across models, coupled with fewer insertions, partly explain the discrepancy. Another interesting element is that the substitutions are massive. The smallest number is 274 for the SO/MO. The SO/MO is at more than ten times that, with 3301. Overall, the Word Accuracy percentages are not that bad. For the model applied to the set it trained on, the results are good, with 95% for the set War, and 97% for the set Other. For the model applied to the other set, this is not the same for both. The model Other wasn't so bad on the set War, with a Word Accuracy (Wacc) of 91%. On the other hand, the model War was not so great with the set Other, with a word accuracy of only 83%. This leads to wonder, that, if the set War had had the same number of pages on its training set as the set Other, it might have been better. Indeed, the problem of the model might come from the lack of content rather than the content itself.

The results of the model War Retrained partially prove it. It did a little better on its own set, by a dozen characters. Nevertheless, the prediction for the set Other is fifty characters short, compared to the model War. When applied to the set Other, there are far more hits<sup>16</sup> (+200) and way fewer substitutions (-230). The deletions did not really vary, and there were fewer insertions (-50). For the set War, there were more hits (+161) and fewer substitutions (-135), deletions (-26), and insertions (-14). It correlates with the length of the prediction that is better than with the model War. As for the accuracies, with the set War, the Wacc improved by two points and the CER is very low (0.51%). With the set Other, the CER lowered but is still pretty high (3.76%). The Wacc improved by one point. This hints that, although doubling the amount of data can improve the model, doing it with the same data leads to overfitting more than better recognition. It could explain why the model was better with the set War, but still pretty bad with the set Other.

To further understand these observations and help me answer the question of the lexicon impact, I examined the model's behaviour on specific pages.

First, when observing the models applied to the set Other, as shown in Tables 5.2,

---

16. A hit is a correct prediction, i.e. character of reference = character of prediction

5.4, and 5.6, few things stand out. The model Other did rather well. The model War was a big miss, except for one page. The model War Retrained is better than the model War, but not by much.

The model Other has an accuracy between 96 and 99% for four out of the five pages. The difficulties it encountered seemed to come either from uppercase characters or some hardly readable characters in the facsimile. For the page, made of only persons' names, the WER of 9% is due to several substitutions and insertions, probably because they were uppercases.

The model War did good on one page, with a Wacc of 91%. The errors were mostly due to uppercases, numbers and some similarly shaped letters, like p and b, m and n, and o and e. For the other pages, the metrics are awful, with even a Wacc of 25% for the page made of uppercases. When looking at the "versus text", no reasoning behind those results can be deduced. It seems that they are no visible pattern to the prediction errors, except for a few similar looking characters.

The model War Retrained usually have a better WER and/or CER, with an improvement of ten points on the complex page of uppercases. However, it struggles on the same elements as the model War, like uppercases, numbers, specific signs, similar looking letters. However, its mistakes were different from the model War. Some words were incorrectly recognized in both cases, and others were recognized correctly with the model War but not with the model War Retrained, and vice versa. This suggests that, despite improving a model's accuracy with fine-tuning, doing it with the exact same data as the original model is rather pointless.

Then, I observed the models applied to the set War, as shown in Tables 5.3, 5.5, and 5.7. The model Other did not do so great this time. On the other hand, the model War was not so much better, especially considering the model was applied to the set it learned from. The model War Retrained improved, if it is compared to the prediction of the model War. Yet, it could be overfitting rather than better recognition's skills.

The model Other, just like before, had problems with the uppercases, and it also had difficulties on items from the letter's opener, such as header, letterhead, dateline, or title. They can be found plenty in the pages of the set. Lastly, it struggled on punctuations, whether it is forgetting some or adding inexistent ones. In some cases, the "versus text" demonstrates completely gibberish's predictions, as in Figure 5.6.

The model War, in addition to still struggling with the same elements as with the set Other, created errors on very random situations. It could be double characters where only

one was correct or beginning of words with lowercases put in uppercases or vice versa. It had difficulties as well with the elements from the opener. Those elements being mainly made of uppercase characters could explain the issue, and imply that the training data needs a variety of character cases, from lowercases to uppercases.

Overall, the model War Retrained got better on the opener’s elements. It probably learned to recognize it doubly, hence being a result of an overfitting rather than a skill. The WER and CER globally improved, making it the best model out of the three for this set. However, the “versus text” demonstrates that the model still encounters the same challenges that the model War sometimes had, and it even created new one, such as with numbers.

<p>LE DÉSARMEMENT MORAL de L'ALLEMAGNE.- GERLACH et FOERSTER.- RÉPONSE à VOTRE LETTRE du 8 DÉCEMBRE.</p>	<p>VE FÉSARMEENT MORAL de j'ATLACR.- DEULACS et gOEMCIER- RUFONSR A VORE LUTRE du S FÉORÉERE.</p>
(a) Reference	(b) Prediction

Figure 5.6: Versus text from the model Other applied to the page 948-1

To conclude these observations, the evidence suggests that the difficulties encountered by the lexicon-based models were due to a lack of content in terms of quantity, rather than a specific vocabulary. There were no error patterns linked to the absence of certain words in the datasets. When the models misrecognized characters, it was mostly due to the scarcity of uppercase letters in the training data. In the predictions of either model, the vocabulary specific to each of them was usually recognized correctly, regardless of the pages it was applied to. In addition to these peculiar characters, another frequent issue, as observed with the “versus text”, was the confusion between similar looking characters. For example, there were confusions between b and p, n and u, or c and o. Since these characters are not consistently misrecognized, the characters surrounding them may influence recognition, particularly when the adjacent letters create visual ambiguities. Given these findings, this might be an interesting avenue for further research.

During this part, I conducted various experiments centred around the lexicon of the training data of the model. All my results indicated that it had no impact whatsoever on the training of the model. However, some errors seemed to have been a question of placement, and the errors were more frequent when the character was in a certain sequence instead of another. Consequently, a need to understand this phenomenon prompted a

deeper investigation of the composition of the ground truth. Instead of focusing on a word-level study of the training data, as I did until now, the focus turns to an infralexical level examination, which will be done through short sequences of characters.

## PART III

---

### EXPLORING THE IMPACT OF THE INFRALEXICAL LEVEL: THE N-GRAM





# STUDYING THE N-GRAM AND ITS DISTRIBUTION WITHIN THE DATASET

---

The experiments presented in the previous part disproved the theory of the impact of the lexicon, which lead me to formulate a new hypothesis. My various observations oriented me towards the effect of the n-grams during the text recognition. To determine its impact, it is also key to know the composition and distribution of the n-grams of the dataset. The n-gram is a frequent source of study in the domain of Natural Language Processing (NLP),<sup>1</sup> allowing them to perform various tasks/be used in various applications such as “automatic machine translation [...], text classification [...], search engines [...] and dialog systems.” [Eisenstein 2019, p. 1] but it is not frequently used in the DH community. Therefore, in this chapter, I am presenting the various outcomes of the studies I conducted on the n-grams of the dataset of the correspondence of Paul d’Estournelles de Constant. First, however, I will clarify the implications of these n-grams for the subsequent experiments.

## 6.1 A New Level of Study

### 6.1.1 A Choice Induced by the Previous Experiment’s Results

Until now, my experiments have focused on the word-level analysis of the training data. My interest was in its thematic composition, but I found that the model’s performance was unrelated to it. However, the experiments were not totally inconclusive. They gave me some new leads to follow regarding the prediction errors. Observing the errors during the comparative analysis<sup>2</sup> highlighted the fact that the models were occasionally confused with some characters. They changed one for another having a similar appearance. This raised questions about the influence of a character’s placement in context, and how the

---

1. “Natural Language Processing is the set of methods for making human language accessible to computers” [Eisenstein 2019, p. 1]

2. See section 5.2

model might predict an incorrect character when surrounded by specific ones. It would have learned, from the training data, the sequence of characters that was predicted more than the sequence in the reference. Therefore, given these findings, and based on guidance received during my PhD,<sup>3</sup> I decided to explore this lead. It could possibly be the answer to the model's performance. I chose to work at a level placed between the character and the word. Character-level was done in OCR but became too complicated for HTR, due to the handwriting way of linking characters to each other. Word-level was proven ineffective in the prior part. I will then study the infralexical level. It could be described as “locating the boundaries of low-level linguistic units such as syllables or phonemes” [Bagou and Frauenfelder n.d.]. It means that I will go even deeper in the exploration and application of my training data.

### 6.1.2 Same Dataset, New Interests

As explained in the previous subsection, the infralexical level can include units such as the syllables or the phonemes.<sup>4</sup> They “are the distinctive sounds in a language. Every spoken language has phonemes, but they differ from language to language.” For my experiment, the studied unit is the n-gram. According to [Jurafsky and Martin 2008, p. 33], “an n-gram is a sequence of n words: a 2-gram is a two-word sequence of words like ‘please turn’, ‘turn your’, or ‘your homework’, and a 3-gram is a three-word sequence of words like ‘please turn your’, or ‘turn your homework’”. N-grams are typically described as a sequence of n adjacent symbols in particular order. The symbols can be letters, syllables, phonemes, or words (like with [Jurafsky and Martin 2008]), but for the subsequent experiments, the n-gram designates a sequence of n characters.

The lexicons in the test sets did not significantly affect the model's recognition abilities, but they differ sufficiently to provide valuable insights for the new experiments. There are two primary reasons for this choice. First, while conducting the lexicon and content analysis, I was able to acquire extensive knowledge on those sets. I know that they are made of varied words, whether it is thematically or grammatically. I know that, despite not

---

3. I am grateful to Jean-Philippe Magué, member of the Comité de Suivi Individuel for my thesis. During our first annual meeting, as my lexical analysis was heading towards an impasse, he advised me to explore the influence of an infralexical level, and more precisely the n-grams. He also mentioned the idea of considering the context during the recognition. Those are the main aspects around which the subsequent experiments revolve.

4. A phoneme, as defined by [Denham and Lobeck 2013, p. 72], is a “unit of sound that makes a difference in the meaning of a word”

being transcribed from the same documents, they have many common elements between them. Those two opposite elements should give enough leeway to conduct the experiment. Second, as observed during the analysis, the set War, set Other, and set Ground Truth, are of a considerable difference in size. The set War, with its 31 documents, is a small dataset. The set Ground Truth, with its approximately 500 documents, is a big dataset. Lastly, the set Other is placed in the middle. It is far smaller than the set Ground Truth, with its 76 documents. It could then be qualified as a medium dataset. To understand the impact of n-gram frequency, it is crucial to compare datasets with varying sizes, to understand what the model learned and in what quantity. Therefore, analysing instances with frequencies of 10, 100, or 1000 should provide valuable insights.

With these goals in mind, I will now determine which n-grams to analyse and the methods for doing so.

## 6.2 The N-Gram from Every Angle

### 6.2.1 Which N-Grams Will Be Studied?

In this experiment, I explore the impact of n-grams in three different lengths. My dataset is made of typewritten documents, which contain no cursive text, making it clear that the ATR software with neural networks does not rely on bounding boxes that process recognition character by character. Moreover, as previously mentioned, the focus on n-grams is closely tied to understanding character context in the training data. Therefore, the smaller unit is a sequence of two characters. They will be called 2-grams, or bigrams. I am also interested in the bigger unit of a sequence of three characters. They will be called 3-grams, or trigrams. Lastly, if n-grams significantly affect the model's recognition ability, I also want to assess the impact of larger sequences. For this reason, the last and biggest unit is a sequence of four characters. They will be called 4-grams, or tetragrams. By studying sequences of different lengths, I aim to determine whether increasing the n-gram size improves recognition performance by providing more context.

Consequently, all the lists, diagrams, and visualisations produced during this experiment are always made in three parts, one for each unit of n-gram. Given the differing sizes of the n-grams but also of the test's sets, there should be wide gaps between the sizes of the results. Moreover, there could be some instances where there will not be an n-gram for each case, or where the context is going to be way different from a bigram, to

a trigram, to a tetragram.

The differences in n-gram lengths result in variations for the output, as illustrated in Table 6.1. The three-characters token ("all") does not appear in the list of tetragrams. It is not long enough. The four- ("wild") and sixteen-characters tokens ("superficialities") are entirely present in the lists of bigrams and tetragrams, as they are multiples of 2 and 4. However, they are not in the third one. The twelve-characters token ("definitively") is entirely partitioned in each list, since it is a multiple of each unit. Lastly, the five-characters token ("fewer") does not have its ending characters in each case. The numerous sequences of characters that can already be retrieved with such a small example (19, 12, and 9 for the lists of bigrams, trigrams, and tetragrams, respectively) hint that the three sets I am using for the n-gram experiments should provide enough data to correctly evaluate the impact of the n-grams.

Token	all	wild	fewer	definitively	superficialities
Bigram	"al"	"wi", "ld"	"fe", "we"	"de", "fi", "ni", "ti", "ve", "ly"	"su", "pe", "rf", "ic", "ia", "li"; "ti", "es"
Trigram	"all"	"wil"	"few"	"def", "ini", "tiv", "ely"	"sup", "erf", "ici", "ali", "tie"
Tetragram	X	"wild"	"fewe"	"defi", "niti", "vely"	"supe", "rfic", "iali", "ties"

Table 6.1: Examples of n-gram divisions according to the tokens given

## 6.2.2 How to Obtain My Series of N-Grams?

For this experiment, I was starting from scratch again, because I needed the texts of the sets as they were initially. Despite using the same data as I did for the lexicon and content analysis, I required more elements from the content of the sets than I did before. Therefore, I created a new version of Script P.2.4, Script P.3.2, since, for this experiment, preserving punctuation and original casing was essential for accurate tokenization. Furthermore, original casing is also required, because the study of the context will differentiate between an n-gram with only uppercases,<sup>5</sup> uppercase as the first character then lowercases,<sup>6</sup> and only lowercases. Afterwards, the text having now been partitioned in tokens, I created

5. They will be called All Caps for this experiment

6. They will be called Initials for the experiment

n-grams with those tokens. To do so, I created Script P.3.4. I used in it the Python module `textwrap` and the method `wrap()`, ideal to easily divide my texts into the n-grams I need. They will “wrap the single paragraph in text (a string) so every line is at most width characters long”.<sup>7</sup> The width is, in that case, 2, 3 or 4, i.e. the unit of the n-grams. This wrapping provided several long lists of bigrams, trigrams and tetragrams, for each set. Its content will look like the one in Table 6.1. The lists produced from the two scripts mentioned above can be observed, in a [Python file of results](#):

- In the lines 12 (tokens), 18 (4-grams), 30 (3-grams) and 42 (2-grams) are elements from the set Other;
- In the lines 55 (tokens), 61 (4-grams), 73 (3-grams) and 85 (2-grams) are elements from the set War;
- In the lines 97 (tokens) 104 (4-grams), 116 (3-grams) and 128 (2-grams) are elements from the set Ground Truth.

Though those lists contain extensive data, they are currently not giving me any insightful information, because it lacks details on the distribution. To derive meaningful insights, I needed to create dictionaries that count the frequency of each token in the data. Using the `counting()` function, presented in the subsection 4.2.1, I called, in Script P.3.3, the various lists created, and produced dictionaries from it. Each dictionary was outputted from the highest occurrences to the lowest. The dictionaries produced by the script mentioned before can be observed, in the same file of results as the lists:

- The lines 15 (tokens), 21 (4-grams), 33 (3-grams) and 45 (2-grams) are for the set Other;
- The lines 58 (tokens), 64 (4-grams), 76 (3-grams) and 88 (2-grams) are for the set War.
- The lines 101 (tokens), 107(4-grams), 119 (3-grams) and 131 (2-grams) are for the set Ground Truth.

The numbers of occurrences in each of those lists and dictionaries of raw, unclean data are available in the tables 6.2 and 6.3.

---

7. <https://docs.python.org/3/library/textwrap.html#textwrap.wrap>

The extraction and division of the sets' text is done automatically, which means that irrelevant data is present in the dictionaries, notably sequences of characters incompatible with the purpose of the dictionary it is in. To be able to rightly exploit the data I just produced, I then need to clean it. As mentioned in the previous subsection, the division is made on the token according to the unit required. While I only included the right units in Table 6.1, in the current dictionaries, the result is not the same. With a token like “fewer”, the wrapping would have been done as such:

- Bigram: “fe”, “we”, “r”
- Trigram: “few”, “er”
- Tetragram: “fewe”, “r”

In consequence, the dictionaries might be full of those units of one character. They could also be units of two or three characters in lists that should contain respectively units of three and four characters. The cleaning of this data has been done manually, using regular expressions,<sup>8</sup> such as the ones seen in Table 6.4. Then, the data has been separated into three subsets: All Caps, Initials, and Lowercases. The [lists](#) are ordered from largest to smallest occurrences. When several n-grams have the same number of occurrences, they are ordered alphabetically.

Once supplied with all the useful data for my experiment, I resolved to do some more transformations, and I also wanted visualizations. I could then be as precise as possible with the observations I will make with those data.

### 6.2.3 Various Visualizations to Better Understand My N-Gram Series

If the numbers in the tables 6.2 and 6.3 are a good indication of the amount of data available, it is unlikely that I will be able to learn everything about them. Moreover, not everything will be valuable for my study. Therefore, as the datasets were too large to analyse completely, I decided to create subsets of the results, focusing on the most and least frequent n-grams, which could provide more actionable insights. To do so, it was necessary to establish thresholds for least and most popular. For the least popular, I decided that it would only comprehend the n-grams with an occurrence of 1, because

---

8. Language, using algebraic notation, for characterizing a set of strings

	Set Other	Set War	Set Ground truth
Tokens	20499	7944	123407
Bigrams	46832	18137	257751
Trigrams	35610	13762	195195
Tetragrams	29895	11578	164356

Table 6.2: Occurrences of the raw lists created from the sets

	Set Other	Set War	Set Ground truth
Tokens	4408	2199	13771
Bigrams	961	726	1528
Trigrams	2651	1782	5371
Tetragrams	3915	2321	8968

Table 6.3: Occurrences of the raw dictionaries created from the sets

Regular expression	What does it do?
<code>^(' ").{3}(' "): .+</code>	It searches lines with a single or double quotation marks (' ") at the beginning of the line ^, then 3 characters .{3} immediately followed by the closing quotation marks (' ") then the colon and the number of occurrences : .+. This regex ensures to dismiss all non-trigrams lines. This can also be easily adapted to bigrams or tetragrams, by changing the number in the curly brackets.
<code>.+(- " \. , € / £).+</code>	It searches, in the line, cases where one of the characters mentioned in the parenthesis is present between the opening of the line and the occurrence's number.
<code>'[0-9]{3}': .+</code>	It searches, in the line, cases where, between the quotation marks, there are only numbers. This can be adapted to other n-grams by changing the number in the curly brackets, like above.

Table 6.4: Regular expressions to clean the dictionaries of units of trigrams



it is the minimum occurrence that I can obtain. For the most popular, comparing the various sets demonstrated that the number should not be too high. Otherwise, not every set would be present, and this would not be ideal for my experiment. Consequently, the most popular lists take into consideration n-grams with 11 or more occurrences, as 11 to 20 is the biggest number of occurrences some n-grams can attain. Those lists of n-grams are available in the same [Python file of results](#) that was mentioned before:

- The least popular n-grams can be found:
  - At the lines 27 (4-grams), 39 (3-grams) and 51 (2-grams) for the set Other;
  - At the lines 70 (4-grams), 82 (3-grams) and 94 (2-grams) for the set War;
  - At the lines 113 (4-grams), 125 (3-grams) and 137 (2-grams) for the set Ground Truth (SGT).
  
- The most popular n-grams can be found:
  - At the lines 24 (4-grams), 36 (3-grams) and 48 (2-grams) for the set Other;
  - At the lines 67 (4-grams), 79 (3-grams) and 91 (2-grams) for the set War;
  - At the lines 110 (4-grams), 122 (3-grams) and 134 (2-grams) for the set Ground Truth.

Additionally, they are the sequences that supposedly have the biggest impact, so I created [large tables](#) that contain, in three columns, the bigrams (column 1), trigrams (column 2), and tetragrams (column 3) that have more than ten occurrences. They are separated by set, ordered alphabetically, and no distinction has been done between All caps, Initials, and Lowercases.

From those lists of n-grams, I decided to also do some comparison between the sets. I wanted to see, at the infralexical level, how much the sets have in common or how different they are. To do so, I used the same methods that I presented in the subsection 4.2.1,<sup>9</sup> which I implemented in Script P.3.1. The commonalities and differences of the n-grams are retrieved between sets, as well as between sets and subsets. First, the comparison is done between the ensembles Other/War, Other/Ground Truth, and War/Ground Truth. Then, the ensemble Ground Truth is rather voluminous, and I wanted to have a proper idea of the distinction between the sets War and Other. The comparison is therefore done

---

9. Those methods are `difference()` and `intersection()`. They find respectively the unique and the common elements of the arguments given to them.

between the ensemble Ground Truth and the n-grams unique to the ensemble War, the ensemble Ground Truth and the n-grams unique to the ensemble Other, and between the ensemble Ground Truth and the n-grams common to the ensembles War and Other. The content of those lists are available respectively for the [least](#) and [most](#) used n-grams.

In this subsection alone, I mentioned a large quantity of data and numbers. To obtain a clearer understanding of the quantity and scale of the data, I inserted the results into tables, rendered with numbers or percentages, in Annex C. The tables present the distribution for each set, and then the partitioning produced by the comparison. Finally, some tables contain many rows, because some n-grams have very high occurrences. It was necessary at times to create numerous levels in the n-grams with more than 10 occurrences. In order to get a more tangible knowledge of this distribution and to allow easier identification of significant patterns across sets, the tables have been supplied with a graphical representation. It appears in the form of bar charts, that can be found in Annex M. For example, Figure 6.1 is the graphical representation of Table 6.5:

- Three colours are used in the bar chart:
  - The green displays the numbers of the bigrams.
  - The blue displays the numbers of the trigrams.
  - The yellow displays the numbers of tetragrams.
- The vertical axis represents the quantity of tokens.
- The horizontal axis is divided into parts, one for each range of occurrences.

I am now equipped with plenty of material, statistics, and figures about the distribution of the n-grams in and between the sets, and I will then be able to share my observations about it. Furthermore, I will also draw few first conclusions as to why some models might work better than others.

## 6.3 What Did We Learn About the Distribution of the N-Gram?

### 6.3.1 A Significant, Sizable Gap Between the Sets

First, I observed the tables in the first section of Annex C.

I noticed that the set Other (SO) has significantly more n-grams than the set War (SW).

	2grams	3grams	4grams
1	32	101	119
2 to 5	43	61	58
6 to 10	10	12	9
11 to 50	20	10	8
Total	105	184	194

Table 6.5: Table of the distribution of the Initials n-grams in the set War

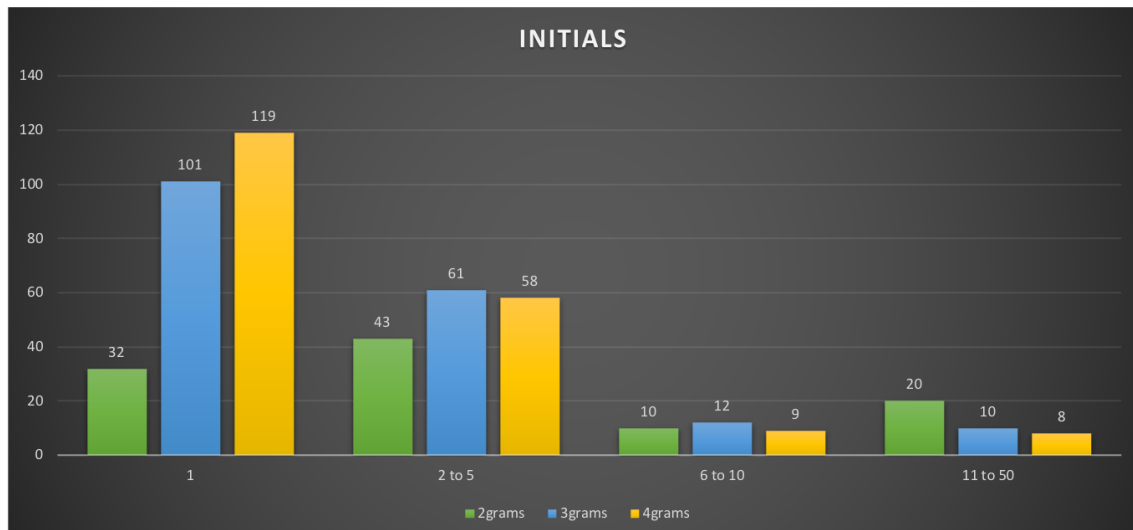


Figure 6.1: Bar chart representation of the distribution of the Initials n-grams in the set War

The numbers of n-grams between both sets are multiplied by almost 3 for tetragrams, and by 2.5 for trigrams and bigrams. The difference in All Caps is relatively minor, with only 100 more for SO, usually. It is also very few for the rest, as they are barely 1-3% of all the n-grams.

As for the repartition in percentages, both sets are pretty equivalent. The numbers in Lowercases are huge, notably in bigrams, with more than 32,000 for SO. It adds up with the quantity of text. As for the set Ground Truth (SGT), I can see that those numbers are even bigger, with a total in the hundred of thousands. The considerable amount of pages and lines in the training data are the likely explanation for this. Compared to the other sets, SGT is multiplied by 14 (SW) and 5.5 (SO) for tetragrams, trigrams, and bigrams. The percentage of repartition is also quite the same as the other two sets, except for the All Caps. Its percentage is usually 2 or 3 points bigger than the other two. SGT was strengthened with lines of only uppercases, so this specific composition easily explains this situation.

Between the table of all n-grams and those counting the distribution by range of occurrences, there are massive discrepancies. The numbers of n-grams were divided by 4.2 in tetragrams, 9.3 in trigrams and 48.8 in bigrams for the SO. It was divided by 2.7 in tetragrams, 5.2 in trigrams and 23.9 in bigrams for the SW. It was divided by 10.6 in tetragrams, 27.02 in trigrams and 185.6 in bigrams for the SGT. Those numbers really highlight the size difference of the three sets. However, such high dividers also imply that there are many n-grams with plenty of occurrences. They are more present than those with only 1. This could explain why the models are working, and it is even more probable for the SGT. It seems to have substantially higher occurrences than the other sets, and also has a great diversity of single occurrences, as we will observe afterwards.

Then, I concentrated my observations on the tables of unique n-grams and their occurrences' distribution. From it, I obtain valuable additional details.

For the All Caps data in Table 6.7, SO and SW don't have many in All Caps. It could explain the difficulty in recognizing those parts in the text. It does not have enough examples to learn how to recognize it.

The SO mostly has 1 to 5 occurrences in the text, and not many with more than 10 occurrences. For the SW, it is pretty much the same. There are many single occurrences, but barely more.

On the other end, the SGT should be much better at recognizing those. Firstly, it has notably more unique n-grams of All Caps: it is about five to ten times more for trigrams

and tetragrams. Secondly, their occurrences are also pretty high: they go to 51 to 100, and even more than 100. The others were usually at 50 occurrences maximum. Even more, for the bigrams, for example, the numbers of unique n-grams do not differ extensively from the other sets (281 for SGT, 143 for SO and 105 for SW), but the occurrences are substantially higher. There are 101 unique n-grams with 11 to 50 occurrences, 25 with 51 to 100 and even 15 with more than 100.

The Initials in Table 6.9 show that SO and SW sometimes have substantially more than All Caps. It is the case with the tetragrams and trigrams, but not for the bigrams. Those numbers of Initials are similar to All Caps, whether it is SO or SW. The SGT has even less unique Initials for bigrams and trigrams than for All Caps, but it is immensely higher for tetragrams. In terms of occurrences, there are many n-grams with single occurrence or 2 to 5 occurrences, and it is the case for all three sets.

For the SO, there are also a good number (30 to 40) with 6 to 50 occurrences for all n-grams. There are even few with more than 50 for bigrams.

The numbers are way lower for SW. There are barely more than 10 n-grams in the ranges 6-10 and 11-50, except for about 20 bigrams with 11 to 50 occurrences. An explanation for this situation is that Initials could be pronouns or conjunctions, and they are usually made of only two characters. It would mean that they exist in the bigrams, but can't be found in the other n-grams.

At their own scale, the SGT has a pretty similar distribution. There are 100 to 200 n-grams with 6 to 50 occurrences. There are only 10 to 15 for the trigrams and tetragrams that have 51 to 100 and more than 100 occurrences. Similarly to the other sets, the numbers are a little higher for the bigrams, with 20 to 30 with the high occurrences.

The Lowercases in Table 6.11 are on another level as they are far superior. They are in the thousands for trigrams and tetragrams, and hundreds for bigrams. It coincides with the fact that the numbers of all n-grams were higher for the lowercases, compared to All Caps and Initials, with one or two more digits typically. The table of occurrences allows me to observe why the divide is that big. There are many recurrent n-grams with lots of occurrences, as proven by the additional rows of the tables. Two or three rows are added, depending on the set. The tables 6.7 and 6.9 were going to 50, 100 and sometimes a bit more over, in terms of occurrences. With the lowercases, the occurrences reach ranges of 100, 500, 1000. They even reach more than 1000 in the case of two bigrams in the SO, and 52 bigrams and 7 trigrams in the SGT. Therefore, I could infer that there are fewer unique bigrams, but they are considerably more present. On the contrary, there are many unique

	Set Other			Set War			Set Ground Truth		
	2grams	3grams	4grams	2grams	3grams	4grams	2grams	3grams	4grams
1	57	121	99	48	68	58	27	365	528
2-5	63	34	17	40	22	13	79	359	303
6-10	14	7	4	11	4	4	34	81	34
11-50	9	2	2	6	2	1	101	53	29
51-100	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	25	9	3
>100	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	15	3	2
Total	143	164	122	105	96	76	281	870	899

Table 6.7: Distribution of the n-grams in All Caps

	Set Other			Set War			Set Ground Truth		
	2grams	3grams	4grams	2grams	3grams	4grams	2grams	3grams	4grams
1	34	158	218	32	101	119	29	287	515
2-5	40	150	160	43	61	58	54	273	356
6-10	39	38	38	10	12	9	24	99	95
11-50	44	33	25	20	10	8	68	107	96
51-100	5	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	27	16	13
>100	1	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	22	13	10
Total	163	379	441	105	184	194	224	795	1085

Table 6.9: Distribution of the n-grams in Initials

tetragrams in lowercases, with 818 (SW), 1,005 (SO), and 1,349 (SGT). Meanwhile, there are only 69 (SW), 61 (SO) and 76 (SGT) in bigrams. A justification could be that there are plenty of unique words or long combination of words, made of bigrams, appearing recurrently.

### **6.3.2 The Most and Least Popular N-Grams: A Promising Lead Towards Understanding the Efficiency of the Model**

I can observe and compare the three sets of the least and most popular n-grams, as they are displayed in Table 6.12. It enables me to detect even more differences between them. For set Other (SO), the numbers and distribution is quite disparate from an n-gram to another. With the tetragrams, there are much more unique n-grams (45%) than recurrent ones (9%). In the trigrams, the percentage of unique n-grams is higher than that of the recurrent. The difference is not too much, though, with 32% and 21%. It means that one third of the set is unique n-grams, and one fifth is recurrent ones. On the contrary, it is the opposite for the bigrams: almost half of them (43%) are recurrent n-grams, and one fifth (21%) are unique.

For set War (SW), the numbers change. The difference in tetragrams is greater between single and recurrent n-grams. More than half (56%) are for unique n-grams, and less than one twenty-fifth (4%) are for recurrent ones. For the trigrams, the recurrent n-grams have ten points less than SO (11%). The single n-grams have ten more (41%). Lastly, for the bigrams, there are fewer recurrent ones than in SO, with only one third of them (37%). One fourth (26%) of all unique n-grams are single occurrences.

Finally, for the set Ground Truth (SGT), the percentages are not the same as the other two. While the percentages of most popular n-grams are higher, those of least popular are lower. The difference between single and recurrent n-grams for tetragrams and trigrams is not that large. There are about one third (36%) for the trigrams, one fifth (20%) for the recurrent tetragrams, and less than two fifth (36%) for the single tetragrams. The bigrams are the biggest indicator of the gap between the SGT and the other sets. There are not many unique n-grams, only 12%, but more than half (57%) are recurrent n-grams. It means that the model should be skilled on more various words.

Going even deeper in the observations, I created comparison between the various sets, as rendered in the section C.5 of Annex C. With it, I can obtain even more leads on the possible efficiency of the models. Regarding the most popular n-grams, SW seems to have

	Set Other			Set War			Set Ground Truth		
	2grams	3grams	4grams	2grams	3grams	4grams	2grams	3grams	4grams
1	61	390	1005	69	418	818	76	440	1349
2-5	75	543	924	68	425	534	88	531	1510
6-10	30	238	243	42	166	97	48	311	602
11-50	113	332	214	110	134	58	89	650	907
51-100	42	46	15	40	11	1	33	207	149
101-500	83	20	4	36	4	∅	124	203	79
501-1000	10	∅	∅	∅	∅	∅	44	17	8
>1000	2	∅	∅	∅	∅	∅	52	7	∅
Total	416	1569	2405	365	1158	1508	554	2366	4604

Table 6.11: Distribution of the n-grams in Lowercases

	Set Other			Set War			Set Ground Truth		
	2grams	3grams	4grams	2grams	3grams	4grams	2grams	3grams	4grams
1	152	669	1322	149	587	995	132	1092	2392
% on the total	21%	32%	45%	26%	41%	56%	12%	27%	36%
11 and more	309	433	260	211	161	68	600	1285	1296
% on the total	43%	21%	9%	37%	11%	4%	57%	32%	20%
Total	722	2112	2968	575	1438	1778	1059	4031	6588

Table 6.12: Distribution of the least and most popular n-grams



the same recurrent n-grams as SO and SGT.

The numbers of unique n-grams to SW are often seriously low. There are even nulls for the bigrams with SO and every unit of n-grams for SGT. Compared to the SGT, the same can be observed for the SO. It has barely any unique n-grams to itself. Between SO and SW, there are a lot of common n-grams in bigrams. There are less in trigrams and tetragrams, where SO has more than 60% of unique ones.

The numbers are all pretty low for SW. It means that there are not many unique elements to SW that are only unique recurrent n-grams of the SW. According to the table, there is less than 10% for all. It makes me expect that those n-grams should be well recognized in SW. It should happen no matter the model used, since it knows them.

On the other hand, in SO, many are unique to the set. For bigrams, the unique elements of SO that are not found in SW are only 30% of all recurrent occurrences. For trigrams and tetragrams, it is respectively three fifth (64%) and three-quarter (77%) of all n-grams. Therefore, those n-grams are likely not going to be recognized by the model War (MW). Finally, the model Ground Truth (MGT) should be able to perform correctly with both sets. Compared to SGT, there is no element absolutely unique to SW. For the SO, the percentage of uniqueness to the set compared to SGT is even lower than SW to SO. It is less than 4% for every n-gram. Consequently, I can assume that the recognition should not be hindered. As for the least popular n-grams, there are a completely different batch of numbers. Indeed, there are many more single n-grams. Few commonalities of single n-grams exist between SO and SW, SO and SGT, or SO and SGT. The percentages of unique single n-grams are pretty much the same whether it is bigrams, trigrams or tetragrams, and SO or SW. This percentage is about 80%. It comprehends the majority of all.

With SGT and the other sets, those numbers climb to 90-95%. This could imply problems during the recognition, no matter the model. The models could have trouble recognizing some n-grams that it never saw. The model Other and model War might even have issues on their own set, because it did not learn well enough some n-grams.

To conclude this chapter, if the n-grams are indeed at the centre of the model training, then, the distribution between the sets that I just observed should explain the difference in recognition's accuracy.

As there are many bigrams with plenty of occurrences in SO, this would explain why the recognition is working.

On the other hand, the numbers are far down with SW. It means that it learns fewer

patterns. It could explain the lower recognition's ability of its model. In both cases, the All Caps and sometimes Initials are as well less present. It could prevent an accurate recognition.

Regarding those results, the model Ground Truth should have no substantial issues in recognizing both sets. It has a lot of recurrent n-grams. However, it is possible that it makes some mistakes. It has learned more characters' sequence with various number of examples, which could lead to confusion and a wrong recognition.



# EVALUATING THE IMPACT OF THE N-GRAM USING PREDICTION ERRORS

---

To establish the correlation between model's accuracy and lexicon, I observed the metrics of the model's predictions. Essentially, it meant finding a link between the amount of the errors and the presence of a specific lexicon in the training data that generated the model used. In this chapter, the followed approach is rather similar, but the arguments are not the same any more. Instead of using the lexicon, which proved ineffective in Part II, I am exploiting a smaller unit, the n-grams. These were thoroughly retrieved and presented in the previous subsection. Then, instead of focusing on metrics, I am directly examining the errors produced in the prediction, before introducing the results I obtained and the conclusions I drew from it.

## 7.1 Obtaining an Error List

### 7.1.1 The Data from the Comparative Analysis As the Source

Before evaluating the impact of the n-gram on the model's accuracy, through its errors, it was necessary to retrieve those errors.

During a previous experiment,<sup>1</sup> the models produced from the two test sets selected were tested on specific pages. Several errors were observed for each page. The objective of this experiment was to determine the impact of the lexicon on the recognition skills of the model. Although this approach was ultimately inconclusive, a shallow observation of the prediction errors led me to redirect my attention to the n-grams, and I then decided to make those prediction errors the focus of the n-grams experiment.

To facilitate this, the eScriptorium interface<sup>2</sup> allows for an easy retrieval of these errors

---

1. See section 5.2

2. The interface was thoroughly presented in subsection 2.1.2

with compare transcriptions.<sup>3</sup> Only available through the “Transcription” panel of the interface, it provides the opportunity to obtain the elements needed for my experiment.

First, I can choose a base version, i.e. the version to compare to. Here, the base version is the ground truth.

Then, I can select the models I want to compare the ground truth with. For this experiment, I notably reused the models applied during the previous experiment.<sup>4</sup>

Finally, all that was needed was to “toggle the transcription comparison”.<sup>5</sup> Then, I observed the parts in green and red that indicate insertions or deletions in the predictions of the model picked. Once all the errors were retrieved, they were placed in a table.

From it, I obtained the results available in Annex D. While the table still contains the errors present in the prediction of the model War and the model Other, a new model has also been added to the study: the model Ground Truth.

Compared to the 31 (SW) and 76 pages (SO) of training data of the other two sets, the ground truth<sup>6</sup> is much larger, with roughly 500 pages.

In addition to the errors of each model, indicated with their line number, I added a last column. It indicates a possible explanation, for the error, outside the model recognition’s skills, which includes information such as “two-characters overlay”, “manual addition of character”, “light transparency of the paper”, or “problem with the segmentation”.

The next objective is to obtain n-grams from those errors, to understand why the model has been wrong. To do so, errors must first be retrieved token by token, as in sequences of characters separated by punctuation and/or a space. This explains why two error instances in the table have not been transposed to the table of errors. As demonstrated by one of the instances in Figure 7.1, the entire line is incorrect in the models’ prediction, and so completely unintelligible that it is impossible to match it token by token with the ground truth.

---

3. <https://escriptorium.readthedocs.io/en/latest/transcribe/#compare-transcriptions>

4. Those are model War and model Other. See section 3.3

5. Term used in eScriptorium to designate the button that allows to see the various versions of a same line, according to the models selected for comparison. See <https://escriptorium.readthedocs.io/en/latest/transcribe/#compare-transcriptions>

6. See subsection 4.1.1

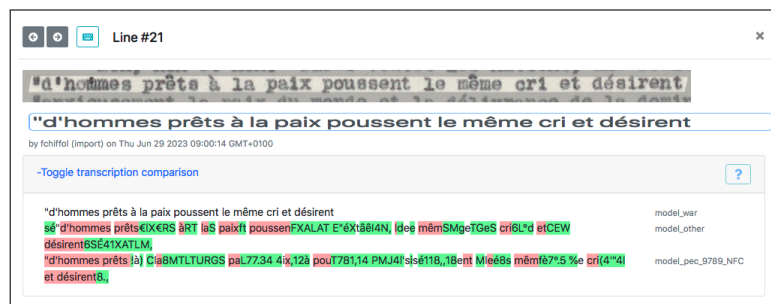


Figure 7.1: Comparison of the model War (MW), Other (MO), and Ground Truth (MGT) to the ground truth of the 21st line of the page 1 of the letter 1367

Finally, among the ten pages and approximately 2,400 tokens constituting the test set, 294 tokens were incorrectly recognized across all models, resulting in 12% of error.

## 7.1.2 The Structure of the Table of Erroneous N-Grams

I have now a list of prediction errors, that needs to become a table of erroneous n-grams. Modifications are needed for transformation, and they will be done automatically by a Python script. In subsection 6.2.2, I mentioned Script P.3.4 used with Python and the module `textwrap`. It produced a list of n-grams from the training data of my test sets, by dividing the tokens into sequences of two, three, or four characters. A modified version of this script has been created to make the table distribution.

### 7.1.2.1 Table Distribution

For this experiment, Script P.3.4 was adapted into Script P.4.1. This time, the list of references and errors of each prediction, instead of the training data, are transformed into the three units of n-grams: 2-grams or bigrams (2 characters), 3-grams or trigrams (3 characters), and 4-grams or tetragrams (4 characters). Each n-gram is represented in a separate table, and in each, there are four parts:

- the correct transcription;
- the prediction of the model War;
- the prediction of the model Other;
- the prediction of the model Ground Truth.

### 7.1.2.2 Table Colours

Each column of prediction is classified in one of the following colours: white, blue, grey, and red.

**White Cells** White is the regular colour, signifying that this token cell is studied in the experiment.

**Blue Cells** On the contrary, a blue cell indicates a token that is ignored in the experiment. The blue cells designate the instances where the size of the token predicted by the model is different from the one in the correct transcription. In this experiment, I want to find reasons behind the error. I do that by studying the difference between the referenced n-gram and the predicted one. In order to do that, the prediction needs to be a mirror image of the correct transcription in its n-gram split.

AL, LE, MA, GN, E	AL, LE, RM, AG, NE
co, mm, is, es	co, mi, se, s

Figure 7.2: Examples of blue cells with insertion(s) (top) or deletion(s) (bottom) in the prediction

In Figure 7.2, the n-grams splits imply that the comparison would have to be made between “MA” and “RM”, “GN” and “AG”, “E” and “NE”, “mm” and “mi”, “is” and “se”, and “es” and “s”. However, the addition or deletion of a character in the middle of the token creates an imbalance in the comparison, which could distort the analysis. It is similar to the Hamming distance that can’t be obtained unless both the ground truth and the prediction have the same length.<sup>7</sup>

**Red Cells** Red cells indicate that the model did not make an error on this token. The table contains predictions from three different models (War, Other, Ground Truth), and the column “Correct transcription” lists the 294 errors from the test set. However, although models sometimes made a mistake on the same token, it is not the norm. In many cases, only one model made an error in the prediction. In those situations, the cells of the other one or two models are red.

<sup>7</sup> See subsection 5.1.1 for more information

**Grey Cells** Finally, grey indicates a condition where the token size matches, but the n-gram size is incorrect. If a cell is grey, it means that the erroneous token has the same number of characters as the correct transcription. However, the error is in an n-gram that is not of the right size for the concerned table. It usually happens for small tokens or in the last part of the token, as shown in Figure 7.3.

nouv, eau	nouv, <b>ebu</b>
CAI, LLA, UX	CAI, LLA, <b>UT</b>
GE, RL, AC, H	GE, RL, AC, N

Figure 7.3: Examples of grey cells for tetragrams (top), trigrams (middle) and bigrams (bottom)

The full content of the tables of erroneous bigrams, trigrams and tetragrams is available in Annex E. Outside the unusable cells, the table contains a wealth of information that I will now explore and expose.

## 7.2 A Diversity of Information About N-Grams

### 7.2.1 What Information About the N-Grams Does the Table Provide?

In the previous section, I presented the general outline of the tables of n-grams, but it can provide even more information than that.

#### 7.2.1.1 Composition of the Table

Although the whole token is provided, with its split of 2, 3 or 4 characters, the only valuable part of the token is the wrong n-gram(s), as I am working on identifying the errors. In order to find it more easily, it has been highlighted in green in the white cell of the prediction column. However, the objective of this experiment is to understand the reasons behind these errors. With the aim to achieve that, I decided to search towards the occurrences in the various training data, which prompted the addition of two columns to the tables. Their cells are filled when the prediction can be studied, because they are there to identify patterns or reasons behind the errors.



### 7.2.1.2 New Table Elements

The columns contain the number of occurrences of the n-gram in the training data, for the reference n-gram and for the incorrectly predicted n-gram.

From there, in each cell, two types of information can be found:

- a number, indicating the occurrence frequency in the training data;
- a  $\emptyset$ , indicating that the n-gram did not exist in the training data.

In some cases, cells have several numbers because some tokens had more than one erroneous n-gram. Additionally, a new colour coding was introduced: a green filling, specifically found in the column “Number of occurrences for the erroneous n-gram”, present when the number of occurrences in the training data is higher for the erroneous n-gram than it is for the correct n-gram.

### 7.2.1.3 Illustrated Examples from the Table

**Figure 7.4** In this figure, taken from the table of bigrams, all models made an error in their prediction.

For the model War and Ground Truth, the error was made on the fifth n-gram of the token, while, for the model Other, it was the fourth.

In all cases, the n-gram, whether it is right or wrong, existed in the training data. However, it is only for the training data of the model War that the erroneous n-gram frequency was higher than the correct n-gram, meaning a more justifiable error.

**Figure 7.5** In this figure, taken from the table of bigrams, the model War and model Ground Truth are not involved in the study. The prediction is too short for the former, with one character missing, making it a blue cell, and the prediction is correct for the latter, making it a red cell.

For the model Other, the prediction was wrong for two n-grams: the third and the fourth. In both cases, both n-grams were present in the training data, but only the fourth n-gram had a higher erroneous prediction frequency than the correct transcription. Therefore, the cell is green, but the first number is highlighted in white, to make the distinction.

**Figure 7.6** In this figure, taken from the table of trigrams, the model Other and model Ground Truth are not involved in the study. The prediction is correct for the former, and

it is too short for the latter, as one character is missing.

For the model War, the prediction was wrong on the first n-gram. Occurrences were found in the training data only for the erroneous n-gram. It puts a  $\emptyset$  on the “Number of occurrences for the correct n-gram” cell, but also an automatic green cell for the “Number of occurrences for the erroneous n-gram”.

**Figure 7.7** In this figure, taken from the table of tetragrams, the model War and model Other are not involved in the study. The prediction is too short for the former, as one character is missing, and the latter is correct.

For the model Ground Truth, the error was on the second n-gram. In this case, the correct n-gram had occurrences in the training data, but not the erroneous one, meaning that no explanation could be found for the prediction in the training data.

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
tr, an, qu, il, li, té	tr, an, qu, il, ai, té	42	135	tr, an, qu, im, li, té	265	67	tr, an, qu, il, bi, té	871	513

Figure 7.4: Example of a line from the table of bigrams where all the predictions had an error

Al, le, ma, gn, e	Al, le, ma, ge			Al, le, sa, gu, e	284 39	217 68			
-------------------	----------------	--	--	-------------------	--------	--------	--	--	--

Figure 7.5: Example of a line from the table of bigrams where one prediction has not the same length (model War), one has several erroneous n-grams (model Other) and one has the right prediction (model Ground Truth)

fal, lu	fai, lu	∅	41				fal, l		
---------	---------	---	----	--	--	--	--------	--	--

Figure 7.6: Example of a line from the table of trigrams where one prediction has an erroneous n-gram (model War), one has the right prediction (model Other) and one has not the same length (model Ground Truth)

touj, ours	touj, ons						touj, orrs	139	∅
------------	-----------	--	--	--	--	--	------------	-----	---

Figure 7.7: Example of a line from the table of tetragrams where one prediction has not the same length (model War), one has the right prediction (model Other) and one has an erroneous n-gram (model Ground Truth)

		4-grams			3-grams			2-grams		
		MW	MO	MGT	MW	MO	MGT	MW	MO	MGT
COLUMNS	CT ≠ M (for the same number of characters)	146	58	59	152	60	63	169	71	68
	char CT ≠ char M	40	21	14	40	21	14	40	21	14
	CT = M	82	200	211	82	200	211	82	200	211
	Not taken into account (error but not the right n-grams)	26	15	10	20	13	6	3	2	1
N-GRAMS	Number of erroneous n-grams	158	63	62	171	66	66	204	84	75
	Occurrences in M = ∅	140	51	46	104	42	32	49	19	7
	Occurrences in M > Occurrences in CT	11	4	10	43	12	17	103	35	34

Figure 7.8: Results by numbers from the Token Error Analysis tables

	4-grams			3-grams			2-grams		
	MW	MO	MGT	MW	MO	MGT	MW	MO	MGT
Tokens correctly recognized	27,89%	68,03%	71,77%	27,89%	68,03%	71,77%	27,89%	68,03%	71,77%
Tokens with simple substitutions	49,66%	19,73%	20,07%	51,70%	20,41%	21,43%	57,48%	24,15%	23,13%
Occurrences where the substitution "makes sense"	6,96%	6,35%	16,13%	25,15%	18,18%	25,76%	50,49%	41,67%	45,33%
Occurrences where the substitution has "no base"	88,61%	80,95%	74,19%	60,82%	63,64%	48,48%	24,02%	22,62%	9,33%

Figure 7.9: Results by percentages from the Token Error Analysis tables

## 7.2.2 What Additional Information Can We Gather from the Table?

The tables offer a large amount of information, especially considering there are three different tables (bigrams, trigrams and tetragrams). Together, they contain 294 tokens presented four times, leading to over 8,000 cells. In addition to what I can learn about the errors and the reasons behind it, by observing the differences in the numbers of occurrences, I thought I would get essential information from statistics about the repartition of results between the n-grams, and the models.

### 7.2.2.1 Table Data Numerical Distribution

The table displayed in Figure 7.8 presents the numerical distribution of the tables of erroneous n-grams. It is divided into multiple columns and lines:

- In terms of columns:
  - The first division concerns the size of n-grams (bigrams, trigrams, or tetragrams);

- The second division concerns the model that produced the prediction (War, Other, and Ground Truth).
- For the lines:
  - The first division is between columns in general;
  - The second division is about the n-grams more specifically, since, as already mentioned, some tokens could be made of several erroneous ones.

For the division “COLUMNS”, the distribution is in four parts. They are the four colours present in the table:

- white (tokens studied);
- blue (not the same length between reference and prediction);
- red (prediction was right);
- grey (not the right n-gram size for the erroneous token).

Considering a prediction is the same, whether it is divided in a sequence of two, three, or four characters, the numbers of blue and red cells remains the same from one n-gram to another.

The white and grey cells are the only one with numbers changing. The number of grey cells decreases while the n-gram become smaller, making the number of white cells increase.

For the division “N-GRAMS”, there are three lines:

- The first line counts the number of erroneous n-grams by unit and model. As I observed it several times, the count is higher than that in the white cell of the previous division, considering the cases of multiple mistakes;
- The second line counts the number of times when the occurrence of an erroneous n-gram equals  $\emptyset$  in the training data;
- The third line counts the number of times when the occurrences of the erroneous n-grams in the training data was higher than that of the correct ones. These are the green cells in the table.

### 7.2.2.2 Table Data Percentage Distribution

Based on these numbers, I decided to calculate some percentages, rendered in the table of Figure 7.9.

Regarding the columns, it is divided the same way as the previous figure. Therefore, it is first unit of n-gram, then prediction model.

For the lines, the first two have been calculated with the same denominator. It is the total number of erroneous tokens (294). The goal is to obtain the percentages of tokens that were correctly recognized (numerator = number of red cells) and of tokens with only substitutions (numerator = number of white cells).

For the last two lines, the numerator and denominator all come from the second part of the first figure, the division "N-GRAMS". The denominator is always the number in the cell of the first line. The numerator is either the number in the cell of the second or third line. The objective is to obtain a percentage for the instances where the substitution could be explained by the occurrences (green cells) or had no possible explanation (cells with  $\emptyset$ ).

## 7.3 Many Results, No Concrete Outcome

### 7.3.1 A First General Idea of the Recognition's Skills of the Models

Firstly, I will observe the statistics comparing the models War (MW), Other (MO), and Ground Truth (MGT).

#### 7.3.1.1 "COLUMNS" Section Analysis

**Accurate, and Erroneous but Not Studiable Recognition** My examination started with the "COLUMNS" part of Figure 7.8, presenting the overall results. First, I observed the numbers in the row "CT = M", which represent instances where the model produced the same token as the reference, i.e. instances of the expected performance of the model. There is a significant difference between what MW can correctly recognize and what the other two models can do. While MW only has 82 tokens correctly recognized, MO has 200 and MGT 211. This entailed that MO and MGT recognized twice more tokens than MW, which proved once again the weak state of MW. By comparing MO and MGT, I

noticed that their recognition level seems equivalent. Yet, the training of MGT was made with way more training data than MO, meaning that further observations will be needed to understand the phenomenon.

This trend of performance disparity is also evident with the numbers in the row “CT  $\neq$  M”, i.e. instances where the model not only provided a poor performance, but its prediction can’t be studied, according to the rules I established here. MO and MGT numbers are consistently close:

- The tetragrams are 58 (MO) and 59 (MGT);
- The trigrams are 60 (MO) and 63 (MGT);
- The bigrams are 71 (MO) and 68 (MGT).

**N-Gram Unfit For Study** Finally, for the instances where the error was not considered because it is not in the right n-gram, there was a decrease, the shorter the n-gram gets:

- There are many that are not considered in tetragrams, with 26 for MW, 15 for MO, and 10 for MGT.
- There are 20 for MW, and a little less for the other trigrams, with 13 for MO and 6 for MGT.
- Then, there are almost none left in bigrams, as MW has 3, MO 2, and MGT 1.

This could signify that some one-character tokens incorrectly recognized remain. It could also mean that there are tokens with an odd number of characters and the one character wrong is at the very end, as demonstrated in Figure 7.3.

### 7.3.1.2 “N-GRAMS” Section Analysis

**Erroneous N-Grams** Next, I turned my attention to the "N-GRAMS" part of the figure, with which I could gather even more information than with the columns. The “Number of erroneous n-grams” row showed that MW is even worse than what was suggested in the columns. It happens no matter the n-gram in question, but mostly with the bigrams. While there are 146 (4-grams), 152 (3-grams) and 169 (2-grams) cells with incorrect outputs, there are actually 158 tetragrams, 171 trigrams and 204 bigrams that are erroneously predicted. Those variations of numbers mean that some tokens have several incorrect n-grams. Those numbers were also a better indication to split the results

between MO and MGT. Although MO had sometimes less incorrect tokens than MGT, the erroneous n-grams showed numbers equal or bigger than MGT. This denoted that MO made more mistakes than MGT:

- The trigrams have 60 (MO) and 63 (MGT) erroneous tokens, but 66 (MO) and 66 (MGT) erroneous n-grams;
- The bigrams have 71 (MO) and 68 (MGT) erroneous tokens, but 84 (MO) and 75 (MGT) erroneous n-grams.

**Groundless Errors** Subsequently, while the first line of the “N-GRAMS” part indicated a possible conclusion about the efficacy of the models, the last two are more an indication about the efficacy’s level of the n-grams. For the null occurrences of the erroneous n-gram in the prediction, I observed a decrease of the numbers. It happens when the n-gram gets shorter, no matter the model:

- For MW, it went from 140 (4-gram), to 104 (3-gram), to 49 (2-gram);
- For MO, it went from 51 (4-gram), to 42 (3-gram), to 19 (2-gram);
- For MGT, it went from 46 (4-gram), to 32 (3-gram), to 7 (2-gram).

The numbers continued to be high with trigrams, but they were considerably less important with the bigrams, compared to the first two. It could induce that there are more various trigrams and bigrams in the models, even if they don’t have a lot of occurrences.

**Justifiable Errors** As to the last line of the table (“Occurrences of the prediction > Occurrences of the reference”), the results are reversed: the number rises as the n-gram gets shorter, reaching 35 (MO) and 34 (MGT) instances with a higher frequency for the prediction. The difference is even bigger for MW, with its number multiplied by almost a hundred (11 in tetragrams to 103 in bigrams). It suggested that the errors with MW seem to come from what it learned, and mostly in the bigrams. However, despite the table attesting that the occurrences are bigger for the erroneous n-grams, the scale’s difference is unknown, as it could have three or thirty more occurrences than the reference n-grams. A detailed observation of the tables would be an excellent method to confirm or infirm this, and will be carried out in the next subsection.



### 7.3.1.3 Percentages Analysis

**General Percentages** Now, analysing Figure 7.9 reveals significant differences in recognition between MW, and MO and MGT. Only one fourth of the tokens from the list have been correctly recognized by MW, and it is almost three-quarter for MGT, and slightly less for MO. As for the substitutions that do not involve the blue and grey elements, the percentages vary significantly across n-grams and models. About half of the MW tokens are errors, whether it is tetragrams or trigrams, and almost three fifth with the bigrams, corroborating my previous observations. For MO and MGT, the results are pretty similar: there is about one fifth for the tetragrams, a little more than one fifth for the trigrams and less than one fourth for the bigrams. However, while the percentages are similar, I remarked that MGT has one more point in trigrams, and one less point in bigrams. This could indicate that MO was more prone to errors than MGT. It could mostly be the case on the tokens not considered before because they didn't fit the unit of n-gram.

**Occurrences Analysis** My focus then shifted to the analysis of the occurrences percentages. I firstly notice that substitutions start to make more sense as the n-gram gets shorter. It could signify that the model learns to recognize the bigrams, but not really the trigrams or the tetragrams:

- The percentage of tetragrams of MO and MW is rather low, with about 6%;
- It begins to rise with the trigrams, with 25% for MW and 18% for MO.
- Afterwards, the percentages even double with the bigrams, as MW has 50% and MO 41%.

This means that for MW, of all substitutions in the predictions, only 6% of them made sense regarding the tetragrams extracted from the token analysis. However, when it came to the bigrams, half of them have rational ground. Even though the progression is not as high, the same could be said about MO. Regarding MGT, the percentages were already pretty good for the tetragrams, with about 16%, which could be plausibly explained by its ground truth quantity. It is much bigger than those of the other two models. The progression reaches a similar percentage than MW with the trigrams (25%). For bigrams, its number is positioned between MO and MW's results (45%). Similarly to the other tables, the scale of difference is unknown between the two columns. It seemed especially

the case with MGT, and it would be interesting, then, to see the numbers. The substitutions also became less groundless as the n-gram shortened:

- 74-88% for tetragrams;
- 48-63% for trigrams;
- 9-24% for bigrams.

The erroneous tetragrams in MW have a percentage of 88%. It mainly indicates a lack of correlation to the results of the token analysis. The number is still more than half for the trigrams (60%), but it decreases considerably for the bigrams (24%). Likewise, MO follows an equivalent path, going from 80% (4-gram), to 63% (3-gram), and finally, to 20% (2-gram). On the contrary, MGT already starts way lower than the other two for the tetragrams (74%), became fewer than half of all for the trigrams (48%), and even decreases to less than 10% for the bigrams. Yet again, this could be due to the quantity of ground truth for MGT, as it would offer more various n-grams, even if the frequency is as low as one or two.

#### 7.3.1.4 Tables Data Distribution Analysis

Before focusing on a detailed observation of the tables, I wanted to observe the overall distribution of the tables. Firstly, I wanted to do a little comparison between the number of cells in red, grey, and blue.

**Model War (MW)** For MW, I see that there are 16 red cells where MO and MGT got it wrong (blue and grey cells included). It represents about one fifth of all the red cells in MW. In those 16, there are 10 red cells where the errors were studied, meaning that no blue cells were included. By contrast, MW has 23 blue cells where the other two models correctly predicted the results. It is about half of all its blue cells, meaning that half of those errors are mistakes made only by this model.

**Model Other (MO)** For MO, I observe that there are 26 red cells where MW and MGT got it wrong. It represents about one eighth of all the red cells in MO. In those 26, there are 18 red cells where the errors were studied. To the contrary, MO has only five blue cells, where the other two models correctly outputted the results. It is a pretty low number compared to MW but also to all the blue cells (21).

**Model Ground Truth (MGT)** For MGT, there are 25 red cells where MO and MW got it wrong. It represents about one tenth of all the red cells in MGT. In those 25, there are 14 red cells where the errors were studied. In comparison, MGT has an even lower number of blue cells than MW. There are only three blue cells where the other two models correctly predicted the results.

**General** Overall, there are also 13 cases where everybody got it wrong (whether it is grey, blue, or white). Among them, there are five cases where the errors did not involve an uneven length of characters between reference and prediction. It is also possible to notice that MW mostly did well on its own text, as this is where most of its red cells are located in the table, and the same goes for MO. MGT, which trained with many parts of the dataset, where the content of MO and MW might have been included, had most of its red cells in the MO part of the table. It means it did pretty well with the text from MO, but was less good for the text from MW.

### 7.3.2 Some More Elements of Response With the Details of the Tables

I will now proceed to a detailed observation of the tables from Annex E, to answer three key questions regarding the n-grams and models:

- When the number of occurrences for an n-gram in the prediction is marked as  $\emptyset$ , is the number of occurrences in the reference also  $\emptyset$ ? If not, how many occurrences in the reference are significant?
- When the cell is green in the column of the prediction, meaning that it has more occurrences than for the reference, what is the scale of the gap, i.e. is there barely any difference between both numbers or are there not even in the same hundred or thousand?
- What is the general range of numbers in the columns of reference, which will now be designated as “CT” and prediction, designated henceforth as “MXX” ?

These questions will guide the analysis of the data presented in the tables.

#### 7.3.2.1 Tetragrams Results

First, I centred my attention on the table of tetragrams, shown in Figures E.23 to E.33.

**Model War (MW)** For the model War (MW) group of columns, to answer the first question, there are 140 cells in “MXX” that have  $\emptyset$  as its value. For those cells, “CT” have a value of  $\emptyset$  in 67 instances. This indicates that for about half of those “MXX” cells, the model did not know the correct n-gram either. Moreover, 63 times, “CT” have 10 or fewer occurrences alongside the 140  $\emptyset$ . This indicates that even when the model knew the correct n-gram, it was not in a sufficient quantity to learn it. Only ten cells have a number that could be considered high against those ‘ $\emptyset$ ’. Except for three with the values of 59, 36 and 21, they barely have more than 20 occurrences.

To answer the second one, there are 11 instances where the occurrences in “MXX” are bigger than in “CT”. The numbers in “MXX” are barely exceeding 10 or 20 occurrences. The cells are still green because, on five occasions, the model did not have the correct n-gram in its training data. On another five occasions, it only had it one time, and it would not have been enough to learn it. There is only one instance where the numbers are a little higher: [7 25].<sup>8</sup> The n-grams are “*fran*” (CT) and “*Fran*” (MXX). It means that the model learned to recognize this n-gram with an uppercase more than with a lowercase. As for the range of the two columns, the lowest is  $\emptyset$  in both cases. The highest is 59 for “CT” and “25” for “MXX”. It showed that the numbers of the model War do not rise very high.

**Model Other (MO)** For the model Other (MO) group of columns, there are 51 cells where “MXX” is equal to  $\emptyset$ . For those cells, “CT” have the same value 20 times and the same amount for 10 or fewer occurrences. This entails that, in majority, the model did not know or barely know either tetragram. There are only 11 instances of higher number. Similarly to MW, there are only very few climbing higher than 20 occurrences. Even then, the numbers are still pretty low (21, 24, 38), except for one cell with an occurrence of 194. MO has significantly fewer instances of green cells. Yet, the four have numbers of bigger occurrences, although the gaps are pretty insignificant (+7, +3, +5), except in one case (+24).

Regarding the columns’ range, the lowest is, yet again,  $\emptyset$  in both cases. The highest is 41 for “MXX”, but 194 for “CT”, with 69 as the second highest and 38 as the third. It indicates that the n-gram with 194 occurrences is surely an exception.

---

8. The format [a b] will be used as of now to present the occurrence numbers from the table, with "a" as the reference cell, and "b" as the prediction cell.

**Model Ground Truth (MGT)** For the model Ground Truth (MGT) group of columns, there are 46 cells with the value  $\emptyset$  in “MXX”. For those cells, “CT” are the same only 13 times. Eleven times, they have 10 or fewer occurrences. To the contrary, there are 22 cells with higher number. This is about half of the “MXX”  $\emptyset$  cells. However, the numbers are not on the same scale as earlier. Among the 22 high numbers, the five highest are 469, 243, 194, 163 and 139. It makes the model errors odd. However, I would need to compare it with the occurrences for trigrams and bigrams for the same cells. It would be an interesting way to check if the tetragrams really do have an impact during recognition. For the cases of the green cells, there are ten instances where the occurrences in “MXX” are bigger than in “CT”. The results are rather diverse. In some cases, the occurrences are pretty low whether it is “CT” or “MXX”, and the gap is minor: [3 6], [ $\emptyset$  1], [1 4], [ $\emptyset$  3]. In other cases, the occurrences in “MXX” are getting higher, as well as the gap. The numbers are as such [18 43], [ $\emptyset$  59] or [5 97], with two of them even having a gap of 80-90, which could explain the erroneous recognition.

Lastly, as for the range of the two columns, the lowest is still  $\emptyset$  in both cases. The highest is 97 for “MXX”, with 86 and 59 as second and third highest respectively. For “CT”, the highest is 469, with 243 and 194 as second and third highest respectively. Those are way bigger numbers than for the two previous models. It demonstrated once more the size difference of the training data of MGT.

### 7.3.2.2 Trigrams Results

Then, I focused my attention on the table of trigrams, shown in Figures E.12 to E.22.

**Model War (MW)** For the MW group of columns, there are 104 cells in “MXX” that have  $\emptyset$  as its value. For those cells, “CT” have  $\emptyset$  as a value 47 times and 10 or fewer occurrences, 49 times. Equally to the tetragrams, the model did not know or barely know either trigrams. There are only eight cells with higher number. Except for two (38 and 36), the others barely exceed 20 occurrences.

Next, there are 43 instances where the occurrences in “MXX” are bigger than in “CT”. For 28 of them, this can be explained by an opposite value in “CT” of  $\emptyset$  or 1. Nevertheless, there are also examples where, even if the cell value in “CT” is  $\emptyset$ , the occurrences in “MXX” are important, such as [2 140], [ $\emptyset$  103], and [11 103]. These gaps, sometimes quite significant, could suggest that trigrams have an impact on the recognition.

When it comes to the range of the two columns, both lowest are  $\emptyset$ . The highest is 140 for

“MXX”, with 103 and 67 as second and third highest respectively. For “CT”, the highest is 130, with 103 and 80 as second and third highest respectively.

**Model Other (MO)** For the MO group of columns, 42 cells of “MXX” columns have the value  $\emptyset$ . Opposite to those, “CT” column has null occurrences 11 times, and 10 or fewer, 20 times. 11 cells have a higher number, with one highly superior, at 246, and then, numbers at 68 or 45. This could explain the recognition’s difficulties of the model.

Alike to its tetragrams, there are few instances (12) where the occurrences in “MXX” are bigger than in “CT”. Except for one specific instance, with "*que*" (45) and "*que*" (286), though the cells are green, the numbers do not rise much.

As for the range of the two columns, the lowest remains  $\emptyset$  for both. The highest is 286 for “MXX”, with 58 and 45 as second and third highest respectively. For “CT”, the highest is 246 with 214 and 96 as second and third highest respectively. It tends to demonstrate that, except one specific case, the majority of errors by the model do not come from the training data. Indeed, the forms have not been learned much or more than the real n-grams.

**Model Ground Truth (MGT)** For the MGT group of columns, there are 32 cells where “MXX” equal  $\emptyset$ . For those cells, the “CT” column has  $\emptyset$  as a value only five times and five times as well for 10 or fewer occurrences. By contrast, more than 20 cells have a higher number, such as 513, 353, and 325. While the number of n-grams with a value of  $\emptyset$  is lower than with tetragrams (-14), the number of errors without rational explanation remains similar. However, a continuity seems to be observed from one n-gram to the other. The five highest numbers of occurrences are from the same tokens as in tetragrams.

Additionally, there are 17 instances with green cells. There is again diversity in the results, with pretty low occurrences and gaps. There are also really high gaps, with even one difference located in the thousand ([19 1744]), which might explain the incorrect recognition. It is likely that it proposed the trigram it saw 1700 times during its training rather than one it only saw 19.

In terms of range, the lowest is  $\emptyset$  in both cases. The highest is 1744 for “MXX”, with 395 and 316 as second and third highest respectively. For “CT”, the highest is 1748, with 513 and 353 as second and third highest respectively. Although the numbers reach very high level, I remark that there are only two cases in the thousand. They are “*que*” (which) and “*ont*” (have in the 3rd person singular present tense), i.e. some very specific n-grams, as

they can exist independently as tokens. The next two are in occurrences of around 500, so about half of them.

### 7.3.2.3 Bigrams Results

Lastly, I concentrated my observations on the table of bigrams, shown in Figures E.1 to E.11.

**Model War (MW)** For the MW group of columns, there are 49 cells in “MXX” that have  $\emptyset$  as its value. For those cells, “CT” have, on 12 occasions,  $\emptyset$  as a value, and, on 25 occasions, 10 or fewer occurrences. There are 12 cells with higher numbers. They vary from large numbers (186, 173, 142) to medium (63, 43, 23).

In more than 100 instances, the occurrences in “MXX” are bigger than in “CT”, with variety once more. Sometimes, the numbers are very low and so are their gaps, such as [ $\emptyset$  2], [7 10], and [24 26]. Other times, the numbers are higher but on both sides, such as [173 200], [116 134], and [107 134]. Finally, there are also times when the numbers are rising. In those cases, the gap is widening, such as [58 365], [24 442], [1 239], [7 265], and [ $\emptyset$  365]. Over those 100 instances, about a third of it concerns green cells with more than 100 occurrences.

Regarding the range of the two columns, the lowest is  $\emptyset$  in both cases. The highest is 442 for “MXX”, with 365 and 271 as second and third highest respectively. For “CT”, the highest is 442 with 302 and 241 as second and third highest respectively.

**Model Other (MO)** For the MO group of columns, there are 19 cells in “MXX” with  $\emptyset$  as its value. For those cells, the “CT” column have no null occurrences this time, but 10 or fewer occurrences, 12 times. Additionally, there are seven cells with higher number. Three have a really high number: 391, 278 and 105. The others are placed between 10 and 19 occurrences.

Then, there are 35 instances where the occurrences in “MXX” are bigger than in “CT”. There are about a dozen of low gaps and numbers, and it is mostly high numbers and wide gaps. Sometimes, the difference is even of 100, 200 or more: [33 162], [113 479], [68 616], [49 580], and [197 1039].

As for the range of the two columns, the lowest is  $\emptyset$  in both cases. The highest is 1039 for “MXX”, with 616 and 532 as second and third highest respectively. There are also 17 cells with 100 occurrences or more, and 13 with 50 to 100 occurrences, for a total of 84

erroneous n-grams. For “CT”, the highest is 1039, with 616 and 411 as second and third highest respectively. Besides, 20 cells have 100 occurrences or more, and five have 50 to 100 occurrences.

**Model Ground Truth (MGT)** For the MGT group of columns, there are only seven cells in “MXX” that equal to  $\emptyset$ . For those cells, “CT” have no  $\emptyset$ , and 10 or fewer occurrences one time. There are various levels of high numbers on five occasions: 6569, 3473, 696, 275 and 91.

On 34 instances, the occurrences in “MXX” are bigger than in “CT”. The gaps vary widely, such as [85 1120], [493 2151], [800 3473], [24 202], [10 3473], and [8 334]. There are rare cases of low numbers: [8 23], [2 11], [2 7]. Some cases also involve an almost inexistent gap: [20 35], [19 24], [58 60]. Then, there are very high numbers with large gaps: [10 3473], [8 334], [18 463], [24 732]. Furthermore, even when the numbers in “CT” are pretty high too, the differences can be observed in more than several thousands: [1420 2388], [800 3473], [593 1185], [500 2453].

Lastly, as for the range of the two columns, yet again, the lowest is  $\emptyset$  in both cases. However, only very few cells have just one or two digits in both cases. There are 36 for “CT”, and 30 for “MXX”. The highest is 3473 for “MXX”, with 3347 and 2608 as second and third highest respectively. Six others are also in the thousands of occurrences. For “CT”, the highest is 6569, with 5894 and 4726 as second and third highest respectively. There are also 13 others in the thousands of occurrences. Those numbers showed that the bigrams, especially with this model, are on a different scale as the others. It also demonstrated that the occurrences exist, whether it is of the correct or incorrect n-gram.

#### 7.3.2.4 General Analysis

**Model War (MW)** To sum up, I could point out that, for MW, the tetragrams barely have any occurrences, whether it is in “CT” or “MXX”. When there are, the numbers are low.

The situation is similar with trigrams in terms of errors with no rational ground. It is not the same with the green cells. There seems to be more changes that make sense. It is proved with good numbers of occurrences in the “MXX” rather than in “CT”. Although, those numbers are still pretty low.

Finally, the bigrams still have many null occurrences in “MXX” and “CT”. However, the numbers on the opposite column are rising. Moreover, there are way more occurrences in



the green cells combined with consequent gaps.

**Model Other (MO)** For MO, the tetragrams are similar to MW. There are barely any occurrences in the incorrect or correct n-grams. The numbers are also rather low, except for one single case. It could be explained by the fact that it is a conjunction.

There are more occurrences in the “CT” rather than “MXX” with the trigrams. The numbers are still very low, meaning that the trigrams might be inconsequential too.

At last, for the bigrams, there are far fewer unknown n-grams and much bigger numbers. When there are errors, it appeared to come from huge disparity in occurrences with sharply increasing numbers. I also observed that almost half of the erroneous n-grams seem justified by the numbers in the columns.

**Model Ground Truth (MGT)** For MGT, the tetragrams do not seem impactful either. They have few high numbers. Few cases of high numbers are against null occurrences in errors. Lastly, few instances are of expectable errors.

The trigrams have more groundless errors. There are some high numbers, and not always against cells with null occurrences. However, there are also high numbers that support the error in recognition. The gaps are wide as well. Therefore, the model seems to already suggest that it is learning from trigrams.

Finally, the bigrams have barely any unknown n-grams, no matter if it is “CT” or “MXX”. There are very high numbers and wide gaps. Similarly to MO, almost half of the erroneous n-grams came from justifiable errors from the models. For the other half, I observed that whether it is in “CT” or “MXX”, there are usually some pretty high numbers of occurrences. Gaps are not always very wide, even when “CT” have more occurrences. The only odd results with MGT are five instances of unjustified errors. Additionally to the fact that the occurrences in “CT” are pretty big, be it trigrams or bigrams, they are low or even null in “MXX”, proposing no reason whatsoever for the mistakes.

### 7.3.2.5 Conclusion

In conclusion, I can suggest that, with a small model like MW, the trigrams and bigrams seem to both have an impact on the recognition. The model appears to rely on a combination of n-grams for predictions. Though, it might not always be enough. The double amount of errors for MW comparing to MO and MGT demonstrated it.

Then, with a medium model like MO, only the bigrams seem impactful. The numbers of

occurrences in trigrams and tetragrams might be too low in comparison.

Finally, with a large model like MGT, the tetragrams, and trigrams might be taken into account. However, it is the bigrams that seemed to be the most important. They had a considerably higher number of occurrences. Furthermore, with that many occurrences, the errors might come from a better knowledge of one form over another comparable, during recognition. When both forms are familiar to the model, confusion may lead to errors.

Although the word-level studies on the lexicon were unsuccessful, the analyses of n-grams yielded some positive insights, though not completely conclusive. Questions remain about the impact of larger n-grams and the factors considered for smaller n-grams. To address this, I chose to work with a dataset with more than one language. I hypothesize that analysing data from multiple languages will clarify these uncertainties and offer more precise insights into the roles of different n-grams.



## PART IV

---

# MULTILINGUALISM: AN ANSWER TO THE N-GRAMS



# A NEW DATASET: MULTILINGUAL DOCUMENTS FROM THE HOLOCAUST

---

The correspondence of Paul d'Estournelles de Constant was a useful corpus that I explored from every angle to confirm or refute my theories. But at a certain point, the redundancy of the topics and the global regularity of the correspondence became too constrained to allow me to fully work on my hypothesis. Therefore, a new and different corpus was needed. Multilingualism, as it would offer various and diverse sequences of characters, seemed a promising solution to verify the effect of the n-grams. So, a corpus of documents from the Holocaust provided a suitable solution. Extracted from a project I have also been working on for several years, those documents offer the extra aspect of having a multitude of languages. Before I start detailing the experiments I conducted on this dataset, I will deliver a detailed overview of the corpus, including its composition and the methodological approach used.

## 8.1 History and Presentation of the Corpus

### 8.1.1 Historical Background

During the 1930s, Hitler came into power in Germany. Aggressively antisemitic and resentful of the perceived humiliation from World War I, he quickly began to create and impose anti-Judaic laws. He also invaded neighbouring countries, which he believed should be part of the Reich, the German territory's name, such as Austria or Czechoslovakia [Wiewiorka 2023]. In 1939, Germany, allied with the Soviet Union, invaded Poland. It prompted a declaration of war by the United Kingdom and France.

The Axis, led by Nazi Germany, was allied to Italy and Japan. The Allies notably counted the United Kingdom. From 1941, the Allies also included the Soviet Union, after the rupture of the alliance between Hitler and Stalin, and the United States of America,

after Japan's attack on Pearl Harbor, in Hawaii. The implication of those two powerhouses in the Allies' side shifted the course of the war and the balance of power. The war spanned many fronts (Europe, Africa, Pacific Ocean, etc.), rapidly becoming a world war, and was the theatre of terrible events, such as the Holocaust.

### 8.1.2 The Holocaust

In 1933, part of the German population was already antagonizing the Jewish population, who were seen as “an enemy, a criminal, and a parasite” [Hilberg et al. 2006b, p. 65]. Restrictive anti-Judaic laws were already in the program of the Nazi party at its foundation in 1920, and this consensus between societal antagonism and Nazis ideas eased their establishment after Hitler's rise to power. At the beginning of Hitler's term, the actions took various forms: a campaign of individual violences and a general boycott against Jews, the abuse and killing of Jews by SS or the interdiction for Jews to create business. This culminated in 1938, with the *Kristallnacht*.<sup>1</sup> This antisemitic pogrom received a pretty adverse reaction from the rest of the world, mostly shocked rather than just indifferent, contrary to what Hitler might have anticipated. It was also the last event of anti-Jewish violence in the German street, before the establishment of proper measures for the “destruction of the European Jews” [Hilberg et al. 2006b]. This process of destruction was elaborated in several steps: expropriation procedures, concentration in the ghettos,<sup>2</sup> then the annihilation of the Jews in two ways, mobile team of killers, and deportation and extermination in special camps. The process was not the work of only one group or organization, but an action from the whole German administrative apparatus [Hilberg et al. 2006c].

After the war, the catastrophic toll of five million deaths, one third of the whole community, was established by the Jews, but no true reaction nor help were given by the Allies powers. They focused more on addressing the aftermath of Germany's defeat than on the plight of the Jewish victims. The Holocaust was eventually recognized through the

---

1. The *Kristallnacht*, or “Night of Broken Glass” (for the shattered glass that lined German streets in the wake of the pogrom), refers to the wave of violent anti-Jewish pogroms which took place on November 9 and 10, 1938, throughout Germany, annexed Austria, and in areas of the Sudetenland in Czechoslovakia recently occupied by German troops. Source: <https://encyclopedia.ushmm.org/content/en/article/kristallnacht>

2. A ghetto is the part of a city in which members of a minority group are concentrated, especially as a result of political, social, legal, religious, environmental or economic pressure. During WWII, this term was used to designate sectors created by the Nazi to segregate and confine Jews. Source: <https://encyclopedia.ushmm.org/content/en/article/ghettos>

collection of documents, publication of books, and the establishment of an official term for the destruction [Hilberg et al. 2006a]. The United States were the first, around the 1970s, to dedicate many activities to the preservation and dissemination of the memory of the Holocaust. There were television shows, conferences, and educational programs. It culminated in a decree in 1978 that created the President’s Commission on the Holocaust. It was later transformed in the American Council of the Holocaust memorial. Its goals were to create a museum and establish research and education programs. Other institutes around the world also devoted their works around the preservation and dissemination of this memory. For example, the YIVO Institute in New York, the Center for Contemporary Jewish Documentation in Paris, the Jewish Historical Institute in Warsaw, and Yad Vashem in Jerusalem, the official authority for everything related to the Holocaust memorial, were part of it [Hilberg et al. 2006a].

### 8.1.3 Working on the Holocaust

Nowadays, many institutions are still dedicated to collect, preserve and work on the archives and documents of the Holocaust. Notable among these are the [Yad Vashem’s International Institute for Holocaust Research \(Israel\)](#), the [United States Holocaust Memorial Museum](#), or the [Mémorial de la Shoah \(France\)](#).

Alongside those diverse entities, an organization was created to dedicate its work to this task as an international collaboration: the European Holocaust Research Infrastructure (EHRI). Bringing together institutions from across Europe, Israel, and the United States, including some mentioned above, the project is “dedicated to the further integration of Holocaust archives and research”.<sup>3</sup> It is done through the activities of several groups, called *work packages*, each with their own mission, whether it is “management”, “training and education”, “virtual access” or “connecting micro archival communities and standards”.<sup>4</sup> As a part of the EHRI project, I am working on the WP10, “Thematic layers across collections” lead by Michal Frankl,<sup>5</sup> from the Masaryk Institute and Archives of the Czech Academy of Sciences. One of the main tasks of this work package is to create, publish and maintain digital editions of thematic collections of Holocaust archives. They are called the EHRI Online Editions.<sup>6</sup>

---

3. <https://www.ehri-project.eu/division-work>

4. <https://www.ehri-project.eu/division-work#Workplan%20EHRI3>

5. <https://www.ehri-project.eu/michal-frankl>

6. <https://www.ehri-project.eu/ehri-online-editions>



### 8.1.4 The EHRI Online Editions

My first foray into those editions was with the “Early Holocaust Testimonies”,<sup>7</sup> created in 2020. I was asked to update, homogenize and try publishing it on another publication platform than Omeka,<sup>8</sup> the one the project currently uses. This new platform of publication uses TEI Publisher<sup>9,10</sup> [Bénière, Chiffolleau, and Scheithauer 2024]. This collection gathers testimonies, written or oral (available after having been transcribed) made by Jewish witnesses and survivors. It recalls the horror they experienced during the war or even before, after the rise to power of Hitler, in Germany, and the beginning of the German occupation, in the invaded countries. The testimonies relate many diverse stories such as the experiences of the ghetto, whether it is [Riga](#), [Lwów](#), [Białystok](#) or [Theresienstadt](#). It also relates murders perpetrated by the Nazis, like the [murder of a brother](#), of an [entire family](#), of [children](#) or even [mass massacre](#).

I continued my work with the EHRI project by exploring various online editions, each offering unique insights into Holocaust documentation. Currently, there are six editions published by the EHRI project. I am only working with the first four, as the last two were not published when I started experimenting on this dataset. The three other online editions are:

- “BeGrenzte Flucht”,<sup>11</sup>
- “Diplomatic Reports”,<sup>12</sup>
- “Von Wien ins Nirgendwo: Die Nisko-Deportationen 1939”.<sup>13</sup>

BeGrenzte Flucht is an edition available in German, made in 2018. It presents documents about “the Austrian refugees on the border with Czechoslovakia in the crisis year of 1938”. There is a diversity of documents : it comes from the [government](#), [aid organizations](#) or the [press](#). It can also be [interviews](#), [personal documents](#), or some [other kinds](#).

Diplomatic Reports is an edition available in English, made in 2021. It presents documents

---

7. <https://early-testimony.ehri-project.eu/>

8. Omeka is a free, flexible, and open source web-publishing platform for the display of library, museum, archives, and scholarly collections and exhibitions: <https://omeka.org/>

9. TEI Publisher is an instant publishing toolbox <https://teipublisher.com/>

10. [https://discholed.huma-num.fr/exist/apps/discholed/index\\_ehri.html?collection=ehri%2Fcorpus](https://discholed.huma-num.fr/exist/apps/discholed/index_ehri.html?collection=ehri%2Fcorpus)

11. <https://begrenzte-flucht.ehri-project.eu/>

12. <https://diplomatic-reports.ehri-project.eu>

13. <https://nisko-transports.ehri-project.eu/>

that “focus on how diplomatic staff reported the persecution and murder of European Jews during World War II”. The reports are not restricted to only one side of the war. Currently, there are reports from four countries: [Denmark](#), [Italy](#), [Japan](#), and the [United States](#).

Lastly, Von Wien ins Nirgendwo: Die Nisko-Deportationen 1939 is an edition available in German, made in 2023. This thematic collection focuses on a specific time of the Nazi regime. Before establishing the various concentration and extermination camps, the Nazis planned to “deport Jews to conquered areas”. In “the second half of October 1939, the Central Office for Jewish Emigration sent 1,500 Jewish men from Vienna to Nisko am San”. This was ultimately an inconclusive experiment. Ultimately, many Jews were subsequently chased away and sent to many other places, where no traces are found. In Nisko edition, there are various documents from multiple countries, lacking a specific organizational structure, contrary to the previous editions.

Those editions, by their large sizes, their similarities of subject (the Holocaust) and of writing (typewritten), and their diversities of structure (testimonies, reports, correspondence) and languages (German, English, Polish, etc.), constitutes a compelling source. From it, I am able to select various documents to create a multilingual dataset. Yet, it is imperative that I take into consideration several elements.

## 8.2 A Multilingual Dataset

### 8.2.1 Choosing the Right Script

"Writing is defined as a system of more or less permanent marks used to represent an utterance in such a way that it can be recovered more or less exactly without the intervention of the utterer [...] Half a dozen fundamentally different types of writing systems have been devised with respect to how symbols relate to the sounds of languages." [Daniels and Bright 2010, pp. 3–4]

The *abjad*, mostly found with Arabic and Hebrew, only represents consonants [Daniels and Bright 2010, p. 4]. The *abugida*, found with Ethiopic and North and South Indic, is written as units of consonant-vowel sequences [Daniels and Bright 2010, p. 4]. The *alphabetical*, found with Greek, Latin, or Cyrillic, is a “set of letters written to represent particular sounds in a spoken language” [Daniels and Bright 2010, p. 4]. The *logographic*, found with Chinese characters, is “a written character that represents a semantic component of a language, such as a word or morpheme” [Daniels and Bright 2010, p. 4]. The *syllabary*, found with Cherokee, has written symbols that represent the syllables [Daniels and Bright

2010, p. 4]. Finally, there are also cases of languages that have hybrids systems, i.e., combination of multiple types of writing systems, such as the Japanese [Daniels and Bright 2010, p. 209] or the Hankul (Korean language) [Daniels and Bright 2010, p. 219].

This diversity of writing systems is interesting, culturally speaking. However, it becomes a complication when working on ATR, because in order for a text recognition model to be able to function correctly, it is essential that some specific aspects inherent to the training data are thoughtfully considered. One of them is to have the same script<sup>14</sup> for every image in the set of ground truth. Indeed, training to recognize lines of text is already a complicated task. It would become impossible if we start to propose text where the author used an *alphabetical* system, with letters distinctively recognizable, then an *abjad* system, with symbols only representing the consonants. A situation as such would only create confusion during the model training, and would make it fail to learn anything. Therefore, while doing a multilingual dataset is a viable option, maintaining a consistent script across languages is essential for effective model training.

### 8.2.2 Same Script, Different Languages

Creating a multilingual dataset for text recognition involved two main requirements:

- The documents had to be from the EHRI Online editions, with ideally each of the four editions present at least once in the dataset.
- The multilingual dataset needed to contain documents in several languages, to make a multilingual dataset. However, regarding what was mentioned previously, it is important to ensure that they have the same script.

Regarding those requirements, the Yiddish, highly present in the testimonies, have been instantly discarded. Not only the script is different from the others documents, the writing system is not the same either. Instead of being *alphabetical*, it is written with the Hebrew alphabet, using the *abjad* writing system [Daniels and Bright 2010, pp. 735, 743]. The same goes for the Japanese, present in the diplomatic reports. This language is of a different writing system as the Yiddish and the rest of the documents. Also, it is even a mix of two writing systems, the *logographic* kanjis and the *syllabary* kana [Daniels and Bright 2010, p. 209]. It is totally incompatible with creating a functioning multilingual model.

---

14. Writing system comprised of a set of symbols

Then, except for those two languages, the rest of the editions are written with the Roman alphabet.<sup>15</sup>

As the objective is to obtain a model capable of recognizing various texts, it is also imperative that there is a minimum of material for each language selected, to produce something constructive. All told, but in exclusion of Yiddish and Japanese, there are ten languages across the editions: Czech, Danish, Dutch, English, French, German, Hungarian, Italian, Polish, and Slovak. Only seven of those have been selected for the multilingual dataset. The three languages discarded were Dutch, French, and Italian, due to insufficient quality or quantity of documents, which affected their suitability for training the model. The documents in Dutch and French couldn't be used because there was only one document of barely four pages of each, meaning too few to produce something viable. The Italian documents are in a bigger quantity, as there are nineteen, with usually one to three pages. Yet, it is not the quantity, but the quality that prevented me from using it. The majority of the images were either too light or too noisy. When too light, it created a superposition of text between several pages. When too noisy, it was due to some various stamps put on the images or to multiple lines underlined with a red pencil crayon.

So, the multilingual dataset is made of seven languages. Yet, the quantity is not the same for each language, as observed with Table 8.1.

Language	Collection	Documents	Lines
German	BF; Nisko; EHT	56	2287
English	BF; EHT; DR	54	1989
Czech	BF; EHT	46	1713
Danish	DR	36	1007
Hungarian	EHT	30	1334
Polish	EHT	15	468
Slovak	BF	15	395
Multilingual	BF; Nisko; DR; EHT	252	9193

Table 8.1: Distribution of the EHRI training data

Three languages (Czech, English, and German) are heavily represented, mostly due to the fact that they are found in multiple EHRI Online Editions. Then, the four remaining languages appear in only one of the four collections. Their quantity seems enough (15 to 36

15. This is another denomination for the Latin script.

pages) to learn from it during the model training. In summary, the multilingual dataset comprises seven languages with varying quantities, in the Roman alphabet, i.e. 400 to 2200 lines of ground truth, with a total of 9193 lines, which supported the development of a robust text recognition model.

### 8.2.3 Several Languages, Various Diacritics

Creating a multilingual dataset aims to see how much the languages specificities impact the recognition of the model. As previously mentioned, the languages chosen for the multilingual dataset are all written with the Latin alphabet. It means that they all use the same basic characters. However, one element creates a difference between each: the diacritic. A diacritic, also called diacritical mark/point/sign or accent, is a

"mark above, through, or below letters [...] used in many orthographies to remedy the shortcomings of the ordinary Latin alphabet." [Wells 2000]

Except for the English alphabet, which have no diacritic, every other languages of the dataset have at least one. Some are also pretty close in terms of alphabet because they usually have the same diacritics. [Wells 2000], in their article, makes a pretty thorough list of the various diacritics created by and for the languages using the Latin alphabet:

- The German orthography uses the *umlaut*, two dots placed over the letter (ä, ö, ü);
- The Danish alphabet uses the *o-slash* (ø) and the *a-overring* (å);
- The Hungarian alphabet uses the *umlaut* (ö, ü), as well as the *acute* (á, é, í, ó, ú) and *double acute* accent (ő, ű), that last one being unique to this language;
- The Czech alphabet uses the *acute* (á, é, í, ó, ú, ý), *caron* (č, ď, ě, ň, ř, š, ť, ž), and the *ring* in one specific case (ů);
- The Slovak alphabet uses the *acute* (á, é, í, ó, ú, ý, ĺ, ŕ), *caron* (č, ď, ĺ, ň, š, ť, ž, dž), *umlaut* (ä) and *circumflex* accent (ô);
- The Polish alphabet uses the *acute* accent (ć, í, ó, ś, ź), the *overdot* (ż), the *ogonek* (ą, ę) and the *stroke* (ł).

In order to fully know my multilingual dataset and its variants, I used a Python script, to analyse it and provide a list of all the characters present in the dataset. In addition

to the rendition of the alphabet distribution, it revealed that most diacritics are well-represented, with only minor omissions in Danish, Czech, and Slovak. Moreover, many of them are also available in their uppercase variants in the ground truth, which could prove to be advantageous for the model training. Some instances of those diacritics can be observed in Figure 8.1, that contains documents for the EHRI Online Editions in various languages.

## 8.2.4 Training a Multilingual Text Recognition Model

After gathering those ground truths, I started the training to obtain a multilingual text recognition model. I used it for the experiments I will subsequently present. I also intend to use it for the transcription of the next EHRI online editions, according to its accuracy and efficiency. The model was trained using Kraken<sup>16</sup> with the Command-Line Interface (CLI). It was done on an outside server to have access to a GPU,<sup>17</sup> as it is needed when training is done on a significant quantity of data, such as the nine thousand lines and more of text from the multilingual training data. About twenty-five epochs<sup>18</sup> were necessary to properly trained the model. A model of 97.2% of accuracy, which is a rather promising result, was produced.<sup>19</sup> Although no use is made of them afterwards, I also worked on developing single-language text recognition model, for each of the language of the dataset. They were ranging from about 93% of accuracy (the shortest language datasets) to 97% (the largest language datasets).<sup>20</sup> With extensive and varied training data and a new, rather accurate, model, I was then able to retake the steps done with my previous dataset, to verify the hypothesis of the impact of the n-grams I was working on.

---

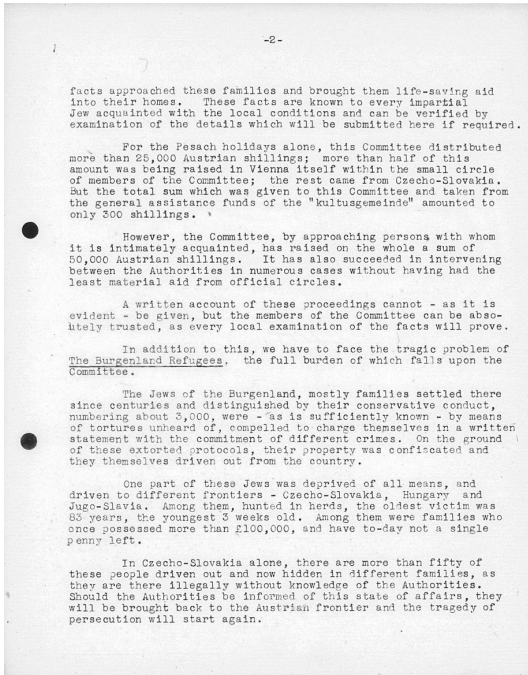
16. Kraken is the recognition software used during this thesis. See subsection 2.1.2

17. A GPU, or Graphics Processing Unit, is a specialized electronic circuit designed to accelerate calculations

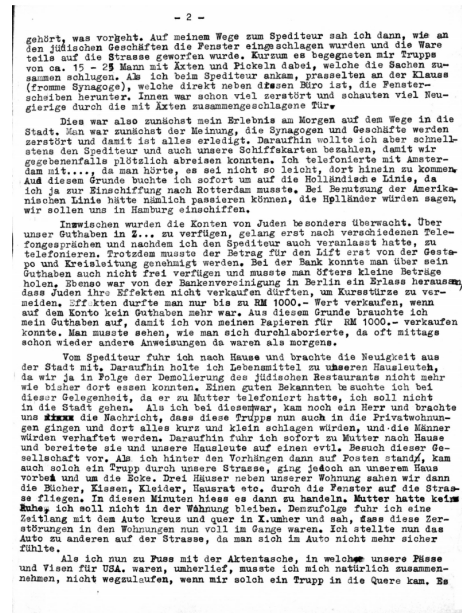
18. An epoch is a cycle during which training, testing and validating will be done to progress into the production of an accurate model

19. [https://github.com/FloChiff/ehri-dataset/blob/main/models/ehri\\_nfd\\_9720.mlmodel](https://github.com/FloChiff/ehri-dataset/blob/main/models/ehri_nfd_9720.mlmodel)

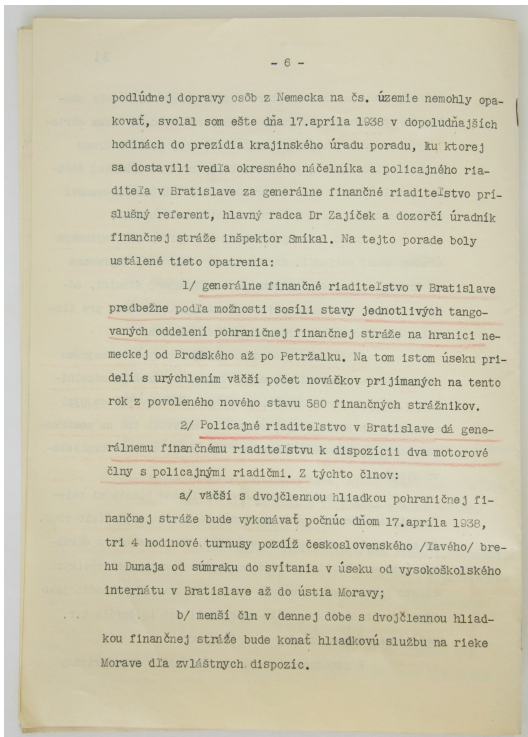
20. <https://github.com/FloChiff/ehri-dataset/tree/main/models>



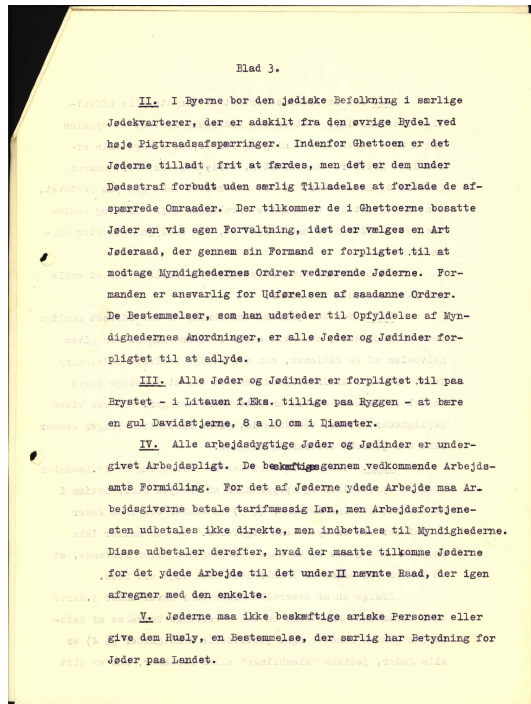
(a) English document with no diacritical sign



(b) German document with some characters with umlaut



(c) Slovak document with many characters with caron or acute accents



(d) Danish document with many o-slash characters

Figure 8.1: Images from the EHRI dataset

# ANALYSING HOW THE MULTILINGUAL MODEL COOPERATES

---

A new dataset means a new source to analyse, utilize, and investigate. Therefore, before evaluating the effect of the variety of n-grams on the text recognition of the multilingual model, gaining a thorough understanding of this dataset is crucial. However, a different dataset implies different aspects to take into consideration.

The dataset composition and distribution is still the core of this study, but the scale has changed, as I am now working with multilingualism, so I aimed to understand how it was rendered in the dataset.

To address this issue, this chapter is dedicated to exhibiting what was observed in the multilingual dataset, first at the character-level, then at the n-gram level.

## 9.1 Learning About the Alphabet of the Multilingual Dataset

### 9.1.1 Obtaining Insight into the Alphabet

To understand how the multilingual model cooperates, I decided to conduct a token analysis, for which I followed the same methods used and presented in the chapter 6. However, I am working with a multilingual dataset, which have its own peculiarities. For this reason, I chose to perform an alphabet analysis first. While the languages are all in Latin script, they all have some unique diacritics, and I want to explore those unique diacritics before focusing on the n-grams. Therefore, I want to obtain the distribution of the alphabet in the training data. I also want their occurrences, in total but also by language.

In order to do that, I used Script P.5.1. It contains a single function that finds every character used in each language set, and counts their occurrences. After obtaining the data,



I compiled a table with eight columns, one for the characters and one for each language. However, the table required cleaning. The script retrieved every element present in the text of the dataset, so it also included data irrelevant to this analysis, such as punctuation and numbers, which were meaningless items for the study, as I am only focusing on the alphabet distribution of my dataset.

After cleaning, the table includes 108 unique characters. A zero was placed in cells where no occurrences were found for a given character. In the end, I created two totals:

- A total presents the sum of occurrences of the characters across languages;
- Another provides the full count of characters for each language set.

An example of the top, middle, and bottom of this table, with both totals, is illustrated in Figure 9.1.

After obtaining the character distribution data, I reordered the table to enhance readability and added colouring for better visualization, as shown in Annex F.

First, I made a table that presents the general distribution of the alphabet, that is available in Figures F.9 to F.11:

- The columns render the language;
- The rows represent individual characters, and are classified from the highest number of occurrences to the lowest number;
- In each cell is provided the number of occurrences of the character in the language of the column;
- The last row of the table gives the total of characters for each language;
- The last column gives the total of occurrences of the character of the row.

Subsequently, to make patterns in character occurrence more apparent, I applied a heatmap visualisation, which represents the magnitude of individual values within a dataset as a colour. The goal was to observe distinctly the character distribution, while identifying any anomalies or patterns in it. The colouring was applied to the whole table with no distinction between columns: the redder the cell, the higher the occurrence number.

For example, in Figure 9.2a, the character *e*, ranked at the top of the table, has all its cells red. Its occurrences are high, no matter the language.

Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
e	18264	11443	6533	6226	6735	1475	1162	51838
a	6699	6430	5512	4753	2638	1961	1263	29256
t	6221	7721	4079	5988	2546	752	601	27908
i	8430	5744	3630	2174	2388	1939	791	25096
r	7851	5544	3247	2353	3503	949	749	24196
s	7052	5296	3819	2839	2316	782	675	22779
o	2683	6499	6206	2489	1642	1488	1297	22304
l	3480	3064	3897	3452	2061	507	565	17026
n	154	5832	5169	59	2903	1077	979	16173
N	11745	115	124	4034	77	55	18	16168

(a) First ten characters

L	273	85	56	34	85	13	6	552
V	195	22	210	42	61	0	22	552
ø	0	0	0	0	536	0	0	536
W	335	103	5	17	6	56	8	530
†	0	0	0	0	0	525	0	525
ő	0	0	0	485	0	0	0	485
ä	384	1	25	8	7	0	14	439
O	69	84	83	42	91	38	15	422
ú	0	0	104	153	0	0	146	403
æ	0	0	0	0	397	0	0	397

(b) Ten characters from the middle

Ś	0	0	0	0	0	5	0	5
Ø	0	0	0	0	4	0	0	4
Ř	0	0	4	0	0	0	0	4
ê	2	0	0	0	0	0	0	2
à	0	0	0	0	1	0	0	1
ă	0	0	1	0	0	0	0	1
ë	0	0	0	1	0	0	0	1
Ň	0	0	1	0	0	0	0	1
Ó	0	0	0	1	0	0	0	1
ß	1	0	0	0	0	0	0	1
Total	113176	86205	81980	60425	40665	21475	14364	418290

(c) Last ten characters

Figure 9.1: Excerpt from the general distribution of the alphabet

Secondly, after examining the general distribution, I further modified the table to focus on the distribution of the alphabet by characters. The result can be seen in Figures F.1 to F.4. Once again, the rows of the table are classified from the highest number of occurrences of a character, all languages combined, to the lowest number. Then, in each cell, there is a percentage: the result of the occurrence's number of the character in the language of the column, divided by the sum of occurrences of the character across the languages.

For example, the character *e* appears 18,264 times in the German part of the dataset. In the full dataset, it appears 51,838, so this division is equal to 0.3523, which makes a percentage of 35.23%, as seen in Figure 9.2b.

Finally, this table was transformed, as well, into a heatmap: the redder the cell, the highest the percentage. The goal here was to observe more easily the character's distribution between languages.

Lastly, as a mean to learn about the distribution in regard to the language, I created the table of distribution of the alphabet by language. It is rendered in Figures F.5 to F.8:

- The rows are ordered alphabetically;
- The diacritic versions of the character are available after the plain ones;
- For each column of language, the number of occurrences of the character in the language has been divided by the total number of characters in the language set, and was then transformed into percentages.

To take the same example as before, the character *e* in the German subset has the same numerator, 18,264, but now, the denominator is 113,176. This makes 0.1613769, which becomes a percentage of 16.1377%, as illustrated in Figure 9.2c. The percentages have been calculated to the fourth decimal after the point, because some percentages are so low that two or three decimals would not give enough information.

Finally, this table was switched into a heatmap. Contrary to the previous two, the colouring is not to be observed in the whole table, but by column, as in the language. Consequently, in a column, the redder the cell, the more present the character is in the language set. Equivalent percentages in other columns may not appear with the same colour intensity due to variations in character distribution across languages.

Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
e	18264	11443	6533	6226	6735	1475	1162	51838

(a) General distribution

e	35,23%	22,07%	12,60%	12,01%	12,99%	2,85%	2,24%	51838
---	--------	--------	--------	--------	--------	-------	-------	-------

(b) Distribution by characters

e	16,1377%	13,2742%	7,9690%	10,3037%	16,5622%	6,8685%	8,0897%
---	----------	----------	---------	----------	----------	---------	---------

(c) Distribution by language

Figure 9.2: The character *e* in each table

## 9.1.2 An Uneven Distribution of the Alphabet

The alphabet heatmaps provide valuable insights into character distribution and frequency across languages. The tables complement one another, offering a more comprehensive view of the data.

First, in the general distribution of the alphabet, I noticed that the first rows are mostly made of common letters across every language. The character distribution is relatively consistent across languages, but with some notable irregularities:

- For example, the *v* has many occurrences in Czech and Hungarian, but is inexistent in Polish;
- The *N* appears frequently but is concentrated in German and, to a lesser extent, Hungarian;
- Additionally, it seems to be an exception as for the presence of uppercase characters at the top of the table. With the exception of *N*, *D*, and *H*, most uppercase letters are ranked toward the bottom of the table.
- At the very bottom, the uppercase letters found are characters that are not among the most common in the languages selected, like *X*, *Q* or *Y*. Their presence there can be explained by the fact that they are found in low number and often in only one language.

While uppercase letters show some irregularities, the most significant anomaly lies in the distribution of diacritics. Those diacritics can have relatively high frequency, shown in light red, with 150 occurrences or more. However, they are located at the bottom of the

table, meaning that those occurrences are present in only one language, which also makes their total, hence the ranking.

More information can be supplemented to those observations by looking at the distribution of the alphabet by characters or by languages. Those tables provide a better understanding of the distribution of the letters in the dataset. The table of the distribution of the alphabet by characters is ranked like the general distribution of the alphabet. Its top really shows, this time, the disparity in data quantity between the datasets. The Polish and Slovak have barely any red or in an extremely light intensity in their columns, except for the cells where they are single language occurrences or for diacritics common to Czech, for Slovak.

The distribution of the alphabet by languages brings a more accurate distribution regarding the languages. Therefore, I observed that, in the general distribution of the alphabet, the characters at the top were very red, but the distribution becomes less homogenous when normalized by the total occurrences in each language. The intensity of red is not the same from one language to another. The characters that appear to be the most common seem to be more distributed in Czech and Hungarian. The Polish and Slovak, with their small dataset, also happen to have a good distribution. Although, most characters make up only about 3% of the whole language set when their presence is not very high. From the distribution of the alphabet by characters, I can confirm some of those observations:

- The Czech seems to have almost every shade of red. It indicates that the language set has a quite diverse alphabet. It also has almost all the diacritics of the entire dataset in it, as well as some unique too;
- The German, similarly to Czech, has a lot of red, in very variegated shades;
- The English is mostly present at the top of the table, and gradually disappears as you move to the middle and lower sections of the table. This is likely because the end of the table is mainly diacritics, which is an element absent in the English alphabet;
- The Hungarian and Danish are similar to English. One exception is that they have some unique diacritics in their sets;
- The Polish does not have many high percentages in the table. When it does, it is a bright red cell, indicating a 100%.

Finally, the diacritics might be the most striking elements in those tables. They deliver contradictory information from one table to another. In the distribution of the alphabet

by characters, there are many bright red cells, but many cases are of all occurrences of a character in one single language or in only two languages. For 108 characters, 30 have 100% of the occurrences, and 7 have more than 90% of the occurrences. It means that more than 30% of the table is made by characters found almost entirely in one language, which suggests that those characters are of great deal.

However, the distribution of the alphabet by languages adds some perspective, and sometimes provides a stark contrast. As seen in Figure 9.3, for the character  $\acute{z}$ , a 100% in the previous table became a 0.1723% in this table, and for  $\ddot{O}$ , it became a 0.0132%. This suggests that while certain diacritics are exclusive to specific languages, their overall frequency is low, implying that they may not significantly influence the model's performance.

$\acute{z}$	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	37
$\ddot{O}$	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	8

(a) Distribution by characters

$\acute{z}$	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,1723%	0,0000%
$\ddot{O}$	0,0000%	0,0000%	0,0000%	0,0132%	0,0000%	0,0000%	0,0000%

(b) Distribution by language

Figure 9.3: Difference of percentage for the characters  $\acute{z}$  and  $\ddot{O}$ 

To conclude, I can point out that, despite the model being trained from a multilingual dataset, there are many similarities between the languages, which could help with the recognition.

There are two problems that could really occur. The first one is common and has been observed with the previous dataset: the uppercases. There are not many in the dataset, and it could create issues during the recognition. The second one, and probably the most important in this case, is that there are a wide diversity of diacritics present in the alphabet, with a quite irregular distribution, which could be problematic.

## 9.2 Exploring the Interactions of the N-Grams of the Multilingual Dataset

### 9.2.1 A Study of the N-Grams, Notably the Least and Most Popular

After conducting a detailed analysis of the alphabet, I am now focusing on the n-grams of the dataset. The study will concentrate on the least and most popular n-grams, but I am first retrieving all the n-grams of the language sets. In order to do so, I followed the same steps I did during the token analysis of Paul d'Estournelles' dataset.<sup>1</sup> I altered the scripts I used to adapt them to this new dataset. I used Script P.5.5, which contains the tokenized version of each language set, and I retrieved the bigrams, trigrams and tetragrams from the training data. In this [Python file of results](#), the lists produced from the script and divided by sets can be found:

- At lines 55 (tokens), 61 (4-grams), 73 (3-grams) and 85 (2-grams) for Czech;
- At lines 98 (tokens), 104 (4-grams), 122 (3-grams) and 128 (2-grams) for Danish;
- At lines 141 (tokens), 147 (4-grams), 159 (3-grams) and 171 (2-grams) for English;
- At lines 184 (tokens), 190 (4-grams), 202 (3-grams) and 214 (2-grams) for German;
- At lines 227 (tokens), 233 (4-grams), 245 (3-grams) and 257 (2-grams) for Hungarian;
- At lines 270 (tokens), 276 (4-grams), 288 (3-grams) and 300 (2-grams) for Polish;
- At lines 313 (tokens), 319 (4-grams), 331 (3-grams) and 343 (2-grams) for Slovak;
- At lines 12 (tokens), 18 (4-grams), 30 (3-grams) and 42 (2-grams) for the full set EHRI.

Next, I used Script P.5.4, which retrieved the occurrences of each element of those lists. The dictionaries are available on the same file as before and can be found:

- At lines 58 (tokens), 64 (4-grams), 76 (3-grams) and 88 (2-grams) for Czech;
- At lines 101 (tokens), 107 (4-grams), 119 (3-grams) and 131 (2-grams) for Danish;

---

1. See section 6.3

- At lines 144 (tokens), 150 (4-grams), 162 (3-grams) and 174 (2-grams) for English;
- At lines 187 (tokens), 193 (4-grams), 205 (3-grams) and 217 (2-grams) for German;
- At lines 230 (tokens), 236 (4-grams), 248 (3-grams) and 260 (2-grams) for Hungarian;
- At lines 273 (tokens), 279 (4-grams), 291 (3-grams) and 303 (2-grams) for Polish;
- At lines 316 (tokens), 322 (4-grams), 334 (3-grams) and 346 (2-grams) for Slovak;
- Lastly, at the lines 15 (tokens), 21 (4-grams), 33 (3-grams) and 45 (2-grams) for the full set EHRI.

The number of occurrences in each of those lists and dictionaries of raw, unclean data is available in Tables 9.1 and 9.2.

Language	Tokens	Tetragrams	Trigrams	Bigrams
Czech	19680	30804	37190	49166
Danish	9760	14531	17451	24122
English	22803	30727	37399	50455
German	25905	38079	45671	65180
Hungarian	14415	21904	26342	35692
Polish	5491	8100	9652	12911
Slovak	3715	5425	6439	8710
Full dataset	101769	149570	180144	246236

Table 9.1: Occurrences of the raw lists created from the sets

Languages	Tokens	Tetragrams	Trigrams	Bigrams
Czech	6627	6167	4338	1269
Danish	2792	3024	2175	836
English	4106	3799	2562	843
German	5630	5025	3227	1001
Hungarian	4450	4541	3231	1096
Polish	2066	2316	1914	761
Slovak	1545	1797	1630	802
Full dataset	25735	21315	11125	2557

Table 9.2: Occurrences of the raw dictionaries created from the sets



Similarly to the previous token analysis, those dictionaries were full of irrelevant and noisy data. So, I proceed to prune all of that, which produced [clean lists](#) of n-grams. There is one for each unit, and there are available either by language or for the full dataset.

From those dictionaries, I then collected the most and least popular n-grams. I abided by the same rules I chose earlier:<sup>2</sup>

- The least popular n-grams have only one occurrence;
- The most popular n-grams have 11 or more occurrences.

Those lists of n-grams are available in the same [Python file of results](#) that was mentioned before:

- The least popular n-grams can be found:
  - At the lines 70 (4-grams), 82 (3-grams) and 94 (2-grams) for the set Czech;
  - At the lines 113 (4-grams), 125 (3-grams) and 128 (2-grams) for the set Danish;
  - At the lines 156 (4-grams), 168 (3-grams) and 180 (2-grams) for the set English;
  - At the lines 199 (4-grams), 211 (3-grams) and 223 (2-grams) for the set German;
  - At the lines 242 (4-grams), 254 (3-grams) and 266 (2-grams) for the set Hungarian;
  - At the lines 285 (4-grams), 297 (3-grams) and 309 (2-grams) for the set Polish;
  - At the lines 328 (4-grams), 340 (3-grams) and 352 (2-grams) for the set Slovak;
  - At the lines 27 (4-grams), 39 (3-grams) and 51 (2-grams) for the full set EHRI.
- The most popular n-grams can be found:
  - At the lines 67 (4-grams), 79 (3-grams) and 91 (2-grams) for the set Czech;
  - At the lines 110 (4-grams), 122 (3-grams) and 134 (2-grams) for the set Danish;
  - At the lines 153 (4-grams), 165 (3-grams) and 177 (2-grams) for the set English;
  - At the lines 196 (4-grams), 208 (3-grams) and 220 (2-grams) for the set German;
  - At the lines 239 (4-grams), 251 (3-grams) and 263 (2-grams) for the set Hungarian;

---

2. See subsection 6.2.3

- At the lines 232 (4-grams), 294 (3-grams) and 306 (2-grams) for the set Polish;
- At the lines 325 (4-grams), 337 (3-grams) and 349 (2-grams) for the set Slovak;
- At the lines 24 (4-grams), 36 (3-grams) and 48 (2-grams) for the full set EHRI.

From those lists of least and most popular n-grams, I conducted an examination of how much the languages have in common with each other. To proceed, I used the same method as for the content analysis.<sup>3</sup> However, I only used `intersection()` this time, since I am only interested in the common elements. With Script P.5.3, I produced 126 lists:

- There are 63 lists for [least popular tokens](#);
- There are 63 lists for [most popular tokens](#);
- Each language is compared to the six others;
- The comparison is done for each unit of n-grams.

Now, as to easily study the results obtained, I retrieved the length of each list, and created a heatmaps visualisation, shown in Figure 9.4. The heatmaps are created as such:

- There are three tables, one for each unit of n-grams;
- Each table is divided in two parts:
  - On the bottom left, in red, are the most popular tokens;
  - On the top right, in green, are the least popular tokens;
- Those tables are cross tabulation. The rows and columns have the same headers, which is why there are seven black cells in the table, as no comparison can be done within the same language.
- The number, in each cell, indicates how many n-grams the two languages have in common.
- Then, shades of red (most popular) or green (least popular) were added, and the intensity of colour implies the rising of the occurrences.

---

3. See section 4.2

	Hungarian	English	German	Slovak	Polish	Czech	Danish
Hungarian		10	13	16	13	13	12
English	177		11	7	5	10	8
German	193	236		7	10	11	9
Slovak	99	92	98		14	13	5
Polish	111	123	129	80		3	6
Czech	207	195	220	145	136		7
Danish	152	181	197	89	95	173	

(a) Bigrams

	Hungarian	English	German	Slovak	Polish	Czech	Danish
Hungarian		45	55	42	42	76	45
English	61		48	37	37	52	52
German	66	140		38	42	72	75
Slovak	4	12	12		51	63	30
Polish	7	14	16	8		56	33
Czech	47	75	78	35	26		61
Danish	42	85	99	7	7	47	

(b) Trigrams

	Hungarian	English	German	Slovak	Polish	Czech	Danish
Hungarian		32	40	12	13	41	18
English	6		70	13	15	64	54
German	4	16		16	25	79	74
Slovak	0	2	1		24	84	21
Polish	0	1	0	0		48	20
Czech	3	10	6	5	5		52
Danish	1	10	9	0	0	3	

 Most popular       Least popular

(c) Tetragrams

Figure 9.4: Common n-grams between two languages

## 9.2.2 Exploring the Similarities and Differences Between Languages

The distribution of numbers is inverted in the table. For the most popular parts, the numbers go from low, in tetragrams, to high, in bigrams. This is switched for the least popular, although, the numbers stay really low in the least popular parts.

### 9.2.2.1 Results for the Most Popular N-Grams

For the most popular parts of the tables, first, I notice that the tetragrams are not really useful, as their numbers are insignificant and mostly zero.

Once again, the small size of the Polish and Slovak sets became visible. They have barely any common occurrences with the others languages in tetragrams and trigrams. Their numbers increase slightly for the bigrams, but are still pretty weak, for the Slovak mainly.

The English and German sets always have the highest common number, no matter the unit of n-gram.

The Czech set appears to have an affinity with every language. It is even stronger than English and German at times, despite having a smaller dataset.

Additionally, a huge discrepancy between the minimum and maximum of this part of the table can be observed. In the bigrams, English-German has the highest common number with 236, and Polish-Slovak has the lowest with 80. For the trigrams, English-German is the highest with 140, and Hungarian-Slovak the lowest with 4. Lastly, for the tetragrams, English-German has, yet again, the highest with 16, and several couples are at zero for the lowest, such as Polish-Slovak, German-Polish, and Hungarian-Slovak.

The disparity between the English set and the Slovak set seems to be the more obvious. They are often found in the lowest number of common occurrences: 92 for bigrams, 8 for trigrams, and 2 for tetragrams.

### 9.2.2.2 Results for the Least Popular N-Grams

For the least popular parts of the tables, as before, the small size of the Polish and Slovak sets is very noticeable. By contrast, the large size of the English and German sets are evident as well.

There are not many common, least popular n-grams. This could mean that the dataset was too important for them to have many n-grams with few occurrences, and then, in

common with other languages.

Similarly to the study of the alphabet, the Czech set seems to be the most balanced language. It also seems to be the one that mix with the others the more easily, as it is the section of the heatmap with the highest intensity, and has the most common traits with other languages, no matter the n-gram unit.

The numbers of highest and lowest occurrences are a little different in scope, and the same goes for the languages related to it. In the bigrams, Slovak-Hungarian has the highest common number, with 16, and Polish-Czech has the lowest, with 3. In the trigrams, the Czech-Hungarian is the highest, with 76, and Danish-Slovak the lowest with 30. For the tetragrams, Czech-Polish is the highest with 84, and Hungarian-Slovak is the lowest with 12. The Hungarian and Czech sets are quite present.

This part of the table indicated the poor condition of the Polish and Slovak sets, the high state of the English and German datasets, and the intermediate state of Hungarian and Czech.

### **9.2.2.3 General Observations**

To sum up, I can say that English and Slovak seem to be the languages that are the most at odds with each other. Exploring further with some comparison with those languages could reveal itself to be interesting. The Czech language seems to be the fairest of them all. The German and English languages should have no issues, in terms of recognition, since they appear to really be prominent in the training data, mostly regarding the most popular n-grams. It should be most useful in the future step of the experiment.

With this newfound and thorough knowledge of the multilingual dataset, I am well-equipped to pursue my research on the impact of the n-grams on recognition.

# TESTING THE EFFICACY OF THE MODEL BY OBSERVING ITS ERRORS

---

To fully verify the validity of my hypothesis, I need to thoroughly evaluate the efficiency of the multilingual model. Metrics and study of the prediction errors are required to acquire useful data. But, to truly understand the ability of the model, I aim to investigate additional elements. Therefore, in this chapter, I introduce a new set of texts, from the same corpus, but with data unseen during the training, and I focused my study on testing it against unknown language data this time. Since my dataset relies on multilingualism, it allows me to test the capacities of the model against languages not part of the training.

With those new data, I am subsequently doing comparative analysis, retrieving and analysing the metrics, and token error analysis, observing the n-grams distribution and concluding from the differences between references and predictions.

## 10.1 A Personalized Test Set for the Multilingual Model

### 10.1.1 What Should Be Included for an Efficient Test?

The next step in my study is to test the efficacy of the model I produced with the training data I created from the EHRI dataset. To do so, I need a test set that will present enough specificities and peculiarities to adequately render the strength and difficulties of the model. The best way to do that is to test it on more than one language, since it was created as a multilingual model. The key criterion is that the test languages must use the Latin alphabet, as changing the script will be an inefficient way to discover the ability of the model.

I wondered what would be the best way to effectively test the model. I decided first that the test set would need to have some languages from the ones it was trained with.

Additionally, it calls for some languages unknown to the model. This would allow me to see its capacities in uncharted territories.

Considering the languages it already knew, there are seven in the dataset. They are represented with more or less quantity of documents. I established that there are three levels of quantity in the training data:

- Large (English, German, and Czech);
- Medium (Danish and Hungarian);
- Small (Slovak and Polish).

In regard to that, I decided to select a language from each of those levels.

From the large part of the dataset, I picked the English language, since I wanted to see the ability of the model on the most basic language in all that it knows. Indeed, English has no diacritics or special characters that could create extra difficulties.

From the medium part of the dataset, I selected the Danish language, as it contains unique diacritics. It would bring more challenge to the model than the Hungarian language would have done, because the diacritics in Hungarian are more similar to those of the German or Czech languages.

Lastly, I had two choices from the small part of the dataset: Polish and Slovak. Polish could have been an interesting choice, as it contains diacritics completely unknown to the training data, but I decided to opt for the Slovak language. First, it is the language that includes the higher number of diacritics, about eighteen. Second, the study done on the token analysis in the previous chapter demonstrated the affinities and odds between the languages of the dataset in regard to their n-gram. The results from it showed that Slovak was the language most at odds with English. It was also the case with German, the other most present language of the dataset. In addition to being barely found in the training data, being very different from the popular languages in terms of n-grams could create some trouble for the recognition. The resulting prediction errors could bring precious response elements for my experiment.

I have now three languages chosen for the test set, but to fully evaluate my model's performance, I need to add more. To do so, I have two requirements:

- It needs to be a language not known from the model, i.e. not present in the training data;

- It should be one of the languages present in one of the four EHRI online editions, as it is my source for building the model training data.

This leaves me three choices: French, Italian, and Dutch. Currently, the languages that are part of my dataset have two origins. They are either Indo-European or Uralic languages.

The Uralic language is spoken over a large geographical area,<sup>1</sup> and contains more than thirty languages [Bakró-Nagy 2012], among which Hungarian, represented with thirty documents in the training data.

All the other languages are part of the Indo-European language,<sup>2</sup> that includes more than four hundreds languages [Kapović 2017, p. 1], distributed in the ten principal branches of the Indo-European language family [Kapović 2017, p. 3].<sup>3</sup> The six Indo-European languages of my dataset belong to only two of those branches. English, German, and Danish belong to the Germanic branch.<sup>4</sup> Czech, Polish, and Slovak belong to the Balto-Slavic branch.<sup>5</sup>

The three other languages in the EHRI Online Editions, written with the Latin alphabet, also belong to the Indo-European language family, but they are not part of the same branch. Dutch, like English and German, is part of the Germanic branch [Kapović 2017, pp. 4–5]. Italian and French belong to the same branch, but it is a branch that has not been observed yet. They are related to the Italic branch, also called the Romance languages [Kapović 2017, p. 5]. Considering this fact, and since Dutch belong to a family already studied plenty, I decided to choose documents in French and in Italian for the test set. Moreover, they are languages with their own diacritics, among which some are completely unknown to the model, as they don't exist in the languages of the training data. This presents a twofold challenge for the model, which will be interesting to evaluate.

---

1. It extends “from the northern border of Norway in Scandinavia down south to Hungary (and its neighbouring countries) in East-Central Europe and eastwards to the Ob and Yenisei rivers and their tributaries in Siberia and the Taimyr peninsula in northern Siberia” [Bakró-Nagy 2012]

2. This family of languages, i.e. “languages that have evolved from a single original proto-language” was “originally, a couple of millennia ago, spoken from Europe to India. Now, in postcolonial times, they are spoken all over the world” [Kapović 2017, p. 1]

3. Those ten branches are Albanian, Anatolian, Armenian, Balto-Slavic, Celtic, Germanic, Greek, Indo-Iranian, Italic, and Tocharian.

4. It is divided itself into two branches, North Germanic and West Germanic, relating to the geographical area of the people speaking it [Kapović 2017, pp. 4–5, 394]

5. It is spoken in the Baltic, the Balkans, and Central and Eastern Europe [Kapović 2017, p. 5]



Image	Origin	Date	Lines	Link
Danish 1	DR	1943-05-28	27	<a href="#">Link</a>
Danish 2	DR	1942-06-15	27	<a href="#">Link</a>
English 1	BF	1938-08-05	33	<a href="#">Link</a>
English 2	BF	1938-05-09	42	<a href="#">Link</a>
French 1	EHT	1942-10-10	45	<a href="#">Link</a>
French 2	EHT	1942-10-10	46	<a href="#">Link</a>
Italian 1	DR	1943-03-16	36	<a href="#">Link</a>
Italian 2	DR	1941-10-07	22	<a href="#">Link</a>
Slovak 1	BF	1938-09-06	18	<a href="#">Link</a>
Slovak 2	BF	1938-04-21	26	<a href="#">Link</a>

Table 10.1: Distribution of the EHRI test set

### 10.1.2 Presentation of the Test Set

Now that the languages for the test have been selected, the set can be created, and its distribution is observed in Table 10.1.

In total, it is made of ten pages, with two for each language. All told, the pages contain 322 lines to be recognized, with a heterogenous distribution. There are 54 lines for Danish, 75 lines for English, 91 for French, 58 for Italian, and 44 for Slovak. The two unknown languages to the model are the first and third most important part of the test set, which suggests an efficient test for the model, as it should provide enough unknown data to obtain a comprehensive assessment.

The images for the set have not been chosen randomly. For the languages present in the training data, the goal was to pick one image that was already included in the training data and one that was not seen before, which seemed an ideal way to test the model. That way, the accuracy of the language could be verified, based on a proper learning and not just on overfitting, and the success rate could be compared on the images it knows and the ones it doesn't. However, this was not possible for all the languages.

It was easily feasible for the English, since it is, with the German, the most used languages in the EHRI Online Editions. There were also enough images in Danish to be able to do the distinction between the two images of the test set.

The Slovak, on the other hand, was only used in the 15 pages already included in the training data. Therefore, the model has already seen and learn from both images.

The diversity of the test set can also be observed through the images' origin: they come from three out of the four editions.

Nisko was not one of the sources for the test set, which is easily justifiable by the fact that all Nisko documents are German, a language not retained for the test set.

The Danish and Italian pages come from the same edition, the Diplomatic Reports, as this is the only edition featuring those languages.

The English and Slovak have been both taken from the *Begrenzte Flucht* edition. Although I could have selected English from any edition (except Nisko), the images selected from *Begrenzte Flucht* were different enough to efficiently test the model.

Finally, the French images are from the *Early Holocaust Testimonies* edition. They are also the only pages taken from the same document.<sup>6</sup> The other ones were picked among different documents from the editions.

Lastly, in addition to the variety of languages, lines' quantity, and prior knowledge of the model to some pages of the set, the quality's images fluctuate. It goes from pretty noisy to clean and clear sheets.

The English and the French documents propose images of excellent quality, with regular structure and lines separation, as seen in Figures 10.1a and 10.1b. It is also the case for the second Danish image.

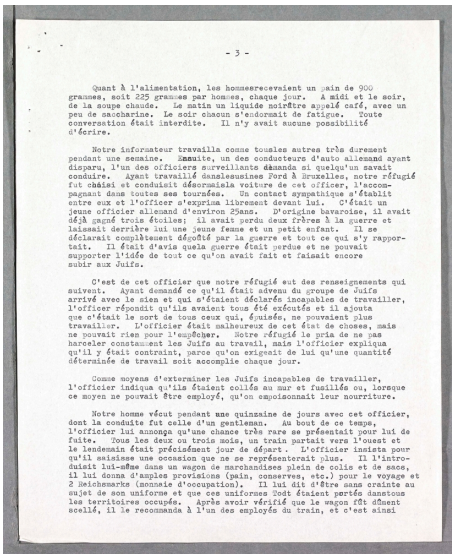
The first Italian and the second Slovak images share a similar yellowish paper and a standard layout, as well as some underlining made in a red pencil. However, this underlining is way more striking in the Italian image. Mixed with a more bold typeface than the other image, it makes it noisier.

The second Italian (see Figure 10.2a) and the first Slovak images shared a similar noise: there are headers in another font, as well as, in the case of the Italian, many stamps scattered around the page, and in the case of the Slovak, blue pencil's underlining.

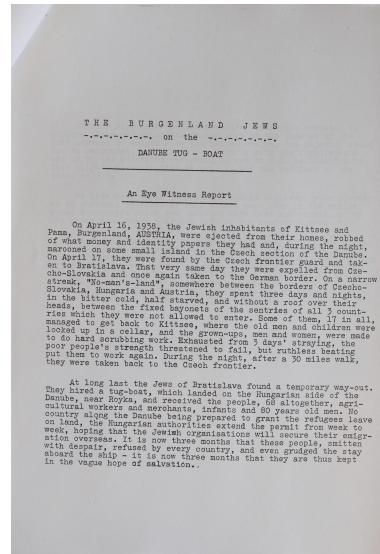
To finish, the first Danish image (see Figure 10.2b) is the noisiest of all. It has some differently coloured text in the top of the image, and verso text interferes with visibility, which tends to create difficulty if the segmentation includes it in the polygons. Lastly, there is a combination of a grainy paper texture and bolder font text, which seems to make the writing a little blurry. This may challenge recognition accuracy.

---

6. A document here is interpreted as a complete testimony, report, letter, or another type of paper, made of several pages/images



(a) French

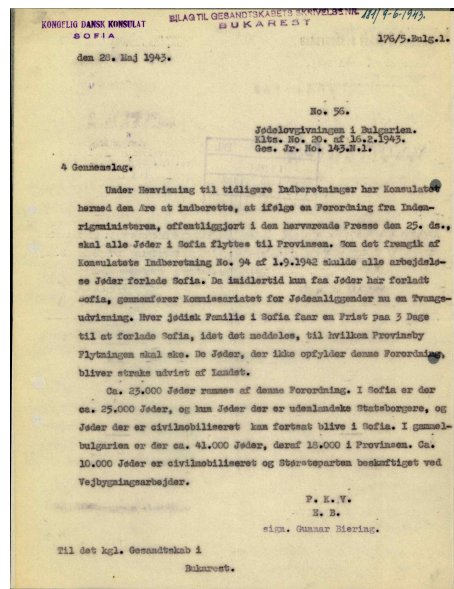


(b) English

Figure 10.1: Clean and clear images



(a) Italian



(b) Danish

Figure 10.2: Noisy images

## 10.2 The Metrics: Evidence of the Model's Overall Efficiency

### 10.2.1 Applying the Model to Produce Metrics

I have now explored the training data and observed some likeness between the languages. I also selected specific pages for a test set, and I am now conducting an experiment to attest if the model is indeed working properly, and if so, how accurate it is. As presented in subsection 8.2.4, I trained several models from the training data, but for the purpose of the subsequent analysis, I am using the generic model. Indeed, I only want to test the efficacy of a multilingual dataset. When I did my training, this general model was produced in two versions, Normalization Form Canonical Decomposition (NFD) and Normalization Form Canonical Composition (NFC), which are Unicode normalization process. The difference between the two is the way diacritics are processed. In one case, the NFD, the normalized version of the diacritics, decomposes the character and its glyph, and in the other case, the NFC, the character and the glyph are considered as a whole. Both models achieved similar accuracies, with respectively 97.29% and 97.20%. Consequently, I decided to apply both models to the pages of the test sets, as I wanted to see if any discrepancies appeared. During manual transcription, I observed that the interface treats diacritics as part of a single character, which matched the NFC normalization format. On the contrary, the model NFD created false errors, by considering every diacritic, whether they were correctly or incorrectly recognized, as mistakes in the prediction. This excluded the NFD model, as it would completely hinder my comparative analysis. Given this, I proceeded with the NFC model for this experiment and those after.

Next, I followed the same steps as the one I did for the previous comparative analysis,<sup>7</sup> by using KaMI App.<sup>8</sup> The comparisons were done in two versions. First, I applied the model standardly and retrieved the metrics. Then, after observing some recurring errors and to obtain more precise metrics, I decided to make use of the options proposed by the Kraken as Model Inspector. I chose to ignore the punctuation in the analysis, because it can frequently be causes for errors, and I am only interested in studying the model's performance on characters' recognition. The results can be seen in Table 10.3. It combines the two series of metrics, as the table's rows alternate between the version of the text

---

7. See section 5.2

8. See subsection 5.1.3

taking into account the punctuation and the version that ignores it.

To add some visualizations and concrete results to those metrics, I also retrieved the “[versus text](#)” for each page, which are an illustration of the substitutions, insertions, and deletions existing between the reference and the prediction.

	EN 1	EN 2	DA 1	DA 2	SK 1	SK 2	FR 1	FR 2	IT 1	IT 2
Levenshtein Distance (Char.)	48	10	106	14	11	11	67	94	98	21
Levenshtein Distance (Char.)*	39	9	106	11	10	10	62	90	90	18
Levenshtein Distance (Words)	33	9	73	19	14	13	62	94	75	21
Levenshtein Distance (Words)*	24	9	73	16	11	13	58	90	68	19
WER in %	9.85	2.325	39.673	8.636	10.37	6.046	13.596	20.042	20.215	11.351
WER in %*	7.185	2.362	39.673	7.339	8.208	6.103	12.803	19.313	18.428	10.439
CER in %	2.51	0.421	8.811	0.893	1.199	0.812	2.433	3.196	4.224	1.76
CER in %*	2.108	0.389	9.298	0.726	1.129	0.759	2.307	3.132	3.956	1.534
Wacc in %	90.149	97.674	60.326	91.363	86.629	93.953	86.403	79.957	79.784	88.648
Wacc in %*	92.814	97.637	60.326	92.66	91.791	93.896	87.196	80.686	81.571	89.56
Hits	1867	2363	1107	1557	906	1343	2689	2850	2226	1173
Hits*	1814	2308	1946	1509	875	1307	2629	2785	2190	1160
Substitutions	44	7	86	8	7	7	57	83	77	19
Substitutions*	31	3	84	5	6	7	46	61	69	12
Deletions	1	2	10	2	4	4	7	8	17	1
Deletions*	5	2	10	1	4	3	12	27	16	1
Insertions	3	1	10	4	0	0	3	3	4	1
Insertions*	3	4	12	5	0	0	4	2	5	5
Total char. in reference	1912	2372	1203	1567	917	1354	2753	2941	2320	1193
Total char. in reference*	1850	2313	1140	1515	885	1317	2687	2873	2275	1173
Total char. in prediction	1914	2371	1203	1569	913	1350	2749	2936	2307	1193
Total char. in prediction*	1848	2315	1142	1519	881	1314	2679	2848	2264	1177

Table 10.3: Comparative results for the EHRI NFC model with and without punctuation (\*)

HNo. 256.  
 Jødelovgivningemn i mBulgsarien.  
 Klts. No. 20. af 196.2.1943.  
 cGes. GJr. HNot. 13493.N.1.  
 34 Genanemslag.  
 Under RHernviemsning til tidligæere  
 DIundberetmninger har kKonsulatévt  
 hermed den Ære at indberette, at ifeølge en  
 Fororchdning fra Inden-

Figure 10.3: Gibberish substitutions in Danish

### 10.2.2 A Result Highlighting the Strengths and Weaknesses of the Model

Among all the results of Table 10.3, one appeared to be very striking. The metrics of the first page in Danish were terrible, in itself, and compared to all the other pages. There was only 60% of accuracy in the recognition, and the low accuracy was not caused by punctuation errors. Indeed, the metrics remained the same from one line to another. The quality of the image could explain this situation, as it is in a pretty low quality. Moreover, while observing the 19 and 20 rows of the table, I noticed that this page has one of the smallest amount of characters in reference, meaning that the small number of characters inflated the Word Error Rate (WER) and Character Error Rate (CER) percentages.<sup>9</sup> Additionally, the “versus text” showed that almost every three words, one is erroneously recognized, and some words contain multiple errors. Also, the errors seemed to be mostly gibberish, as no logic appeared applicable to the substitutions, as it can be seen with Figure 10.3.

The Danish image presented an extreme case of bad recognition, but the other metrics, notably for the languages that were part of the training data, demonstrated that it is not the norm for the model, as there were rather accurate, and revealed good recognition’s performances. The English and Slovak pages both have excellent results, with 90%, 97%, and 93%. There is even a WER of 86% that becomes 91% with the punctuation ignored. The second Danish page, finally, has a WER of 91%. It strengthens the suggestion that the

9. For more information on those metrics, see subsection 5.1.1

problem with the other page was quality of the image rather than accuracy of the model. Supplying those metrics with the “versus text” provide answers as to where the model has issues. The three main errors are on punctuation, uppercases letters and numbers. As those are mistakes usually met during post-OCR correction, this does not attest to any deficiency in the model, which was already observed with the previous dataset.

The punctuation appears to be a major issue in the recognition of this model. First, I remark evident disparities between the metrics ignoring, or not, the punctuation. Many Word Accuracy (Wacc) percentages gain one to three points. It is even an increase of five points for the first Slovak page. When the punctuation is completely ignored in the study, the pages in the languages of the training data, except for the bad Danish, reach a Wacc higher than 90%. Furthermore, the first Italian page attains a Wacc of 89.56%, although, the language is not part of the training data. Additionally, ignoring the punctuation also leads to a drop of the numbers of substitution. The first English page goes from 44 to 31. The French pages go from 57 to 46, and from 83 to 61. According to the “versus text”, those errors are quite diverse, and can be related to hyphens, commas, or apostrophes.

For the two languages that are not part of the training data, French and Italian, two types of results could be found. On one hand, the WER is at 11-13%. On another hand, it can reach 20%. While those were not outstanding results, it could be considered promising. Indeed, the model never learned anything from those languages. Observing the “versus text” furthermore established that. Besides the punctuation and numbers, the diacritics are the main issue. Diacritics, as I explained it in subsections 8.2.3 and 10.1.1, can be quite various. They are widely used in French and Italian, but they are not the same as those in Czech, Danish, or else, that were present in the training data. They appear to be an issue for the second Slovak page as well. Some specific Slovak diacritics, not present in the training data, are erroneously predicted.

Usually, as illustrated by Figure 10.4, those diacritics can be replaced by a random letter. It can be close in shape to the one it was supposed to recognize. In French texts, mostly, other diacritics replace it. Those are typically some that are more frequent in the languages of the training data, such as *ö*, *d'* or *ø*.



Figure 10.4: Prediction errors of the model on diacritics



To conclude this comparative analysis, I can state that the metrics' results are rather encouraging. The model seems to be efficient on the languages of the dataset, whether it has seen it many times or not. Moreover, it is accurate enough as well, on unknown languages, provided that it is the same script. The serious limitation appears to be diacritics. It seems unable to recognize and correctly predict a diacritic, if it has never seen it at all. It is also incompetent when the diacritics have barely been seen. This was deduced from the metrics of the Slovak pages. The model also struggles with numbers, uppercases, and punctuations, but it tends to be a recurrent problem, with medium models in ATR. Therefore, it can be considered a minor issue.

As the model demonstrates effectiveness, particularly for languages included in the training data, I have a good basis to confirm my hypothesis about the impact of the n-grams. I will continue by relying on the errors observed here and the various languages of the model.

## 10.3 Comparing Errors to Ground Truth: A Positive Response to Our Hypothesis

### 10.3.1 Gathering Data About the Prediction Errors of the EHRI Model

In the previous experiment, I observed that the model predicted several errors. Similarly to the previous token error analysis, presented in the chapter 7, I retrieved those errors in order to analyse them at their infralexical level.<sup>10</sup> I want to verify if, on a multilingual scale, what I previously observed, can be confirmed. The errors, in the raw form from which they were retrieved in the eScriptorium<sup>11</sup> application, are available in Annex H. However, they are not all relevant to the experiment I am conducting. I am working on finding correlation between sequences of characters in the training data and efficiency of the model. Therefore, the errors involving digits, punctuation or missing spaces were not included in the tables. Then, I used Script P.5.2 that contains, in its lines 19 and 20, the lists of the reference tokens and the prediction errors respectively. The items are split into

---

10. Level located between character-level and word-level, i.e., sequences of characters

11. <https://escriptorium.inria.fr/>

three lists, partitioned into sequences of two, three, or four characters.<sup>12</sup> Once produced, the lists have been divided by language, one after the other. From there, I can create and fill my tables of n-grams, which are available in Annex I.

The tables are done similarly to the ones from the previous token error analysis.

In blue, are all the cells that presented errors where the length of the prediction differs from that of the reference. An example can be observed in Figure 10.5.

In grey, are the cells where, even though the token might not be correct, the n-gram affected does not have the right number of characters in its sequence. It is demonstrated in Figure 10.6.

The main difference with the previous tables is that there is no red any more. Indeed, priorly, one of my goals was to compare one recognition model with another. I wanted to highlight the varied mistakes that each made. Here, I am only evaluating one model. Therefore, there is no comparison to be made. Then, it renders the red useless in the table.

Lastly, in green, are the cells where the predicted n-gram has a higher occurrence in the training data than the referenced n-gram. One variation can appear in some cases. There are cells highlighted in green and some numbers written in white numbers in it. White numbers happen when erroneous n-grams with lower occurrences in the training data are located in the same token as an erroneous n-gram with higher occurrences than the reference. Therefore, the white numbers are to not be included in the higher occurrences count. An example is provided with Figure 10.7.

THE	EE
For, ord, nin, g	For, orc, hni, ng

Figure 10.5: Examples of blue cells with deletion(s) (English) or insertion(s) (Danish) in the prediction

12. A sequence of two characters is a 2-gram or bigram. A sequence of three characters is a 3-gram or trigram. A sequence of four characters is a 4-gram or tetragram.

gi, à	gi, h
bud, ú	bud, ů
anno, nça	anno, nca

Figure 10.6: Examples of grey cells for bigrams (Italian), trigrams (Slovak) and tetragrams (French)

Correct Transcription	Model EHRI	Nb occ CT	Nb occ MEHRI
ik, ke	ih, le	210 817	157 1604

Figure 10.7: Example of a line, in Danish, from the table of bigrams, where there were two erroneous n-grams but the occurrences results differ

Once the table was structured this way, I gathered statistics to gain insights into the distribution and accuracy of the model's predictions. Those tables contain general information from the tables of n-grams. Figure 10.8a presents table results separated by n-grams, and is divided in two: columns and n-grams. The first part of the table shows the distribution of errors across 236 rows. It categorizes predictions into three cases: (1) errors where the predicted token matches the reference length (white cells), (2) errors where the token lengths differ (blue cells), and (3) instances where the n-gram is irrelevant (grey cells). The blue cells remain the same across the tables, as an unequal token length signify no usable tokenization, no matter the n-gram's size. Therefore, the white and grey cells are the ones where the content of Figure 10.8a changes. In the second part of the table, the cells centred on the erroneous n-grams. Firstly, the numbers of erroneous n-grams can be higher than the ones in the row representing the white cells. Indeed, there can be several erroneous n-grams in the same token. Then, I collected the number of cells where the predicted n-gram had no occurrence in the training data, and I also retrieved those where the predicted n-gram were more present in the training data than the reference one. Figure 10.8b retrieved, in a table, data collected in the previous table. They provide percentage for some of the mentioned information. The first row uses numbers from the first part of the previous table. It calculates the percentage of simple substitutions, and divides the number of predicted errors with the same length as the reference, to the total numbers of tokens. Then, the next two rows consider the second part of the table. It uses, in both cases, the total of erroneous n-grams as a denominator. On one hand, it calculates the percentage of groundless prediction, i.e. when the prediction couldn't be

explained by the composition of the training data, and the numerator is the number of instances where the predicted n-gram had no occurrence in the training data. On another hand, it calculates the percentage of justifiable errors, i.e. when the prediction could be explained by the composition of the training data, and the numerator is the number of instances where the prediction occurrences are higher than the reference occurrences.

Figure 10.8: Results from the Token Error Analysis tables - General Information

		4-grams	3-grams	2-grams
COLUMNS (236 in total)	CT $\neq$ M (for the same number of characters)	120	150	166
	char CT $\neq$ char M	39	39	39
	Not taken into account (error but not the right n-grams)	77	47	31
<b>N-GRAMS</b>				
N-GRAMS	Number of erroneous n-grams	124	155	176
	Occurrences in M = $\emptyset$	90	48	10
	Occurrences in M > Occurrences in CT	20	54	99

(a) By Numbers

	4-grams	3-grams	2-grams
Words with simple substitutions	50,85%	63,56%	70,34%
Occurrences where the substitution "makes sense"	16,13%	34,84%	56,25%
Occurrences where the substitution has "no base"	72,58%	30,97%	5,68%

(b) By Percentages

Then, I created a new set of tables. It contains the same information as the previous one, but divided into the five languages from the dataset. By separating the data by language, I can more accurately identify how the model performs with each specific language, which helps in pinpointing language-specific strengths and weaknesses. Figure 10.9a follows the same principle as the general one, except that each n-gram is separated in five

columns, meaning that summing these columns provides a comprehensive view similar to the general table. Figure 10.9b contains a little more information than the general table had. In the first part of the table, I calculated the proportion of errors specific to each language compared to the total errors and tokens, out of the 236 rows but also out of its language-specific row. This helps in understanding how each language contributes to the overall error rate. In the second part, I computed the percentages of groundless and justifiable errors for each language. This provides insights into how often predictions are unsupported by training data versus how often they are more frequent than the reference.

Figure 10.9: Results from the Token Error Analysis tables - By Language

		4-grams					3-grams					2-grams				
		EN	DA	SK	FR	IT	EN	DA	SK	FR	IT	EN	DA	SK	FR	IT
Columns (236 in total)	CT ≠ M (for the same number of characters)	10	34	11	39	26	15	42	6	50	37	17	45	11	60	33
N-grams	Number of erroneous n-grams	10	36	11	39	28	16	46	6	50	37	20	50	11	60	36
	Occurrences in M = ∅	8	22	8	30	21	9	8	3	18	10	1	1	1	4	3
	Occurrences in M > Occurrences in CT	0	4	2	9	5	2	16	1	25	10	7	23	9	49	11

(a) By Numbers

	4-grams					3-grams					2-grams				
	EN	DA	SK	FR	IT	EN	DA	SK	FR	IT	EN	DA	SK	FR	IT
Number of errors by language (on the total of CT ≠ M)	8,33%	28,33%	9,17%	32,50%	21,67%	10,00%	28,00%	4,00%	33,33%	24,67%	10,24%	27,11%	6,63%	36,14%	19,88%
Number of errors by language (on the total of columns)	4,24%	14,41%	4,66%	16,53%	11,02%	6,36%	17,80%	2,54%	21,19%	15,68%	7,20%	19,07%	4,66%	25,42%	13,98%
Number of erroneous n-grams by language	8,06%	29,03%	8,87%	31,45%	22,58%	10,32%	29,68%	3,87%	32,26%	23,87%	11,36%	28,41%	6,25%	34,09%	20,45%
Occurrences where the substitution "makes sense" (on the total of erroneous n-grams)	0,00%	3,23%	1,61%	7,26%	4,03%	1,29%	10,32%	0,65%	16,13%	6,45%	3,98%	13,07%	5,11%	27,84%	6,25%
Occurrences where the substitution "makes sense" (on the total of erroneous n-grams by language)	0,00%	11,11%	18,18%	23,08%	17,86%	12,50%	34,78%	16,67%	50,00%	27,03%	35,00%	46,00%	81,82%	81,67%	30,56%
Occurrences where the substitution has "no base" (on the total of erroneous n-grams)	6,45%	17,74%	6,45%	24,19%	16,94%	5,81%	5,16%	1,94%	11,61%	6,45%	0,57%	0,57%	0,57%	2,27%	1,70%
Occurrences where the substitution has "no base" (on the total of erroneous n-grams by language)	80,00%	61,11%	72,73%	76,92%	75,00%	56,25%	17,39%	50,00%	36,00%	27,03%	5,00%	2,00%	9,09%	6,67%	8,33%

(b) By Percentages

### 10.3.2 What Can We Learn from the General Observations of the Tables?

First, I analysed the most general information from the tables of Figures 10.8a and 10.8b. I noticed that the errors where the tokens are of different length represent 16.5%, which is about one sixth of the errors. As for the n-grams not taken into account, they represent 32.6% of the tetragrams. There are 19.9% of the trigrams and 13.1% of the bigrams.

It means that the tokens that will be included in this study represent half of all the tetragrams, 63.56% of the trigrams, and 70% of the bigrams. Moreover, by observing the second part of the table, it is evident that several tokens have more than one erroneous n-grams. It is even the case for the tetragrams, despite them being large character's sequences.

While observing Figure 10.8b, the tetragrams seemed, once again, ineffective in the training. There are only 16% of occurrences where the substitution can be explained by the composition of the training data. By contrast, there are 72% of occurrences where it seemingly has no ground.

However, as soon as the unit decreases, the results improve. The trigrams have about 30% for each. It is truly with the bigrams, that I remarked a shift that could denote an impact of the n-grams. More than half of the erroneous n-grams predicted (56%) are present in a higher fraction than those in the reference, while only 5% comes from n-grams unfound in the training data. Compared to the percentages of the tetragrams, the difference is pretty striking. This seemed to confirm the hypothesis that the n-grams might indeed impact the recognition of the model. However, not every unit is useful.

To broaden those observations, I also wanted to compare the numbers of those general tables to some numbers from the tables of Figures 7.8 and 7.9 of the chapter 7. However, I am not comparing the results to the whole previous table, but only to the model Ground Truth (MGT), as it was the more complete model. The training data of MGT is twice the size of that of the model EHRI (MEHRI). Yet, I can easily view that the numbers from MEHRI are significantly better than those from MGT.

The percentage of explainable errors in tetragrams is the same: 16.13%. But it improves a lot for the trigrams and bigrams, by about 10 points. In trigrams, MGT is at 25.76%, while MEHRI is at 34.84%. For the bigrams, MGT is at 45.33%, while MEHRI climbs to 56.25%.

Furthermore, this phenomenon can also be found for the groundless errors. Despite being half its size, MEHRI has fewer of them. The tetragrams' percentage is 74.19% for MGT, and 72.58% for MEHRI. The trigrams' percentage is 48.48% for MGT, and 30.97% for MEHRI. Lastly, the bigrams' percentage is 9.33% for MGT, and 5.68% for MEHRI.

The more prominent variation is observed in general with the trigrams. In the previous token error analysis, their results were not conclusive as to their impact. The numbers were not revealing enough. Here, the trigrams seem to be more impactful on the training of the model. There are 9 more points for the justifiable errors, and 18 points less for

groundless errors.

After studying the two first general tables, I am going to get more details on the results with the two subsequent tables from Figures 10.9a and 10.9b.

First, I analysed the table of numbers to observe language distribution. This analysis confirmed the findings from the comparative analysis. The model errors are mainly done in French and Italian. They are the languages absent from the training data. It is also the case in Danish, which has one image of low quality. When looking at the number of erroneous n-grams, it seems that it goes up mostly for the Danish, which confirmed the comparative analysis' observations. The tokens are usually completely wrong and not made of unique mistakes, here and there. Regarding the distribution of the null occurrences for predicted n-grams, and of the predicted n-grams with higher occurrences than the reference, the results are instructive. For the tetragrams, the Danish language does not have as many null occurrences as the others, compared to the total of errors. This also happens with the trigrams and bigrams. It could be explained by the fact that it is one of the languages in the training data. By and large, for the training data's languages, the numbers decrease at the same time as the unit of the n-gram does. Furthermore, in bigrams, those numbers even lower to one for English, Danish, and Slovak. By contrast, it stayed pretty high for the models of the Paul d'Estournelles de Constant's dataset.<sup>13</sup> As for the unknown languages, the numbers remain high:

- 30 (French) and 21 (Italian) for the tetragrams;
- 18 (French) and 10 (Italian) for the trigrams;
- 4 (French) and 3 (Italian) for the bigrams.

Yet, compared to the total of erroneous n-grams, except for the tetragrams, it seems low enough. Finally, no matter the unit of n-gram, the numbers of predicted n-grams more abundant than those of the reference appear quite elevated for French, Italian, and Danish. It could prove that, indeed, when in doubt during recognition, the model chooses an n-gram it is more familiar with from its training. It appeared especially true when the language is unknown.

Now, regarding the table of percentages, it appears substantially that the Danish language has a terrible recognition result. But it is the French language, no matter the n-gram unit,

---

13. See section 7.3

that has the most errors. In each unit, it represents more than a third of the total of erroneous tokens with the right length. On the contrary, the Slovak language produced pretty small percentages. It is impressive, considering it is one of the smallest sets in the training data. About the distribution, it is mostly divided between the French, the Italian and the Danish languages. The English and the Slovak languages are barely accounting for more than 15 to 18% of the total of erroneous n-grams.

Then, I concentrated my attention on the occurrences where the substitution seems justifiable. The percentages do not really rise compared to the total of the erroneous n-grams, specially for the tetragrams. However, as soon as it is compared to the total of errors by language, the percentages become increasingly more revealing. For the tetragrams, Slovak and French have low percentages when compared to the total of erroneous n-grams, with 1.61% (Slovak) and 7.26% (French). But once it is compared to their total of errors, it reaches almost one fifth (18.18%) for the Slovak and one fourth (23.08%) for the French. It already indicates some pretty interesting conclusions, which will subsequently need to be supported with a detailed observation of the tables. This remains valid with the trigrams and bigrams, particularly for the French. In trigrams, the explainable errors are half of the total of French's errors. In bigrams, it is far superior, with 81.67%, which is about four fifth of the total. Likewise, the Slovak bigrams have a percentage of about the same fraction. The Danish percentages reach high numbers too. The tetragrams are not much, with only 11.11%. The trigrams are more than a third (34.78%) of its errors. The bigrams are almost half of it (46%). The English errors appear to be more random. The percentages of justifiable errors are quite low. There are none in tetragrams, only one eighth in trigrams, and a little more than a third in bigrams. Meanwhile, the other languages in this row are higher. The Italian language seems to be the odd element here. The errors do not transpire to come from the content of the training data. This can be observed with its fairly insignificant percentages and the lack of growth from one unit of n-gram to another:

- The tetragrams have 17.86%;
- The trigrams have 27.03%;
- The bigrams have 30.56%.

Even for the bigrams, which are supposedly the most impactful n-gram, the percentage barely exceeds one third of the total.



Afterwards, I concentrated my attention on the occurrences where substitution appears to be groundless. The percentages indeed diminish when compared to the total of erroneous n-grams. But it is the comparison to the erroneous n-grams' total by language that is even more revealing. Compared to the total of erroneous n-grams, I remarked that, among the languages:

- With the tetragrams, the percentage goes from 6% (English and Slovak) to 24% (French);
- With the trigrams, the percentage goes from 2% (Slovak) to 11% (French);
- With the bigrams, the percentage goes from 0.5% (English, Danish, and Slovak) to 2.2% (French).

Those numbers represent pretty low percentages of groundless errors. Especially, when they are compared to what was observed with the three models of the previous dataset. Furthermore, the discrepancies of percentage are particularly striking when comparing to the errors by language. The numbers varied from:

- 60% (Danish) to 80% (English) with the tetragrams
- 17% (Danish) to 56% (English) with the trigrams
- 2% (Danish) to 9% (Slovak) with the bigrams.

This is a significant decrease, as the percentages are becoming nine times less important from tetragrams to bigrams. This also tended to support the theory that the tetragrams are not one of the elements involved in the model training and recognition skills. Similarly to the observations of the justifiable errors, those specific percentages are more revealing than the general one. For example, for the English tetragrams, a measly 6% can become an 80%. For the Slovak trigrams, a 1.94% can become a 50%. Those observations strengthened my initiative of adding comparison against the specific numbers of the languages. The distribution between them is so wide, that it is significantly more telling to compare it this way. It allows me to learn more precisely about the accuracy of the model and the impact of the n-grams on it. While the Slovak language had a general percentage that seemed to indicate few groundless errors, no matter the n-gram unit, the more specific percentages shared a different narrative. There appear to be many cases where the recognition has no base. It can be for the tetragrams (72.73%), or the trigrams (50%). It is less

prominent for the bigrams (9.09%). The Danish has a big diminution in groundless errors. It goes from 61% of tetragrams, to 17% of trigrams, to 2% of bigrams. Those percentages of the Danish seem to indicate that, no matter the n-gram, the model still had pretty solid ground for the recognition of the Danish. Hence, it explains the fact that it did not often predict nonsense. The situation with the French errors appeared to be similar to that of the Danish. It goes from 76% in tetragrams, to 36% in trigrams, to 6% in bigrams. Those percentages tend to indicate that, apart from the tetragrams, the predictions in French are not complete nonsense. The English's results are rather confusing. The tetragrams contain 80% of errors from occurrences of groundless errors. The trigrams have 56%. It is already substantially less, and only about half of it. Finally, the bigrams have 5%. It represents barely anything. This would tend to prove that, if the n-grams have indeed an impact, the bigrams are the main source of learning, with some extension to trigrams in some cases. At last, the Italian percentages are rather similar to the other languages for the groundless errors: it represents 75% of the tetragrams, plunges down to 27.03% in trigrams, and finishes at 8.33% in bigrams.

Regarding those various numbers, it would be interesting to verify, yet again, the differences of numbers between the occurrences of the reference and the occurrences of the prediction. Despite having a cell in green, it would be more revealing to know the actual gap between both. It could be particularly fascinating to do it for the two unknown languages. I could see how well better it did. In the cases of groundless errors, retrieving what the model predicted, but also what the numbers of occurrences of the reference next to it are, could be insightful. I could be able to verify if the model had ground to recognize the token initially, or if it was completely lost by what it had to recognize. Investigating further on the numbers that are neither in one category nor the other would also be a good idea. I remarked:

- For the bigrams, that it represents more than 60% of the English and the Italian, and 50% for the Danish.
- For the trigrams, it is more than 45% of the Danish and Italian, and more than 30% for the Slovak and English.
- For the tetragrams, it is more than 20% of the English and Danish.

Therefore, those numbers seemed to show that the n-grams might indeed be impactful to the recognition. It is especially seen for languages not part of the training data, but it would be good to support this with detailed observations of the tables of n-grams.

In order to add more details to my statistics, I also decided to retrieve the total of instances where there were null occurrences.<sup>14</sup> I reused, for some parts, the content of the previous tables. There are two tables again, a generic one, and one specific for languages. This time, I retrieved three types of instances of null occurrences:

- There are the cases where the occurrence is null for the reference (Occurrences in Correct Transcription (CT) =  $\emptyset$ );
- Then, I retrieved the amount in which occurrence is null for the prediction (Occurrences in Model (M) =  $\emptyset$ );
- Lastly, I counted the cases where the occurrence is null for both the reference and the prediction (Occurrences in CT and in M =  $\emptyset$ ).

From Table 10.4, I can observe that, for the tetragrams, there are more unknown occurrences for the model's prediction than for the reference. There are also many in both. The instances where the occurrence was null in both cases correspond to half of that of the prediction and almost all in the reference.

For the trigrams, there are more unknown occurrences in the correct transcription than in the prediction. Although, the difference is not that big. Almost half of both is a case of double null occurrences.

Lastly, for the bigrams, there are way more unknown in the reference than in the prediction. But this time, there are barely any double null occurrences.

	4grams	3grams	2grams
Occurrences in CT = $\emptyset$	58	52	35
Occurrences in M = $\emptyset$	90	48	10
Occurrences in CT and in M = $\emptyset$	45	23	2

Table 10.4: General distribution of unknown occurrences in the tables

With Table 10.5, I obtained much more information. First, regardless of the n-gram unit, the majority of null occurrences in the reference are found within the French language. In contrast, those numbers are not very high in the prediction. This tends to indicate that the French language is lexically pretty far from the languages of the training

<sup>14</sup> Null occurrences and unknown occurrences refer to the same thing, i.e. no trace of the n-gram in the training data

data. This could be due mostly to its diacritics.

However, the situation is quite distinct for the Italian language. Although it might have a high number of unknown occurrences in the tetragrams of the correct transcription, the numbers plunge down in the trigrams and the bigrams.

The results of this table can also raise some concerns about the Slovak language. The reference is automatically completely deprived of null occurrences. Indeed, there were not enough Slovak documents to have unknown ones in the test set. However, despite that and the fact that the model was trained on Slovak, there are some null occurrences in the prediction. While there are not that many, the model should have been able to recognize what it was transcribing. Yet, it got it wrong, even more so with character's sequences it did not learn during its training.

For the English, although it does have some unknown occurrences in the correct transcription, there are not many, and the same goes for the prediction.

Finally, the Danish has very few unknown occurrences in the reference, no matter the n-gram unit. Whereas, its number of unknown occurrences in the prediction is rather high. But it quickly plunges down with the smaller n-grams.

	EN	DA	SK	FR	IT
Occurrences in CT = $\emptyset$ (4grams)	4	4	0	35	15
Occurrences in M = $\emptyset$ (4grams)	8	22	8	30	21
Occurrences in CT and in M = $\emptyset$ (4grams)	4	3	0	26	12
Occurrences in CT = $\emptyset$ (3grams)	4	4	0	36	8
Occurrences in M = $\emptyset$ (3grams)	9	8	3	18	10
Occurrences in CT and in M = $\emptyset$ (3grams)	3	2	0	13	5
Occurrences in CT = $\emptyset$ (2grams)	2	0	0	32	1
Occurrences in M = $\emptyset$ (2grams)	1	1	1	4	3
Occurrences in CT and in M = $\emptyset$ (2grams)	0	0	0	2	0

Table 10.5: Distribution by language of unknown occurrences in the tables

### 10.3.3 A Variety of Languages Leading to the Same Conclusion

In this subsection, I am doing a thorough analysis to the tables of token error analysis for the multilingual dataset. For this analysis, I will focus, yet again, on answering some

of the questions I had for the previous token error analysis.<sup>15</sup> I will, however, provide my answer one language at a time.

### 10.3.3.1 English Results

For the tetragrams, observable in Figure I.15, there are ten errors that are considered. In the correct transcription, four of them are occurrences unknown to the training data. The same goes for their prediction counterpart. In the prediction, in addition to those four, there are also four null occurrences.<sup>16</sup> The numbers in the prediction column are quite low. The ones in the correct transcription can sometimes climb high, such as to 149 or 122.

For the trigrams, observable in Figure I.8, there are sixteen errors considered. Four of them in the correct transcription are occurrences unknown to the training data. Three of those have a null occurrence counterpart in the prediction. In the prediction, ten of the errors are unknown occurrences. The only two green cells of the prediction column do not contain big numbers (37 and 1). In the reference column, several occurrence numbers are pretty high, like 164, 365, and 602. Additionally, they are usually opposed to some lacking n-grams.

For the bigrams, observable in Figure I.1, there are twenty errors considered. Only two are unknown occurrences in the correct transcription. One has a counterpart in the prediction. It is the only one in this column. This time, the numbers in the green cells climb way higher. There are occurrences such as 916, 1087, 1215, and 2130. However, it is also the case for the bigrams of the reference, where there are occurrences like 428, 970, 1220, 1697, and 1915. They are opposed, once again, to lacking n-grams.

The errors in the English language seem to be a bit about uppercases. When it is not, it frequently appears to be random. No examination of the content of the training data and the n-grams happens to be able to give a proper explanation.

### 10.3.3.2 Danish Results

For the tetragrams, rendered in Figures I.15, I.16, and I.17, there are thirty-six errors considered. Four in the reference are not known to the training data. Three prediction counterparts have the same value. In the prediction, twenty-two predictions have null

---

15. See subsection 7.3.2

16. Null occurrences and unknown occurrences refer to the same thing, i.e. no trace of the n-gram in the training data

occurrences in the training data. For this language, the numbers are very low in the prediction. Except in one single case (156), it does not climb much higher in the reference. As for the three green cells in the prediction, the gaps were not that wide, with [15 12],<sup>17</sup> [8 2], and [17 2].

For the trigrams, rendered in Figures I.8, I.9, and I.10, there are forty-six errors. Four in the correct transcription are unknown to the training data. There are two prediction counterparts with the same value. In the prediction, eight trigrams are unknown n-grams to the training data. Here, the numbers are slightly rising, in both columns. Except in some exceptional cases, the gaps between the green cell of the prediction and its correct transcription equivalent are not that wide, such as [50 159], [13 25], [23 64], [90 240], and [93 100]. However, by contrast, there are sometimes significant gaps when the reference is higher than the prediction, such as [759 175], [165 21], and [243 1]. In the prediction, the numbers varied much. There are a few between only one to five occurrences.

As for the bigrams, rendered in Figures I.1, I.2, and I.3, there are fifty errors considered. This time, none are unknown in the reference. In the prediction, one n-gram is unknown, and its correct transcription equivalent is only a 2. There are pretty big numbers in both columns. The green cells of the prediction usually win with an overwhelming majority, with numbers such as [134 1215], [57 3349], [489 935], or [133 1744]. There are also some wide gaps the other way around. But they are quite insignificant in comparison.

The errors in the Danish language seems to be mostly due to the substitution of similar looking characters such as *s* and *e*, *s* and *a*, or *m* and *n*. It is also due to some uppercase issues. In many instances, and notably for the bigrams, the model, when faced to something unrecognizable to it, seems to have predicted character sequences that it had seen more during its training.

### 10.3.3.3 Slovak Results

For the tetragrams, observable in Figure I.21, there are eleven errors considered. Obviously, as the Slovak documents are also part of the training data, none of the reference's n-grams are unknown to it. However, except for one cell with an occurrence of 50, all the other ones only have one to five occurrences. In the prediction, there are only three n-grams that are not null, but their numbers are not very high: 7, 13, and 2.

For the trigrams, observable in Figure I.14, there are six errors. Unusually, the numbers

---

17. The format [a b] will be used as of now to present the occurrence numbers from the table, with "a" as the reference cell, and "b" as the prediction cell.

of errors diminished this time. Five tokens have become grey cells, because the error is at the end of the token. Yet again, none are unknown to the training data from the correct transcription. In this column, the numbers are again rather low, except for one single instance ([291 18]). In the prediction, there are three n-grams that have unknown occurrences in the training data. The remaining numbers are not very high, the only green cell is a 3, and its counterpart's occurrence in the reference's column is a 1.

Finally, for the bigrams, observable in Figure I.7, there are eleven errors considered again. Still, like previously, no unknown is in the correct transcription. In the prediction, there is one bigram unknown to the training data, which has a 25 in the counterpart column. The numbers for the reference's n-grams are still rather low. There are either three to nine occurrences, 19 to 27 occurrences, and one case of 43 occurrences. There is also a pretty consequent exception, an n-gram with an occurrence number of 2274. In the prediction, the green cells were usually the bigger number, with significant occurrences such as 260, 279, 157 and 418. The only cases where the occurrences of reference were bigger to that of prediction were due to a null occurrence. There was also an unexpectedly high number (976), matched against an exceptionally higher one (2274).

The errors in the Slovak language were pretty rare. When they happened, they seem to be mostly happening to the diacritics. The model appears to have proposed diacritics that it knows. Although, in those pages, were diacritics it learned, their quantity might have been too small to be properly assimilated. Therefore, in some of those cases, the Slovak diacritics have been replaced by either the Danish ones or the letter without its sign.

#### 10.3.3.4 French Results

For the tetragrams, rendered in Figures I.17, I.18, and I.19, there are thirty-nine errors considered. In the correct transcription, thirty-five are not known to the training data. Twenty-six prediction counterparts have the same value. In the prediction, thirty predictions are unknown occurrences in the training data. The green cells of the prediction are not really high. Except for two instances that are [44 60] and [10 64], the number would automatically be higher, as it was against a null occurrence. In the correct transcription, there are only four cases where the occurrences are not null: 1, 4, 1 and 18. Their prediction counterparts are always a null occurrence.

For the trigrams, rendered in Figures I.10, I.11, and I.12, there are fifty errors. In the correct transcription, thirty-six have null occurrences. Thirteen prediction counterparts

have the same value. In the prediction, eighteen trigrams are unknown occurrences to the training data. Few of them have counterpart that are not null. For the prediction, only two of the green cells do not have a non-zero counterpart: [14 25] and [2 11]. The occurrences numbers in the prediction do not rise above 50, except in about five instances. As for the reference, except for two cells that are 71 and 66, the numbers that were higher than the prediction do not rise above 16.

Finally, for the bigrams, rendered in Figures I.3, I.4, and I.5, there are sixty errors considered. Thirty-two, in the correct transcription, are not known to the training data. Only two prediction counterparts have the same value. In the prediction, four bigrams are unknown occurrences in the training data. The two null occurrences are against an 11 (*Q* predicted as an *O*), and an 8 (the letter *l* predicted as the number *1*). There are many high numbers in the column of the prediction. Thirty cells have an occurrence above 200. Among them, 14 have more than 500. Among those, 10 have more than 1000. Although there are some in the correct transcription too, it is more uncommon, as they do not rise as high as those in the prediction. For the instances where the reference' occurrences were bigger than that of the prediction, the gaps were varied, and except for the null occurrences, they were [709 369], [510 49], [44 13], [1664 625], or [1101 87]. For some of those cases, the facsimile had overlapping characters. It could have prompted the confusion of the model, even though it has known the content, as seen with the numbers in the correct transcription.

The errors in the French language are, in the majority, due to the French diacritics. Indeed, many of those are completely unknown to the model. Considering that, the model appears to have replaced them by diacritics that it learned to recognize and predict. It is mainly Czech and Danish diacritics. It could also have been replaced by the letter without its sign. The bigrams appear to be the most impactful here. But the trigrams also seem involved in the recognition's accuracy. In other cases, the model did not manage to recognize letters that it had not seen much during its training, like the *q*. Therefore, the sequence of characters including that character is replaced by sequences it knows more.

### 10.3.3.5 Italian Results

For the tetragrams, rendered in Figures I.19, I.20, and I.21, there are twenty-six errors considered. Fifteen from the reference column are unknown to the training data. Twelve prediction counterparts have the same value. In the prediction, twenty-one tetragrams have no occurrence in the training data. There are some exceptional cases of numbers



climbing high in the correct transcription, like 44, 62, 37, or 72. Otherwise, the occurrences do not go higher than 10. In the prediction, except for two instances at 64 and 60, the green cells do not have occurrences higher than 20.

Then, for the trigrams, rendered in Figures I.12, I.13, and I.14, there are thirty-seven errors. Eight in the correct transcription have no occurrence in the training data. They have five prediction counterparts too. In the prediction, ten trigrams are unknown occurrences to the training data. There are few cases of high numbers in either the reference or the prediction. In the reference, there are 9 that are over 50, and among them, 3 are over 100. In the prediction, there are 8 that are over 50, and among them, 4 are over 100. The gaps between the green cells of the prediction and the cells of the correct transcription are rather wide. However, the situation is equivalent the other way around. The correct transcription has high numbers and large gaps too. There are only a few instances where, although the reference n-gram was more present, the difference was minimal, such as [23 22], [21 15], [155 143], [3 1], and [5 2].

Finally, for the bigrams, rendered in Figures I.5, I.6, and I.7, there are thirty-six errors considered. Only one bigram of the reference is unknown to the training data. Strikingly, its prediction counterpart has an occurrence of 2264. In the prediction, three bigrams are null occurrences in the training data, and their reference counterparts are 8, 11 and 196. Both the prediction and the reference have occurrences' numbers rising pretty high. In the prediction, there are 19 above 200, 14 above 500 and 8 above 1000, and in the reference, there are 22 above 200, 14 above 500 and 10 above 1000. Those references high occurrences still weren't enough, because they were matched against even bigger ones. There were gaps such as [1286 2274], [458 1797], [260 512], [416 1351], and [877 1087]. Nevertheless, the opposite also occurred, like [1017 516], [258 200], [719 458], [1215 817], [1254 1055], [869 598], and [1120 963].

The errors in the Italian language appear to be less related to diacritics than the comparative analysis revealed. Those results seem to be more baffling. The numbers are quite high in the correct transcription too, at least for the bigrams. However, in a few cases, when the answer for the erroneous prediction could not be found in the bigrams, the trigrams brought some enlightenment. This suggests that the trigrams could indeed be impactful as well in the recognition, especially when the language is unknown. In cases in which the numbers were high in both columns, the error could be due to confusion from the model. It was already previously observed with the model Ground Truth. Lastly, for some cases where there were characters' overlapping, the model was either too confused

to do a right guess and instead predicted gibberish or proposed something close to what it had learned. It happened a few times with uppercases badly recognized or similar looking characters, hence the bad transcription.

Those various languages have different kind of errors and explanation behind it. The conclusion I draw from the last experiment appears to still be standing. This experiment supports it. The bigrams, as well as the trigrams occasionally, seem to have a real impact on the model. They appear to be crucial elements to the training and the recognition skills of the model. The conduct of this experiment has proven it. It becomes notably clear when the model has barely seen the languages of the text images it is used on, or if it has never even seen it at all. This was mostly proved with languages containing many diacritics. It was the case for French here, and the predictions done by the model.



# CONCLUSION

---

The objective of this thesis was to further the knowledge on the training of text recognition models through neural networks. With various experiments, I studied the correlation between the content of the ground truth and the training, in order to find out in what ways the content of the ground truth has an impact on the accuracy of the recognition and the results of the model once applied.

## **An Answer To This Thesis' Research Question: The N-Grams**

This inquiry stems from an interest in increasing the understanding around the functioning of the neural networks for model training, which is still a fairly recent technique in that domain. In order to fill some gaps, I explored several leads.

I started by studying the lexicon, through sets' content analysis and recognition's metrics comparison. It turned out to be an ineffective lead.

However, the results of this experiment directed me towards an examination of smaller units of texts. They were n-grams. During some tests on a single language dataset, some promising results appeared. They implied that the bigrams might be a source of learning for the model. The possibility of the trigrams and tetragrams being effective was also hinted at. The results were not conclusive enough, though.

Following those encouraging outcomes, I decided to diversify my work. I conducted the same experiments as those made to verify the impact of the n-grams. But I did it this time on a multilingual dataset. The languages were all in Latin script. Additionally, to obtain results as conclusive as possible, the tests were not only conducted on languages that were found in the training data. They were also done on languages the machine had not previously encountered and that were hence not taught to the model during its training. The results backed up what was observed during the n-gram study of the single language dataset. They also added more conclusive information on the effective impact of bigrams, the occasional impact of trigrams and the null influence of the tetragrams.

Therefore, it is possible to answer with differentiated precision the research question of the thesis. I initially hypothesized that the recognition was done at word-level. My idea was that, if it was so, it could be pertinent to create lexicon-driven ground truths. It could then be reused on corpora with a similar writing and an equivalent topic. The first experiments I conducted proved that the lexicon was in no way involved in the training of the model. This implies that there is no need to care about the topics included in the ground truth. They could be as diversified as possible without it having an impact.

My first hypothesis, a recognition at the word-level, was not a success, but the second one, a recognition at the infralexical level, proved to be efficient. The n-grams are part of the answer on what the model learns from the training data. The analysis showed that recognition works better with little units, bigrams mainly, than with larger ones. The essential detail there is the quantity of the n-grams. Whether it was by analysing the models War, Other, and Ground truth, or the multilingual dataset and its subsets, the size difference in each case was a key element to the recognition's accuracy. The two weakest links identified during the token error analysis were the training data of the set War, and the subset Slovak. Despite learning from them and being tested on content present on the training data, the quantity was too insufficient for the models to recognise patterns correctly. Similarly, overflowing the training with data can be debilitating, too. It can cause some confusion during the model's performance. It was hinted with the training of the set Ground Truth, and, in some way, with the subset English. Therefore, it is imperative when creating training data for a model to be either constrained to a certain point for a single-language model, or balanced for a multilingual model.

## Uppercase and Diacritics: The N-Grams' Limits

The influence of the n-grams for the recognition does present some limits nonetheless, which become particularly problematic when dealing with a multilingual model for certain languages. Uppercases remain a challenge for text recognition models, as recognizing them can be complicated when not relying on a sufficient amount of references. The experiment about the influence of n-grams established that the training data should contain a vast quantity of lines with only uppercases to enable the model to recognize them. Therefore, the model should be provided a sufficient amount of n-grams of those types of combinations to be able to learn from them.

Studying the n-grams also shed light on another consequent limit, though most likely

observed for multilingual models: the diacritics. It is a minuscule element that can be placed above, below, or across a character, in one direction or the other. However, the diacritic, no matter how minuscule it is, is essential for an adequate comprehension of a word, and, sometimes, it is misrecognized by a model, due to a lack of learning. The diacritic has to be part of the training data to be recognizable, but it also has to be in sufficient quantity for the model to recognize it, which can be quite complicated, since some diacritics can be pretty rare.

## Concrete Applications of Acquired Knowledge

Since I know that the n-grams are part of the answer of the recognition mechanism, I wonder how that knowledge could be concretely applied. I conducted a thorough token analysis to obtain information on the diversity of the distribution. I examined precisely how the training data were composed. This method was helpful to determine the impact of the n-grams. But realistically, an analysis as thorough as this is not doable for every researcher or project member that works on a corpus.

Therefore, using this knowledge to more efficiently produce a single-language model might appear to be difficult. Only two tangible solutions could be directly applied: solving the current issues and privileging quality over quantity.

First, the training data need to have a good amount of uppercase characters to produce an accurate model. Whether it is complete uppercase tokens, or ones with only initials, this would ensure a better recognition of the model.

Second, it is essential to favour quality over quantity, and to not blindly add new content to the training data. Indeed, too much data could become confusing for the model. Knowing that the n-grams impact the recognition could be more applicable to multilingual models. The n-grams are the elements involved in the training of the model. Therefore, multilingualism should not hinder recognition, as long as the languages share a similar enough basis, as proved with the EHRI dataset. Using Kraken, some other projects already created training data with several languages and managed to produce an efficient multilingual model.

Regarding other works, CATMuS (Consistent Approaches to Transcribing Manuscript) developed two types of multilingual models. CATMuS-Medieval is a model made on four languages (Old and Middle French, Latin, Spanish and other languages of Spain, Italian). It has a recognition accuracy of 95.1% [Pinche, Clérice, Chagué, Camps, Vlachou-

Efstathiou, Gille Levenson, Brisville-Fertin, Boschetti, Fischer, Gervers, Boutreux, Manton, Gabay, et al. 2024; Pinche, Clérice, Chagué, Camps, Vlachou-Efstathiou, Gille Levenson, Brisville-Fertin, Boschetti, Fischer, Gervers, Boutreux, Manton, and Gabay 2024]. CATMus-Print is a model with numerous languages (French, Spanish, German, English, Corsican, Catalan, Latin, Italian. . .) and from different times (16th century to now). It has an accuracy of 98.5% [Gabay, Clérice, Jacsont, et al. 2024; Gabay and Clérice 2024]. Open ITI developed training data with Arabic-script printed texts. It produced a model with an accuracy of 96.4% [Romanov and Seydi 2019; Kiessling 2022]. The models have a rather good accuracy, with an efficiency legitimated by their publication.

Studying those datasets in depth could be enlightening to see if it is possible to find similar patterns to those observed in the EHRI dataset. I would also like to find some additional elements to define how the knowledge of the n-grams' impact can be more concretised. Overall, those various examples demonstrate that creating a single text recognition model for a digital scholarly edition with multilingual documents is not out of the question. Once assured that the documents all have the same script and that their writing style is homogeneous, transcribing part of the corpus to create training data is possible. It would be time-saving, as there would be no need to have enough training data for each language, but just enough data of each language to train an accurate multilingual model.

## N-Grams and Digital Scholarly Editions

Regarding the creation of efficient text recognition model(s) for digital scholarly editions, the knowledge acquired here could have been of use when I was working on the DAHN project, notably. It is known that too little data is not enough to learn correctly. However, understanding priorly that too much data is problematic as well could have been time-saving. To generate a really accurate model, I used repeatedly the new transcriptions I had acquired after applying the model and correcting these mistakes. I then added new data to the model training, until 500 images were reached. Consequently, it took many epochs and hours to obtain a model of 97.9% of accuracy, which could have been as efficient with only half of that, according to my tests.

With that acquired knowledge, while preparing the ground truth for the model EHRI, I limited the amount of training data to about 250 images. This proved efficient, as demonstrated by the various analyses I presented priorly. Additionally, with half the documents, I produced an equivalently efficient model at 97.3% of accuracy. Being able to

easily balance between creating efficient recognition models and fastening the creation of a digital scholarly edition can be complicated. My hope is that this thesis provides part of a solution, in a pool of ideas and developments dedicated to that same question, such as is also trying to do so the ARIANE consortium, which I am a part of, that aims at creating “a space for discussion on the interpretation of the results obtained using (semi)automatic methods of text analysis” [Idmhand, Galleron, and Loudcher 2023].

## Impact of the N-Grams: What About Handwritten Texts?

Among the other reflections that could be pursued with that knowledge, it could be interesting to expand the research, by testing on new and different types of documents.

The documents I worked on during my experiments were exclusively typewritten. Likewise, the datasets I mentioned above were mostly made of printed texts. Yet, they are from several centuries ago, which presents its own difficulties. Consequently, the efficiency of the n-grams could differ or require an attention focused on some specific elements. Although those documents are printed texts, they assimilate more to Handwritten Text Recognition, because spaces and separations are harder to detect in those types of printed texts.<sup>18</sup> Nevertheless, the characters must be regular enough for a model to establish patterns.

On the contrary, cursive writing involves linked characters that can not be separated. It is usually complicated to know where one character ends and the other starts. This explains why the bounding boxes technique of OCR were not working for handwriting texts. Likewise, it could create issues for the n-grams. It was observed that it is the bigrams, and occasionally the trigrams, that are taken into account when learning from the training data. However, with cursive writing, it is possible that more than two or three characters are linked together. A token analysis with such cases would be much more difficult. Indeed, in the situation of my datasets, each version of an n-gram from the training data always looked the same. But in a dataset made of handwritten characters, the same token could be written differently, and they would not have the same n-grams sequences because of it.

---

18. See Figure 1.4 in the section 1.2



For example, in Figures 10.10a<sup>19</sup> and 10.10b,<sup>20</sup> the scripters used different ways to write the same token. This could be impactful for the recognition.

The name of a person, “d’herville”, is written in Figure 10.10a. In one version, the *h* has a curve at the end, *e*, *r*, *v*, and *i*, are unseparable from one another, as the cursive characters are joined, and the same can be said for the *l*, *l*, and *e*. In another version, the *h* is more straight, *e* and *r* are linked, *v* and *i* are completely disconnected, and the *lle* are in the same situation as previously. Therefore, by considering that recognition is done with sequences of characters, even for handwritten characters, the tokens would not be interpreted similarly. In the first case, it would be “h”|“ervi”|“lle”. In the second case, it would be “h”|“er”|“v” |“i” or “vi”|“lle”.

The name of the city of Paris is written in Figure 10.10b, but in different ways, although those two words are located on the same page. In both cases, the cutting of the words is “P”|“aris”, but the characters have different shapes. The first *P* looks like a drop cap, while the second one seems more regular. As for *a*, *r*, *i*, and *s*, there is variation in width and size. Here, the sequences of characters are the same, but the variation in writing could hinder the training.

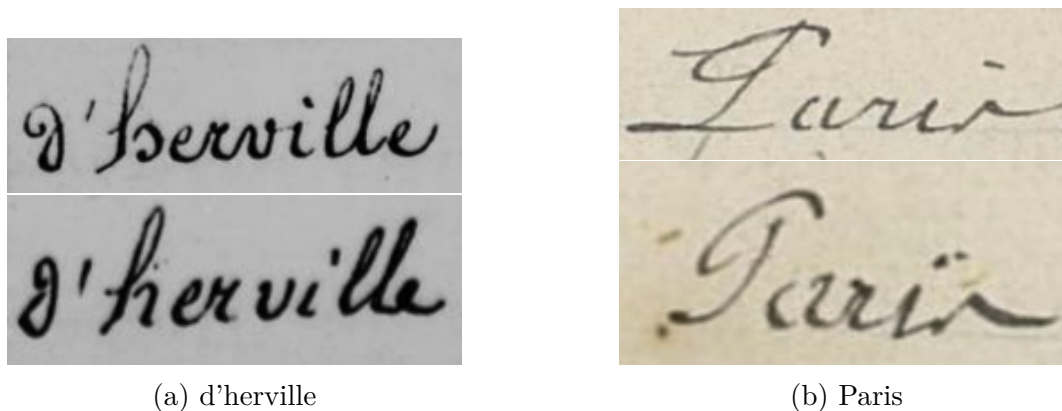


Figure 10.10: Two ways to handwrite a word in a same document

Considering what was observed during this thesis’ demonstration, the model would learn those groups. But what would it learn, actually? Repetitions of n-grams is what makes it good at recognition, because patterns are learned. With so many variations, can those n-grams really be seen as patterns? Additionally, with cursive writing and attached sequences, it is possible that bigrams could rarely be learned. This could diminish the

19. The document containing those images is [DAFANCH96\\_048MIC06487\\_L-0](#)

20. The document containing those images is [FRAN\\_0187\\_16401\\_L-1](#)

accuracy of the model. As I presented it previously in the conclusion, the quantity is also essential in the n-grams' efficacy. In those cases, it would be interesting to conduct token and token error analysis, adapted to handwriting.

Following an experiment similar to that of the sets War, Other and Ground Truth, with an observation of size difference, I could see if it reacts the same way. But while this experiment only required using the transcription of the images, this time images might be necessary. Indeed, some images' annotations that try to track the variations of pattern could be useful. This situation could likely be aggravated with the multilingualism. While adapted to the n-grams' impact, it is possible that it does not match well with handwriting either, because of regional handwriting variation.<sup>21</sup> Across countries, cursive is not taught the same. Some use more straight writing. Others can add some curve. Some put hooks or loops with their characters. This can create issues for the learning phase.

With single-language training data, the probabilities of having recurrent n-grams is higher because the combinations of characters are limited. But with multilingual training data, on top of having disparate characters' shaping, the quantity of combinations is almost endless. Although all versions will be transcribed with the same character, the model might have difficulties apprehending the n-grams with so many differences from one to the other. Would the multilingual model be able to be as efficient on recognition as the one I produced? Could the variations in writing and character combinations be too numerous to properly allow the model to learn patterns for the recognition? I could obtain some answers by conducting a token and token error analysis on another multilingual dataset, this time handwritten, with variations such as the one I mentioned.

To conclude, I can observe that I brought some first elements of response to the questions I had about the way neural network training operates. But the large domain that is Automatic Text Recognition still contains many uncharted territories. They would have to be explored furthermore, as well as be tested to see the influence of n-grams in those cases.

---

21. [https://en.wikipedia.org/wiki/Regional\\_handwriting\\_variation](https://en.wikipedia.org/wiki/Regional_handwriting_variation)



ANNEX I  
TABLES

# TABLES OF RESULTS (CONTENT ANALYSIS)

---

*Those tables were made in the context of the Content analysis. They provide the numeric distribution of the sets used in the experiment, in the various shape (tokens, lemmas, part-of-speech) and subsets (common, unique to war, unique to other) they were produced.*

Table A.1: Table for the words present in the sets

	War	Other
Token	1813	3603
Lemma	1464	2718
Difference	349	885

Table A.3: Table for the words uniquely present in each set and in common

	War	Other	Common
Token	866	2656	947
Lemma	563	1817	901
Difference	303	839	46

Percentage of common words (tokens) in the total of words (war + other): 17,49%

Percentage of common words (lemmas) in the total of words (war + other): 21,54%

Table A.5: Table of the POS for the words present in the sets (tokens)

	War	Other
ADJ	390	860
ADP	35	41
ADV	108	175
AUX	42	69
CCONJ	7	13
DET	33	49
NOUN	559	1077
NUM	14	23
PRON	36	57
PROPN	46	133
PUNCT	1	0
SCONJ	5	6
VERB	530	1098
X	7	2
Total	1813	3603

Table A.7: Table of the POS for the words present in the sets (lemmas)

	War	Other
ADJ	304	598
ADP	31	45
ADV	107	166
AUX	8	20
CCONJ	7	13
DET	19	37
NOUN	502	926
NUM	15	17
PRON	34	41
PROPN	45	127
PUNCT	2	0
SCONJ	4	4
VERB	386	724
X	0	0
Total	1464	2718

Table A.9: Table of the POS for the words uniquely present in each set and in common (tokens)

	War	Other	Common
ADJ	225	722	186
ADP	4	17	29
ADV	36	104	77
AUX	12	60	35
CCONJ	0	4	7
DET	4	13	32
NOUN	254	759	295
NUM	5	6	13
PRON	10	19	30
PROPN	25	100	20
PUNCT	0	0	1
SCONJ	0	0	4
VERB	289	842	217
X	2	10	1
Total	866	2656	947

Table A.11: Table of the POS for the words uniquely present in each set and in common (lemmas)

	War	Other	Common
ADJ	126	452	170
ADP	5	17	27
ADV	39	99	72
AUX	5	12	5
CCONJ	0	4	6
DET	4	12	15
NOUN	199	616	307
NUM	2	6	13
PRON	4	14	23
PROPN	30	99	20
PUNCT	0	1	1
SCONJ	0	0	5
VERB	146	475	237
X	3	9	0
Total	563	1816	901

# METRICS (COMPARATIVE ANALYSIS)

---

*Those tables provide the metrics produced during the Comparative analysis experiment. Three models were applied during this experiment: Model Other (MO), Model War (MW) and Model War Retrained (MWR). Two sets were used: Set Other (SO) and Set War (SW). The models were applied to the sets as a whole, then on specific pages from each set.*

## B.1 Metrics by Letters for the Set Other

Table B.1: Metrics for the models applied to the letter 607 page 3

	Model Other	Model War	Model War Retrained
Levenshtein Distance (Char.)	5	25	19
Levenshtein Distance (Words)	2	22	16
Word Error Rate (WER in %)	0.803	8.835	6.451
Char. Error Rate (CER in %)	0.316	1.582	1.204
Word Accuracy (Wacc in %)	99.196	91.164	93.548
Hits	1576	1557	1558
Substitutions	0	18	18
Deletions	4	5	1
Insertions	1	2	0
Total char. in reference	1580	1580	1577
Total char. in prediction	1577	1577	1576

Table B.3: Metrics for the models applied to the letter 607 page 17

	Model Other	Model War	Model War Retrained
Levenshtein Distance (Char.)	10	90	84
Levenshtein Distance (Words)	8	71	62
Word Error Rate (WER in %)	1.941	17.233	15.012
Char. Error Rate (CER in %)	0.413	3.72	3.466
Word Accuracy (Wacc in %)	98.058	82.766	84.987
Hits	2414	2334	2344
Substitutions	3	66	63
Deletions	2	19	16
Insertions	5	5	5
Total char. in reference	2419	2419	2423
Total char. in prediction	2422	2405	2412

Table B.5: Metrics for the models applied to the letter 722 page 1

	Model Other	Model War	Model War Retrained
Levenshtein Distance (Char.)	9	65	61
Levenshtein Distance (Words)	6	37	37
Word Error Rate (WER in %)	3.947	24.342	24.342
Char. Error Rate (CER in %)	1.032	7.454	6.995
Word Accuracy (Wacc in %)	96.052	75.657	75.657
Hits	866	811	814
Substitutions	5	61	54
Deletions	1	0	4
Insertions	3	4	3
Total char. in reference	872	872	872
Total char. in prediction	874	876	871



Table B.7: Metrics for the models applied to the letter 1170 page 3

	Model Other	Model War	Model War Retrained
Levenshtein Distance (Char.)	11	56	55
Levenshtein Distance (Words)	7	44	46
Word Error Rate (WER in %)	2.661	16.73	17.49
Char. Error Rate (CER in %)	0.684	3.484	3.422
Word Accuracy (Wacc in %)	97.338	83.269	82.509
Hits	1597	1553	1553
Substitutions	7	43	43
Deletions	3	11	11
Insertions	1	2	1
Total char. in reference	1607	1607	1607
Total char. in prediction	1605	1598	1597

Table B.9: Metrics for the models applied to the letter 1358 page 4

	Model Other	Model War	Model War Retrained
Levenshtein Distance (Char.)	7	127	121
Levenshtein Distance (Words)	6	50	43
Word Error Rate (WER in %)	8.955	74.626	64.179
Char. Error Rate (CER in %)	1.369	24.853	23.679
Word Accuracy (Wacc in %)	91.044	25.373	35.82
Hits	504	386	393
Substitutions	4	116	109
Deletions	3	9	9
Insertions	0	2	3
Total char. in reference	511	511	511
Total char. in prediction	508	504	505

## B.2 Metrics by Letters for the Set War

Table B.11: Metrics for the models applied to the letter 678 page 1

	Model War	Model Other	Model War Retrained
Levenshtein Distance (Char.)	11	33	8
Levenshtein Distance (Words)	9	23	7
Word Error Rate (WER in %)	5.202	13.294	4.046
Char. Error Rate (CER in %)	1.038	3.116	0.755
Word Accuracy (Wacc in %)	94.797	86.705	95.953
Hits	1048	1028	1051
Substitutions	10	24	7
Deletions	1	7	1
Insertions	0	2	0
Total char. in reference	1059	1059	1059
Total char. in prediction	1058	1054	1058

Table B.13: Metrics for the models applied to the letter 844 page 1

	Model War	Model Other	Model War Retrained
Levenshtein Distance (Char.)	21	40	6
Levenshtein Distance (Words)	20	32	7
Word Error Rate (WER in %)	6.734	10.774	2.356
Char. Error Rate (CER in %)	1.213	2.312	0.346
Word Accuracy (Wacc in %)	93.265	89.225	97.643
Hits	1709	1700	1724
Substitutions	17	25	4
Deletions	4	5	2
Insertions	0	10	0
Total char. in reference	1730	1730	1730
Total char. in prediction	1726	1735	1728

Table B.15: Metrics for the models applied to the letter 948 page 1

	Model War	Model Other	Model War Retrained
Levenshtein Distance (Char.)	27	59	9
Levenshtein Distance (Words)	20	35	8
Word Error Rate (WER in %)	10.928	19.125	4.371
Char. Error Rate (CER in %)	2.423	5.296	0.807
Word Accuracy (Wacc in %)	89.071	80.874	95.628
Hits	1089	1059	1105
Substitutions	16	45	7
Deletions	9	10	2
Insertions	2	4	0
Total char. in reference	1114	1114	1114
Total char. in prediction	1107	1108	1112

Table B.17: Metrics for the models applied to the letter 1000 page 3

	Model War	Model Other	Model War Retrained
Levenshtein Distance (Char.)	8	6	9
Levenshtein Distance (Words)	8	4	8
Word Error Rate (WER in %)	2.807	1.403	2.807
Char. Error Rate (CER in %)	0.477	0.357	0.536
Word Accuracy (Wacc in %)	97.192	98.596	97.192
Hits	1669	1671	1668
Substitutions	7	6	9
Deletions	1	0	0
Insertions	0	0	0
Total char. in reference	1677	1677	1677
Total char. in prediction	1676	1677	1677

Table B.19: Metrics for the models applied to the letter 1367 page 1

	Model War	Model Other	Model War Retrained
Levenshtein Distance (Char.)	19	50	13
Levenshtein Distance (Words)	18	38	13
Word Error Rate (WER in %)	6.545	13.818	4.727
Char. Error Rate (CER in %)	1.172	3.086	0.802
Word Accuracy (Wacc in %)	93.454	86.181	95.272
Hits	1603	1574	1607
Substitutions	14	41	11
Deletions	3	5	2
Insertions	2	4	0
Total char. in reference	1620	1620	1620
Total char. in prediction	1619	1619	1618

# TABLES OF RESULTS (TOKEN ANALYSIS)

---

## C.1 Total

*Those tables are the results of the distribution of n-grams obtained during the Token Analysis, with a division for each set: War, Other, and Ground Truth (GT). Some additional divisions, given by numbers and percentages, are also available, with a partitioning for each unit between the n-gram in all capitals (AC), with initials (I), and in lowercases (L).*

Table C.1: All tokens

	War	Other	GT
4grams	4949	12751	69883
3grams	7613	19702	108922
2grams	13637	35235	196585

Table C.3: All tokens (by type) (numbers)

	War	Other	GT
4grams (AC)	133	191	2744
4grams (I)	481	1392	6854
4grams (L)	4335	11168	60285
3grams (AC)	195	286	4153
3grams (I)	528	1523	7512
3grams (L)	6890	17893	97257
2grams (AC)	226	489	7233
2grams (I)	641	1831	9094
2grams (L)	12770	32915	180258

Table C.5: All tokens (by type) (percentages)

	War	Other	GT
4grams (AC)	2%	1%	4%
4grams (I)	10%	11%	10%
4grams (L)	88%	88%	86%
3grams (AC)	2%	1%	4%
3grams (I)	7%	8%	7%
3grams (L)	91%	91%	89%
2grams (AC)	2%	1%	4%
2grams (I)	5%	5%	5%
2grams (L)	93%	93%	92%

## C.2 Set Other

*Those tables, retrieving results from the n-gram division of the set Other, render the distribution, by level of occurrences, of the n-gram in its various forms. They also give the amount of the most and least popular n-grams.*

Table C.7: Tokens from set Other (All Caps)

	2grams	3grams	4grams
1	57	121	99
2 to 5	63	34	17
6 to 10	14	7	4
11 to 50	9	2	2
Total	143	164	122

Table C.9: Tokens from set Other (Initials)

	2grams	3grams	4grams
1	34	158	218
2 to 5	40	150	160
6 to 10	39	38	38
11 to 50	44	33	25
51 to 100	5	∅	∅
More than 100	1	∅	∅
Total	163	379	441

Table C.11: Tokens from set Other (Lowercases)

	2grams	3grams	4grams
1	61	390	1005
2 to 5	75	543	924
6 to 10	30	238	243
11 to 50	113	332	214
51 to 100	42	46	15
101 to 500	83	20	4
501 to 1000	10	∅	∅
More than 1000	2	∅	∅
Total	416	1569	2405

Table C.13: Tokens from set Other (11 and more)    Table C.15: Tokens from set Other (Only 1)

	2grams	3grams	4grams
11 and more	309	433	260
Total	722	2112	2968
%	43%	21%	9%

	2grams	3grams	4grams
1	152	669	1322
Total	722	2112	2968
%	21%	32%	45%

### C.3 Set War

*Those tables present the same elements as the previous section, but for the set War.*

Table C.17: Tokens from set War (All Caps)

	2grams	3grams	4grams
1	48	68	58
2 to 5	40	22	13
6 to 10	11	4	4
11 to 50	6	2	1
Total	105	96	76

Table C.19: Tokens from set War (Initials)

	2grams	3grams	4grams
1	32	101	119
2 to 5	43	61	58
6 to 10	10	12	9
11 to 50	20	10	8
Total	105	184	194

Table C.21: Tokens from set War (Lowercases)

	2grams	3grams	4grams
1	69	418	818
2 to 5	68	425	534
6 to 10	42	166	97
11 to 50	110	134	58
51 to 100	40	11	1
101 to 500	36	4	
Total	365	1158	1508

Table C.23: Tokens from set War (11 and more)

	2grams	3grams	4grams
11 and more	211	161	68
Total	575	1438	1778
% on the total	37%	11%	4%

Table C.25: Tokens from set War (Only 1)

	2grams	3grams	4grams
1	149	587	995
Total	575	1438	1778
% on the total	26%	41%	56%

## C.4 Set Ground Truth

*Those tables present the same elements as the previous section, but for the set Ground Truth.*

Table C.27: Tokens from set Ground Truth  
(All Caps)

	2grams	3grams	4grams
1	27	365	528
2 to 5	79	359	303
6 to 10	34	81	34
11 to 50	101	53	29
51 to 100	25	9	3
More than 100	15	3	2
Total	281	870	899

Table C.29: Tokens from set Ground Truth  
(Initials)

	2grams	3grams	4grams
1	29	287	515
2 to 5	54	273	356
6 to 10	24	99	95
11 to 50	68	107	96
51 to 100	27	16	13
More than 100	22	13	10
Total	224	795	1085



Table C.31: Tokens from set Ground Truth (Lowercases)

	2grams	3grams	4grams
1	76	440	1349
2 to 5	88	531	1510
6 to 10	48	311	602
11 to 50	89	650	907
51 to 100	33	207	149
101 to 500	124	203	79
501 to 1000	44	17	8
More than 1000	52	7	
Total	554	2366	4604

Table C.33: Tokens from set Ground Truth  
(11 and more)

	2grams	3grams	4grams
11 and more	600	1285	1296
Total	1059	4031	6588
% on the total	57%	32%	20%

Table C.35: Tokens from set Ground Truth  
(Only 1)

	2grams	3grams	4grams
1	132	1092	2392
Total	1059	4031	6588
% on the total	12%	27%	36%

## C.5 Comparison

*Those tables render the difference in the distribution of the most and least popular n-grams between the sets War and Other, War and Ground Truth, Other and Ground Truth, Elements unique to War and Ground Truth, Elements unique to Other and Ground Truth, and Common elements to War and Other and Ground Truth.*

Table C.37: Most popular tokens (11 and more) (numbers)

	War	Other	Common
2grams	0	98	211
3grams	5	277	156
4grams	7	199	61

	War	GT	Common
2grams	0	389	211
3grams	0	1124	161
4grams	0	1228	68

	Other	GT	Common
2grams	2	293	307
3grams	7	859	426
4grams	10	1046	250

	War_unique	GT	Common
2grams	0	600	0
3grams	0	1280	5
4grams	0	1289	7

	Other_unique	GT	Common
2grams	2	504	96
3grams	7	1015	270
4grams	10	1107	189

	Common_unique	GT	Common
2grams	0	389	211
3grams	0	1129	156
4grams	0	1235	61

Table C.44: Most popular tokens (11 and more) (percentages)

	War	Other
2grams	0%	32%
3grams	3%	64%
4grams	10%	77%

	War	GT
2grams	0%	64%
3grams	0%	88%
4grams	0%	95%

	Other	GT
2grams	0,6%	49%
3grams	1,6%	67%
4grams	3,8%	81%

	War_unique	GT
2grams	0%	100%
3grams	0%	99,6%
4grams	0%	99,5%

	Other_unique	GT
2grams	2%	84%
3grams	2,5%	79%
4grams	5%	85%

	Common_unique	GT
2grams	0%	65%
3grams	0%	88%
4grams	0%	95%

Table C.51: Least popular tokens (Only 1 occurrence) (numbers)

	War	Other	Common
2grams	120	123	29
3grams	475	557	112
4grams	801	1128	194

	War	GT	Common
2grams	139	122	10
3grams	543	1048	44
4grams	910	2307	85

	Other	GT	Common
2grams	144	124	8
3grams	589	1012	80
4grams	1122	2192	200

	War_unique	GT	Common
2grams	112	124	8
3grams	435	1052	40
4grams	730	2321	71

	Other_unique	GT	Common
2grams	117	126	6
3grams	481	1016	76
4grams	942	2206	186

	Common_unique	GT	Common
2grams	27	130	2
3grams	108	1088	4
4grams	180	2378	14

Table C.58: Least popular tokens (Only 1 occurrence) (percentages)

	War	Other
2grams	81%	81%
3grams	81%	83%
4grams	81%	85%

	War	GT
2grams	93%	92%
3grams	93%	96%
4grams	91%	96%

	Other	GT
2grams	95%	94%
3grams	88%	92%
4grams	85%	92%

	War_unique	GT
2grams	93%	94%
3grams	92%	96%
4grams	91%	97%

	Other_unique	GT
2grams	95%	95%
3grams	86%	93%
4grams	84%	92%

	Common_unique	GT
2grams	93%	98%
3grams	96%	99,6%
4grams	98%	99%

# COMPARISON TRANSCRIPTION (TOKEN ERROR ANALYSIS)

---

*Those tables detail the differences observed on some lines of the specific pages selected with the application of the model War, Other, and Ground Truth on the test sets, compared to the manual transcription of the pages. The last column indicates when there was a specific situation within the facsimile that might have prompted the prediction error.*

Line	Manuel Transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.3	tranquillité	tranquilité	tranquimlité	tranquilbité	two-character overlay
1.5	Tout	Lout			
1.8	XXX	EXX			two-character overlay
1.8	tances			ttances	
1.10	de	ss		qa	two-character overlay
1.10	base	pase			
1.13	Chamvres			cham/res	manual addition of characters
1.16	barricades	parricades			
1.16	brusquement	brusquenent			
1.17	Députés	péputés			
1.19	hensible	nensible			
1.20	rellement	relloment			
1.20	mécontentement	mécontentenent			
1.21	bizarre	bigarre			
1.21	exploitée			explottée	two-character overlay
1.22	partisans	parisans			
1.24	Grand	Crand			
1.25	Poincariste	poincariste			
1.25	Clémenciste			Clémenciste	
1.26	bénéfice	pénéfice			

Table D.1: Prediction errors of the letter 607 page 3 (set Other)

Line	Manuel transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.3	circulaire	ciroulaire			
1.6	accusant	accusaut			
1.7	chacun	Chaoun			
1.7	condamnation	Condamnation			
1.9	manifestement	maniestement			
1.11	Restait	Bestait			
1.11	ici	joi			
1.14	bonheur	bonpeur			
1.14	succès	sucès			
1.15	Que	que			
1.16	tôt	tdt			
1.18	souhaitaient	sounaitaient			
1.19	titre	titfe			two-character overlay
1.20	parler	Parler			
1.22	Septem	septen			
1.23	Octobre	octobre		octobre	
1.26	Moi	Noi			
1.27	dois	cois			
1.27	monde	nonde			
1.27	reconnaît	reconnatt	reconnatt		
1.29	ceux	Ceux			
1.29	mêmes	mênes			
1.29	garder	Parde		parde	problem with the segmentation
1.30	premier	prenten			

1.31	carrière	carrisre			slightly bent sheet
1.31	fil	fila			slightly bent sheet
1.32	Bolo			Rolo	
1.33	ces	Ces			
1.33	Messieurs	sessieurs			
1.33	fui	rui			
1.34	Mais	sais			
1.35	fallu	failu		fall	problem with the segmentation
1.36	Répondre	hépondre			
1.37	infâmante	infêmante		infamante	
1.38	Un	In			
1.38	débat	dépat			
1.39	fût	ôt			
1.39	préféré	préréré			
1.41	yeux	veux			
1.41	beau	neau			
1.41	rôle	pôle			

Table D.2: Prediction errors of the letter 607 page 17 (set Other)

Line	Manuel transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.3	PARIS	PARIô			
1.3	Novembre	Noyembre			
1.4	MISS	MIôS			light transparency of the paper
1.4	MARGARET	MandenNr		MARCARET	light transparency of the paper

1.4	ALEXANDER	ALETespRR	ALEWANDER		light transparency of the paper
1.4	PARIS	PaRl	PAuIC		light transparency of the paper
1.5	cher		chef		
1.6	nouveau	nouvebu			
1.7	Albanie	Alaunie			
1.7	où	ou			
1.8	Croix	Cryix	croix		
1.8	Rouge	mouge			
1.8	américaine			américains	
1.8	pensez	pamsez			light transparency of the paper
1.9	dû	dù			
1.10	jeunesse	jounesse			
1.10	faut	fut			
1.11	et	e			two-character overlay
1.11	Je	de			
1.12	recommande	Locommande			
1.13	isolée	jsclée			
1.14	Miss	Nis			
1.14	Alexander	Aloxander			
1.14	descendue	descenque		descenque	
1.14	Pierre	Lierre			
1.15	nous	pous			
1.15	accueillons	acusillons			
1.16	sommes	sonmes			
1.16	notre			hotre	



1.16	bureau	hureau			
1.17	foyer	loyer		Jover	
1.17	modeste	mojeste			
1.18	harmonie	harmouie			
1.19	Haskell	Mastell			
1.20	Murray	Muvray			

Table D.3: Prediction errors of the letter 722 page 1 (set Other)

Line	Manuel transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.2	Mais	bais			
1.2	aspirations	ashirations			
1.2	géné	séné			
1.4	humaine	pumaine			
1.5	ainsi	sinsi			
1.6	çais	Cais			
1.6	soumis	sonmis			
1.10	Seine	seine			
1.11	Montagne	Hontagne			
1.11	Sainte	aainte			
1.13	Rien	hien			
1.13	Quoi	coi		Cooi	
1.13	donc	dondt			
1.14	sans	ans			light transparency of the paper
1.15	faveur			fuveur	

1.15	chrétiennes	chrétienes			
1.16	Orient	crient		orient	
1.16	su	au			
1.17	Islam	felas		Islan/Islam	
1.18	sang	sans			
1.18	par	Ber			
1.20	a	s			
1.21	Son	son			
1.22	inaperçu	insperdu		inaperqu	
1.22	agitation	asitation			
1.26	éphémères	Cphémères		éphénères	
1.26	commises	comises			
1.26	toujours	toujons		toujorrs	
1.27	conserve	couserve			

Table D.4: Prediction errors of the letter 1170 page 3 (set Other)

Line	Manual Transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.3	Murray	Muray			
1.4	Sénateur	sénatour			
1.4	ESTOURNELLES	RsfduRLIEs	ESTOURNELES		
1.4	CONSTANT	CoNaIAN			
1.5	Baron	Maron			
1.5	ADELSWARD	ADELSyARD			
1.6	Recteur	Becteur		Becteur	

1.6	APPELL	ApLNIT			
1.7	SHOTWELL	SHCTNPII	SuOTVELL	SHOTUWELL	
1.8	SANGRO	SANCh0	SANCRO	SANCRO	
1.8	OLANO	CIA440			
1.9	Professeur	professeur			
1.9	REDLICH	BETLION			
1.10	Hellmuth	Mellmuth		Mellmuth	
1.10	GERLACH	GERLACN			
1.11	Professeur	professeur			
1.11	FOERSTER	FOFRCTRR			
1.12	Professeur	professeur			
1.12	GIDE	GILR			
1.13	EFREMOFF	NENEMOTT		EEREMOFF	
1.14	Député	péputé		Ddéputé	
1.14	Justin	Dustin			
1.14	GODART	dopAR			
1.15	Sénateur	sénateur			
1.15	LA	La			
1.15	FONTAINE	FoNLAIRR	FONTATNE		
1.16	Professeur	Frofesseur			
1.16	Henri	Benri			
1.16	LICHTENBERGER	LIONTERERdER		LICHTENRERGER	
1.17	LEJEUNE	LETEUEE	LEJEUE	LETEUE	
1.17	Représentant	Beprésentant			
1.17	Albert	Alhert			

1.17	THOMAS	Lacns		TRORAS	two-character overlay
1.18	NIPPOLD	NIPPGLR			
1.19	JAUDON	TANDON			
1.20	CONVERSET	CoNvERSET			
1.21	ESTOURNELLES	EsfouRETIES			
1.21	CONSTANT	CoRsTANT			
1.22	Professeur	Frofesseur			
1.22	Th	lh			
1.22	RUYSSSEN	BUTSsRN		BUYSSSEN	
1.23	Professeur	Profeseur			
1.23	PRUDHOMMEAUX	PRpRONMURAU	PUDHOMMEAUX		
1.24	DANDIEU	BANRIES			

Table D.5: Prediction errors of the letter 1358 page 4 (set Other)

Line	Manuel transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.1	LETTRE		LETE		
1.2	PARIS		PAaIS		
1.2	Juin			ruin	
1.4	ON		OE		
1.4	PARLE		PAULE		
1.4	PLUS	LUS			
1.4	AFFAIRE		AFATE	APPAIRE	
1.4	CAILLAUX	CAILLAUT	CAILTAU	CAILAU	
1.6	CAVALLINI	CAVAIIINI	SAvALIEL		

1.6	VIENT		VIET		
1.6	ETRE		ETSE		
1.6	ACQUITTE		ACOUITT		
1.8	affaire	arfaire			
1.8	Caillaux	caillaux			
1.8	appartient	apbartient	appartlent		
1.9	Son		Sen	Sen	manual addition of characters
1.11	ACTION		A0TTON		
1.11	FRANCAISE	PRANCAISN			
1.12	parle		pafle		
1.12	accusateurs		asgusateurs	ansusateurs	two-character overlay
1.14	Rome			Bome	
1.15	Public			public	
1.20	On		on		

Table D.6: Prediction errors of the letter 678 page 1 (set War)

Line	Manuel transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.3	PARIS		PARrS		
1.3	le		Ce	ee	two-character overlay
1.4	CONTINUE	CONTINUR	CONTHRE		
1.5	IMPORTUN	INTONTUN	LÉTORCUE		
1.5	ET	NT	EX		
1.5	CONTRE	GONTRE	CONTE		
1.5	NATURE	NATRE	FATURE		

1.8	Sarthe	sarthe			
1.11	pas	bas			
1.13	printemps			brintemps	
1.15	humaines	hnmaines			
1.15	laisseraient	laisséraient	laisséraient	laisssraient	
1.18	mais		Mais		
1.18	non		nom	nun	
1.18	est			esit	
1.19	cependant			cebendant	
1.20	haute			huaute	
1.20	tenue		tenque	tenque	
1.21	infini		infri	infri	
1.22	çaise	Caise	caise		
1.24	guerre	guerrs			
1.25	pourquoi	pourquci		pourççoi	two-character overlay
1.27	aperçoit	apercoit			
1.28	Certes		Gertes	Certeos	
1.31	hommage		hommge		
1.32	Murray			Murrey	
1.32	BUTLER		EUTLER		

Table D.7: Prediction errors of the letter 844 page 1 (set War)

Line	Manuel transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.1	LETTRE		LEVTRR		

1.4	LE		VE	EN	
1.4	DÉSARMEMENT	DÉSARMEEMRNT	FÉSARMEENT	DÉSARUERGENT	
1.4	ALLEMAGNE		ATLACR		
1.4	GERLACH	GRLACN	DEULACS	CERLAGH	
1.5	FOERSTER	FOERSTR	gOEMCIER	POERSTER	
1.5	RÉPONSE	BÉPONSRE	RUFONSR	RÉPONRE	
1.5	à		A		
1.5	VOTRE		VORE	VOTRN	
1.5	LETTRE	LETERE	LUTRE		
1.5	DÉCEMBRE	DÉRSRE	FÉORÉERE	DÉGEMBRE	
1.6	cher		cuer		
1.6	Butler			mutler	
1.7	moral			aoral	
1.7	désarmement	désrmement	désaurmement		
1.7	Allemagne	Allemage	Allesague		
1.8	nos		qos		
1.8	préoccupations	préoccupstions	préoccupatious	préecoupations	
1.9	parlé		parté		
1.10	compatriotes		compEtriotes		
1.10	Professeur	Frofesseur			
1.11	Central			Genral	
1.12	Homme			homme	
1.13	Kessler		Fessler		
1.14	Kessler		Fessler	Nessler	
1.14	appris		apris		

1.15	représentants			représontants	writing slightly faded
1.16	Frioul	Frjoul		Prioul	
1.16	chez	ches			
1.17	propriétés		probriétés		
1.17	Lozère		Lomère		
1.18	facilement	fadilement			
1.19	Parlement	parlement			
1.19	Chambrun	Ghamrun			
1.19	Lascazes	lascages	lascames		
1.19	notamment		notemment	notament	
1.23	Monsieur			Momsieur	
1.23	Nicholas		Nichouas		
1.23	BUTLER		EUTLER		

Table D.8: Prediction errors of the letter 948 page 1 (set War)

Line	Manuel transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.1	LETTRE		LETE		
1.2	PARIS		PAaIS		
1.2	Juin			ruin	
1.4	ON		OE		
1.4	PARLE		PAULE		
1.4	PLUS	LUS			
1.4	AFFAIRE		AFATE	APPAIRE	
1.4	CAILLAUX	CAILLAUT	CAILTAU	CAILAU	



1.6	CAVALLINI	CAVAIIINI	SAvALIEL		
1.6	VIENT		VIET		
1.6	ETRE		ETSE		
1.6	ACQUITTE		ACOUITT		
1.8	affaire	arfaire			
1.8	Caillaux	caillaux			
1.8	appartient	apbartient	appartlent		
1.9	Son		Sen	Sen	manual addition of characters
1.9	lettres			lattres	two-character overlay
1.9	York			Nork	two-character overlay
1.10	Certaines			Gertaines	two-character overlay
1.11	ACTION		A0TTON		
1.11	FRANCAISE	PRANCAISN			
1.12	parle		pafle		
1.12	quand		Chand	dhand	two-character overlay
1.12	vous	Vous			
1.12	Verdun	verdun		verdun	
1.13	Ardennes	Ardenes			
1.13	Vosges	vosges		vosges	
1.14	Rome			Bome	
1.15	Public			public	
1.15	lutttes			luetes	two-character overlay
1.20	On		on		
1.20	noble		noRke	noère	two-character overlay
1.20	Défense	péfense		péfense	

1.24	franco	Franco	granco		
1.26	éphémères	Cphémères		éphénères	
1.26	commises	comises			
1.26	toujours	toujons		toujorrs	
1.27	conserve	couserve			
1.27	apothéose	apothécse			

Table D.9: Prediction errors of the letter 1000 page 3 (set War)

Line	Manuel transcription	Model War	Model Other	Model GT	Facsimile Particularity
1.1	LETTRE		AETOE	RETTRE	
1.3	Créans		créans		
1.3	Août		Acût		
1.4	EN	N	NE		
1.4	ALLEMAGNE	ALLERMAGNE	ALEWAGE	ALLEMASNE	
1.7	encourageons	encouragecns			
1.9	lettres			lattres	two-character overlay
1.9	York			Nork	two-character overlay
1.10	Certaines			Gertaines	two-character overlay
1.12	manifestation		manigestation		
1.12	quand		Chand	dhand	two-character overlay
1.12	vous	Vous			
1.12	Verdun	verdun		verdun	
1.13	Ardennes	Ardenes			
1.13	Vosges	vosges		vosges	

l.15	Voici	Vojoi	voici		
l.15	pour	nour	mour		
l.15	décla		déCla		
l.17	guerre		querre		
l.17	là		lâ		
l.18	misères	Bisèrest			
l.19	Est	Rst	Hst		
l.19	dixième		dixième		
l.20	GT	ncon/Pparmi/npations	<i>unintelligible</i>	<i>unintelligible</i>	narrow polygon/little line space
l.21	GT		<i>unintelligible</i>	<i>unintelligible</i>	narrow polygon/little line space
l.25	invitons		invitonë		
l.26	cimetière	cimetjère	cimetdère		
l.26	matin			mntin	
l.28	répudiation		répadiation		
l.29	poussent		pouscent		
l.29	guerre	suerre	Sierre	Suerre	
l.32	français		français		
l.32	Orateurs		orateurs	orateurs	
l.34	Murray	MMurray			

Table D.10: Prediction errors of the letter 1367 page 1 (set War)

# TABLES OF N-GRAMS (TOKEN ERROR ANALYSIS)

---

*The tables presented here are from the Token Error Analysis. They render, line by line, the errors retrieved from the pages of d'Estournelles' dataset, to which the model War (MW), Other (MO), and Ground Truth (MGT) were applied.*

*A set of three columns for each model is displayed. The first column show the n-gram(s) wrongly predicted. The second column renders the number of occurrences of the n-gram of the reference in the training data of the model. The third column renders the number of occurrences of the n-gram of the prediction in the training data of the model.*

*As for the colours filling the cells, red indicates that the error doesn't exist in the model, blue that reference and prediction do not have the same number of characters, grey that the n-gram is not the right unit of characters for the table, and green that the prediction occurrence was greater than that of the reference.*

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
tr, an, qu, il, li, té	tr, an, qu, il, ai, té	42	135	tr, an, qu, im, li, té	265	67	tr, an, qu, il, bi, té	871	513
To, ut	Lo, ut	4	5						
ta, nc, es							tt, an, ce, s		
de	ss	442	62				qa	6569	∅
ba, se	pa, se	7	216						
Ch, am, vr, es							ch, am, /r, es	85 275	1120 ∅
ba, rr, ic, ad, es	pa, rr, ic, ad, es	7	216						
br, us, qu, em, en, t	br, us, qu, en, en, t	58	365						
Dé, pu, té, s	pé, pu, té, s	7	10						
he, ns, ib, le	ne, ns, ib, le	16	107						
re, ll, em, en, t	re, ll, om, en, t	58	5						
mé, co, nt, en, te, me, nt	mé, co, nt, en, te, ne, nt	173	107						
bi, za, rr, e	bi, ga, rr, e	∅	20						
ex, pl, oi, té, e							ex, pl, ot, té, e	650	92
pa, rt, is, an, s	pa, ri, sa, ns								
Gr, an, d	Cr, an, d	2	1						
Po, in, ca, ri, st, e	po, in, ca, ri, st, e	5	116						
Cl, ém, en, ci, st, e							Cl, ém, en, ci, st, e	696	∅
bé, né, fi, ce	pé, né, fi, ce	1	10						
ci, rc, ul, ai, re	ci, ro, ul, ai, re	14	19						
ac, cu, sa, nt	ac, cu, sa, ut	241	84						
ch, ac, un	Ch, ao, un	65 35	7 ∅						
co, nd, am, na, ti, on	Co, nd, am, na, ti, on	212	17						
ma, ni, fe, st, em, en, t	ma, ni, es, te, me, nt								
Re, st, ai, t	Be, st, ai, t	1	9						
ic, i	jo, i	32	38						
bo, nh, eu, r	bo, np, eu, r	1	∅						
su, cc, ès	su, cè, s								
Qu, e	qu, e	7	265						

Figure E.1: Bigrams Page 1 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
tô, t	td, t	4	∅						
so, uh, ai, ta, ie, nt	so, un, ai, ta, ie, nt	∅	132						
ti, tr, e	ti, tf, e	142	∅						
pa, rl, er	Pa, rl, er	216	16						
Se, pt, em	se, pt, en	1 58	129 365						
Oc, to, br, e	oc, to, br, e	1	12				oc, to, br, e	21	171
Mo, i	No, i	27	15						
do, is	co, is	31	212						
mo, nd, e	no, nd, e	107	134						
re, co, nn, ai, t	re, co, nn, at, t	3	70	re, co, nn, at, t	9	165			
ce, ux	Ce, ux	178	13						
mê, me, s	mê, ne, s	173	107						
ga, rd, er	Pa, rd, e						pa, rd, e		
pr, em, ie, r	pr, en, te, n	58 91	365 115						
ca, rr, iè, re	ca, rr, is, re	13	186						
fi, ls	fi, la	1	239						
Bo, lo							Ro, lo	58	60
ce, s	Ce, s	178	13						
Me, ss, ie, ur, s	se, ss, ie, ur, s	1	129						
fu, i	ru, i	7	8						
Ma, is	sa, is	20	95						
fa, ll, u	fa, il, u	55	96				fa, ll		
Ré, po, nd, re	hé, po, nd, re	1	5						
in, fâ, ma, nt, e	in, fê, ma, nt, e	∅	2				in, fa, ma, nt, e	10	1034
Un	In	8	∅						
dé, ba, t	dé, pa, t	7	216						
fû, t	ôt								
pr, éf, ér, é	pr, ér, ér, é	∅	17						
ye, ux	ve, ux	11	63						

Figure E.2: Bigrams Page 2 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
be, au	ne, au	12	107						
rô, le	pô, le	1	∅						
PA, RI, S	PA, RI, ô								
No, ve, mb, re	No, ye, mb, re	63	11						
MI, SS	MI, ôS	∅	∅						
MA, RG, AR, ET	Ma, nd, en, Nr	4 ∅ ∅ 6	20 94 365 ∅				MA, RC, AR, ET	2	7
AL, EX, AN, DE, R	AL, ET, es, pR, R	∅ 3 5	6 271 ∅	AL, EW, AN, DE, R	2	∅			
PA, RI, S	Pa, RI			PA, ul, C	9	92			
ch, er				ch, ef	355	21			
no, uv, ea, u	no, uv, eb, u	6	∅						
Al, ba, ni, e	Al, au, ni, e	7	123						
où	ou	10	79						
Cr, oi, x	Cr, yi, x	43	∅	cr, oi, x	12	64			
Ro, ug, e	mo, ug, e	4	107						
am, ér, ic, ai, ne							am, ér, ic, ai, ns	1420	2388
pe, ns, ez	pa, ms, ez	84 173	216 ∅						
dù	dù	∅	∅						
je, un, es, se	jo, un, es, se	58	38						
fa, ut	fu, t								
et	e								
Je	de	24	442						
re, co, mm, an, de	Lo, co, mm, an, de	302	5						
is, ol, ée	js, cl, ée	186 30	∅ 13						
Mi, ss	Ni, s								
Al, ex, an, de, r	Al, ox, an, de, r	33	∅						
de, sc, en, du, e	de, sc, en, qu, e	59	265				de, sc, en, qu, e	800	3473
Pi, er, re	Li, er, re	1	3						
no, us	po, us	134	116						
ac, cu, ei, ll, on, s	ac, us, il, lo, ns								

Figure E.3: Bigrams Page 3 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
so, mm, es	so, nm, es	63	∅						
no, tr, e							ho, tr, e	1653	202
bu, re, au	hu, re, au	4	8						
fo, ye, r	lo, ye, r	28	23				Jo, ve, r	447 199	27 732
mo, de, st, e	mo, je, st, e	442	58						
ha, rm, on, ie	ha, rm, ou, ie	179	79						
Ha, sk, el, l	Ma, st, el, l	4 ∅	20 60						
Mu, rr, ay	Mu, vr, ay	33	21						
Ma, is	ba, is	20	7						
as, pi, ra, ti, on, s	as, hi, ra, ti, on, s	12	9						
gé, né	sé, né	6	21						
hu, ma, in, e	pu, ma, in, e	8	41						
ai, ns, i	si, ns, i	135	56						
ça, is	Ca, is	14	12						
so, um, is	so, nm, is	3	∅						
Se, in, e	se, in, e	1	129						
Mo, nt, ag, ne	Ho, nt, ag, ne	27	6						
Sa, in, te	aa, in, te	3	∅						
Ri, en	hi, en	∅	9						
Qu, oi	co, i						Co, oi	96	223
do, nc	do, nd, t								
sa, ns	an, s								
fa, ve, ur							fu, ve, ur	1034	100
ch, ré, ti, en, ne, s	ch, ré, ti, en, es								
Or, ie, nt	cr, ie, nt	1	26				or, ie, nt	18	463
su	au	69	123						
ls, la, m	fe, la, s	∅	35				ls, la, n/, ls, la, m		
sa, ng	sa, ns	18	173						
pa, r	Be, r	216	9						

Figure E.4: Bigrams Page 4 (Token Error Analysis)



Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
a	s								
So, n	so, n	5	107						
in, ap, er, çu	in, sp, er, du	27 2	11 59				in, ap, er, qu	10	3473
ag, it, at, io, n	as, it, at, io, n	19	29						
ép, hé, mè, re, s	Cp, hé, mè, re, s	11	∅				ép, hé, nè, re, s	46	10
co, mm, is, es	co, mi, se, s								
to, uj, ou, rs	to, uj, on, s						to, uj, or, rs	989	463
co, ns, er, ve	co, us, er, ve	173	200						
Mu, rr, ay	Mu, ra, y								
Sé, na, te, ur	sé, na, to, ur	4 115	21 87						
ES, TO, UR, NE, LL, ES	Rs, fd, UR, LI, Es			ES, TO, UR, NE, LE, S					
CO, NS, TA, NT	Co, Na, IA, N								
Ba, ro, n	Ma, ro, n	2	20						
AD, EL, SW, AR, D	AD, EL, Sy, AR, D	∅	∅						
Re, ct, eu, r	Be, ct, eu, r	1	9				Be, ct, eu, r	35	78
AP, PE, LL	Ap, LN, IT	∅ 1 2	3 ∅ 3						
SH, OT, WE, LL	SH, CT, NP, II	∅ ∅ 2	∅ ∅ ∅	Su, OT, VE, LL	1	8	SH, OT, UW, EL, L		
SA, NG, RO	SA, NC, h0	∅ ∅	∅ ∅	SA, NC, RO	1	2	SA, NC, RO	19	24
OL, AN, O	Cl, A4, 40								
Pr, of, es, se, ur	pr, of, es, se, ur	24	105						
RE, DL, IC, H	BE, TL, IO, N	14 ∅ ∅	1 19 ∅						
He, ll, mu, th	Me, ll, mu, th	1	1				Me, ll, mu, th	20	35
GE, RL, AC, H	GE, RL, AC, N								
Pr, of, es, se, ur	pr, cf, es, se, ur	14	∅						
FO, ER, ST, ER	FO, FR, CT, RR	23 2 23	1 ∅ ∅						
Pr, of, es, se, ur	pr, of, es, se, ur	24	105						
GI, DE	GI, LR	2 5	2 ∅						
EF, RE, MO, FF	NE, NE, MO, TT	∅ 14 ∅	8 8 7				EE, RE, MO, FP	2 5	2 ∅
Dé, pu, té	pé, pu, té	7	10				Dd, ép, ut, é		

Figure E.5: Bigrams Page 5 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
Ju, st, in	Du, st, in	8	2						
GO, DA, RT	do, pA, R								
Sé, na, te, ur	sé, na, te, ur	4	21						
LA	La	4	25						
FO, NT, AI, NE	Fo, NL, AI, RR	2 3 8	3 ø ø	FO, NT, AT, NE	2	3			
Pr, of, es, se, ur	Fr, of, es, se, ur	24	26						
He, nr, i	Be, nr, i	1	9						
LI, CH, TE, NB, ER, GE, R	LI, ON, TE, RE, Rd, ER						LI, CH, TE, NR, ER, GE, R	ø	ø
LE, JE, UN, E	LE, TE, UE, E	ø 2	ø 2	LE, JE, UE			LE, TE, UE		
Re, pr, és, en, ta, nt	Be, pr, és, en, ta, nt	1	9						
Al, be, rt	Al, he, rt	12	16						
TH, OM, AS	La, cn, s						TR, OR, AS	4 8	75 23
NI, PP, OL, D	NI, PP, GL, R	ø	1						
JA, UD, ON	TA, ND, ON	ø ø	2 3						
CO, NV, ER, SE, T	Co, Nv, ER, SE, T	ø	ø						
ES, TO, UR, NE, LL, ES	Es, fo, uR, ET, IE, S								
CO, NS, TA, NT	Co, Rs, TA, NT	2 2	17 ø						
Pr, of, es, se, ur	Fr, of, es, se, ur	24	26						
Th	lh	ø	2						
RU, YS, SE, N	BU, TS, sR, N	1 ø 2	19 ø ø				BU, YS, SE, N	15	171
Pr, of, es, se, ur	Pr, of, es, eu, r								
PR, UD, HO, MM, EA, UX	PR, pR, ON, MU, RA, U			PU, DH, OM, ME, AU, X					
DA, ND, IE, U	BA, NR, IE, S	ø 3	ø ø						
LE, TT, RE				LE, TE					
PA, RI, S				PA, al, S	9	ø			
Ju, in							ru, in	38	142
ON				OE	10	ø			
PA, RL, E				PA, UL, E	1	1			
PL, US	LU, S								

Figure E.6: Bigrams Page 6 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
AF, FA, IR, E				AF, AT, E			AP, PA, IR, E	2 13	11 174
CA, IL, LA, UX	CA, IL, LA, UT	1	∅	CA, IL, TA, U			CA, IL, AU		
CA, VA, LL, IN, I	CA, VA, ll, IN, I	2	∅	SA, vA, U, EL					
VI, EN, T				VI, ET					
ET, RE				ET, SE	12	5			
AC, QU, IT, TÉ				AC, OU, IT, T					
af, fa, ir, e	ar, fa, ir, e	19	45						
Ca, il, la, ux	ca, il, la, ux	12	23						
ap, pa, rt, ie, nt	ap, ba, rt, ie, nt	216	7	ap, pa, rt, ie, nt	197	1039			
So, n				Se, n	12	3	Se, n	91	89
AC, TI, ON				A0, TT, ON	2 8	∅ 8			
FR, AN, CA, IS, E	PR, AN, CA, IS, N	1	1						
pa, rl, e				pa, fl, e	24	13			
ac, cu, sa, te, ur, s				as, gu, sa, te, ur, s	121 35	71 68	an, su, sa, te, ur, s	493 238	2151 960
Ro, me							Bo, me	60	58
Pu, bl, ic							pu, bl, ic	12	500
On				on	20	532			
PA, RI, S				PA, Rr, S	9	∅			
le				Ce	1039	38	ee	5894	6
CO, NT, IN, UE	CO, NT, IN, UR	2	∅	CO, NT, HR, E					
IM, PO, RT, UN	IN, TO, NT, UN	1 3 1	3 ∅ 3	LÉ, TO, RC, UE	∅ ∅ 4 5	2 5 ∅ ∅			
ET	NT	6	3	EX	3	2			
CO, NT, RE	GO, NT, RE	2	∅	CO, NT, E					
NA, TU, RE	NA, TR, E			FA, TU, RE	11	2			
Sa, rt, he	sa, rt, he	3	95						
pa, s	ba, s	216	7						
pr, in, te, mp, s							br, in, te, mp, s	1589	208
hu, ma, in, es	hn, ma, in, es	8	1						
la, is, se, ra, ie, nt	la, is, sé, ra, ie, nt	129	21	la, is, sé, ra, ie, nt	411	77	la, is, ss, ra, ie, nt	2241	840

Figure E.7: Bigrams Page 7 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
ma, is				Ma, is	284	81			
no, n				no, in			nu, n	1653	204
es, t							es, it		
ce, pe, nd, an, t							ce, be, nd, an, t	1100	244
ha, ut, e							hu, au, te		
te, nu, e				te, nq, ue			te, nq, ue		
in, fi, ni				in, fi, ri	117	241	in, fi, ri	593	1185
ça, is, e	Ca, is, e	14	12	ca, is, e	20	134			
gu, er, re	gu, er, rs	302	43						
po, ur, qu, oi	po, ur, qu, ci	43	34				po, ur, cc, oi	3473	∅
ap, er, ço, it	ap, er, co, it	2	212						
Ce, rt, es				Ge, rt, es	38	13	Ce, rt, eo, s		
ho, mm, ag, e				ho, mm, ge					
Mu, rr, ay							Mu, rr, ey	174	5
BU, TL, ER				EU, TL, ER	17	1			
LE, TT, RE				LE, VT, RR	8 12	∅ ∅			
LE				VE	22	1	EN	273	88
DÉ, SA, RM, EM, EN, T	DÉ, SA, RM, EM, RN, T	5	1	FÉ, SA, RM, EE, NT			DÉ, SA, RU, ER, GE, NT		
AL, LE, MA, GN, E				AT, LA, CR					
GE, RL, AC, H	GR, LA, CN			DE, UL, AC, S	3 1	10 1	CE, RL, AG, H	35 20	52 11
FO, ER, ST, ER	FO, ER, ST, R			gO, EM, CI, ER	2 22 2	∅ ∅ 1	PO, ER, ST, ER	20	44
RÉ, PO, NS, E	BÉ, PO, NS, RE			RU, FO, NS, R	1 ∅	1 2	RÉ, PO, NR, E	91	∅
à				A					
VO, TR, E				VO, RE			VO, TR, N		
LE, TT, RE	LE, TE, RE	7	∅	LU, TR, E					
DÉ, CE, MB, RE	DÉ, RS, RE			FÉ, OR, ÉE, RE	1 1 1	∅ 3 ∅	DÉ, GE, MB, RE	52	35
ch, er				cu, er	212	35			
Bu, tl, er							mu, tl, er	176	57
mo, ra, l							ao, ra, l	1046	8

Figure E.8: Bigrams Page 8 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
dé, sa, rm, em, en, t	dé, sr, me, me, nt			dé, sa, ur, me, me, nt					
Al, le, ma, gn, e	Al, le, ma, ge			Al, le, sa, gu, e	284   39	217   68			
no, s				qo, s	278	ø			
pr, éo, cc, up, at, io, ns	pr, éo, cc, up, st, io, ns	70	60	pr, éo, cc, up, at, io, us	391	429	pr, ée, co, up, at, io, ns	8	334
pa, rl, é				pa, rt, é	24	193			
co, mp, at, ri, ot, es				co, mp, Et, ri, ot, es	165	52			
Pr, of, es, se, ur	Fr, of, es, se, ur	24	26						
Ce, nt, ra, l							Ge, nt, ra, l	195	31
Ho, mm, e							ho, mm, e	24	202
Ke, ss, le, r				Fe, ss, le, r	ø	7			
Ke, ss, le, r				Fe, ss, le, r	ø	7	Ne, ss, le, r	6	36
ap, pr, is				ap, ri, s					
re, pr, és, en, ta, nt, s							re, pr, és, on, ta, nt, s	4726	2608
Fr, io, ul	Fr, jo, ul	61	38				Pr, io, ul	357	321
ch, ez	ch, es	13	271						
pr, op, ri, ét, és				pr, ob, ri, ét, és	51	15			
Lo, zè, re				Lo, mè, re	ø	6			
fa, ci, le, me, nt	fa, di, le, me, nt	34	57						
Pa, rl, em, en, t	pa, rl, em, en, t	16	216						
Ch, am, br, un	Gh, am, ru, n								
La, sc, az, es	la, sc, ag, es	25   1	239   19	la, sc, am, es	49   1	580   83			
no, ta, mm, en, t				no, te, mm, en, t	156	364	no, ta, me, nt		
Mo, ns, ie, ur							Mo, ms, ie, ur	2388	2
Ni, ch, ol, as				Ni, ch, ou, as	33	162			
BU, TL, ER				EU, TL, ER	17	1			
le, tt, re, s							la, tt, re, s	5894	3347
Yo, rk							No, rk	21	182
Ce, rt, ai, ne, s							Ge, rt, ai, ne, s	195	31
qu, an, d				Ch, an, d	616	29	dh, an, d	3473	1

Figure E.9: Bigrams Page 9 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
vo, us	Vo, us	101	21						
Ve, rd, un	ve, rd, un	1	63				ve, rd, un	24	732
Ar, de, nn, es	Ar, de, ne, s								
Vo, sg, es	vo, sg, es	21	101				vo, sg, es	1367	193
lu, tt, es							lu, et, es	500	2453
no, bl, e				no, Rk, e	105	∅	no, èr, e	501	173
Dé, fe, ns, e	pé, fe, ns, e	7	10				pé, fe, ns, e	63	194
fr, an, co	Fr, an, co	15	26	gr, an, co	61	62			
ap, ot, hé, os, e	ap, ot, hé, cs, e	11	∅						
LE, TT, RE				AE, TO, E			RE, TT, RE	273	203
Cr, éa, ns				cr, éa, ns	12	64			
Ao, ût				Ac, ût	1	1			
EN	N			NE	2	12			
AL, LE, MA, GN, E	AL, LE, RM, AG, NE			AL, EW, AG, E			AL, LE, MA, SN, E	5	1
en, co, ur, ag, eo, ns	en, co, ur, ag, ec, ns	1	34						
ma, ni, fe, st, at, io, n				ma, ni, ge, st, at, io, n	62	70			
Vo, ic, i	Vo, jo, i	32	38	vo, ic, i	28	246			
po, ur	no, ur	116	134	mo, ur	323	223			
dé, cl, a				dé, Cl, a	31	13			
gu, er, re				qu, er, re	68	616			
là				là	9	∅			
mi, sè, re, s	Bi, sè, re, st								
Es, t	Rs, t	2	∅	Hs, t	14	∅			
di, xi, èm, e				di, xi, èm, e	5	3			
in, vi, to, ns				in, vi, to, né	391	∅			
ci, me, ti, èr, e	ci, me, tj, èr, e	124	∅	ci, me, td, èr, e	19	∅			
ma, ti, n							mn, ti, n	1692	23
ré, pu, di, at, io, n				ré, pa, di, at, io, n	113	479			
po, us, se, nt				po, us, ce, nt	411	365			

Figure E.10: Bigrams Page 10 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
gu, er, re	su, er, re	49	69	Si, er, re	68	5	Su, er, re	594	44
fr, an, ça, is				fr, an, ca, is	20	134			
Or, at, eu, rs				or, at, eu, rs	4	75	or, at, eu, rs	18	463
Mu, rr, ay	MM, ur, ra, y								

Figure E.11: Bigrams Page 11 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
tra, nqu, ill, ité	tra, nqu, <b>ila</b> , ité	5	∅	tra, nqu, <b>iml</b> , ité	39	∅	tra, nqu, <b>ilb</b> , ité	137	∅
Tou, t	<b>Lou</b> , t	3	3						
tan, ces							<b>tta</b> , nce, s		
de	<b>ss</b>						<b>qa</b>		
bas, e	<b>pas</b> , e	2	56						
Cha, mvr, es							<b>cha</b> , m/r, es	53 ∅	336 ∅
bar, ric, ade, s	<b>par</b> , ric, ade, s	∅	103						
bru, squ, eme, nt	bru, squ, <b>ene</b> , nt	13	1						
Dép, uté, s	<b>pép</b> , uté, s	∅	∅						
hen, sib, le	<b>nen</b> , sib, le	∅	1						
rel, lem, ent	rel, <b>lom</b> , ent	9	∅						
méc, ont, ent, eme, nt	méc, ont, ent, <b>ene</b> , nt	130	1						
biz, arr, e	<b>big</b> , arr, e	∅	∅						
exp, loi, tée							exp, <b>lot</b> , tée	80	3
par, tis, ans	<b>par</b> , isa, ns								
Gra, nd	<b>Cra</b> , nd	1	∅						
Poi, nca, ris, te	<b>poi</b> , nca, ris, te	1	7						
Clé, men, cis, te							Clé, men, <b>çis</b> , te	17	∅
bén, éfi, ce	<b>pén</b> , éfi, ce	∅	1						
cir, cul, air, e	cir, <b>oul</b> , air, e	2	1						
acc, usa, nt	acc, usa, <b>st</b>								
cha, cun	<b>Cha</b> , oun	10 1	4 ∅						
con, dam, nat, ion	<b>Con</b> , dam, nat, ion	80	8						
man, ife, ste, men, t	<b>man</b> , ies, tem, ent								
Res, tai, t	<b>Bes</b> , tai, t	∅	∅						
ici	<b>joi</b>	4	4						
bon, heu, r	bon, <b>peu</b> , r	7	24						
suc, cès	<b>suc</b> , ès								
Que	<b>que</b>	2	140						

Figure E.12: Trigrams Page 1 (Token Error Analysis)



Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
tôt	tdt	4	∅						
sou, hai, tai, ent	sou, nai, tai, ent	1	7						
tit, re	tit, fe								
par, ler	Par, ler	103	11						
Sep, tem	sep, ten	∅ 9	1 18						
Oct, obr, e	oct, obr, e	1	∅				oct, obr, e	10	9
Moi	Noi	∅	∅						
doi, s	coi, s	2	∅						
mon, de	non, de	32	10						
rec, onn, aît	rec, onn, att	2	15	rec, onn, att	4	26			
ceu, x	Ceu, x	4	∅						
mêm, es	mên, es	20	∅						
gar, der	Par, de						par, de		
pre, mie, r	pre, nte, n	9	12						
car, riè, re	car, ris, re	∅	14						
fil, s	fil, a								
Bol, o							Rol, o	∅	∅
ces	Ces	21	2						
Mes, sie, urs	ses, sie, urs	∅	18						
fui	rui	∅	7						
Mai, s	sai, s	14	14						
fal, lu	fai, lu	∅	41				fal, l		
Rép, ond, re	hép, ond, re	∅	∅						
inf, âma, nte	inf, êma, nte	∅	∅				inf, ama, nte	∅	1
Un	In								
déb, at	dép, at	3	6						
fût	ôt								
pré, fér, é	pré, rér, é	7	∅						
yeu, x	veu, x	2	7						

Figure E.13: Trigrams Page 2 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
bea, u	nea, u	6	∅						
rôl, e	pôl, e	∅	∅						
PAR, IS	PAR, lô								
Nov, emb, re	Noy, emb, re	1	∅						
MIS, S	MIô, S	∅	∅						
MAR, GAR, ET	Man, den, Nr	∅ ∅	2 4				MAR, CAR, ET	8	26
ALE, XAN, DER	ALE, Tes, pRR	∅ ∅	∅ ∅	ALE, WAN, DER	1	∅			
PAR, IS	PaR, l			PAu, IC	8	∅			
che, r				che, f					
nou, vea, u	nou, veb, u	1	∅						
Alb, ani, e	Ala, uni, e	∅ ∅	∅ 4						
où	ou								
Cro, ix	Cry, ix	∅	∅	cro, ix	3	12			
Rou, ge	mou, ge	∅	1						
amé, ric, ain, e							amé, ric, ain, s		
pen, sez	pam, sez	15	∅						
dù	dù								
jeu, nes, se	jou, nes, se	10	36						
fau, t	fut								
et	e								
Je	de								
rec, omm, and, e	Loc, omm, and, e	10	∅						
iso, lée	jsc, lée	1	∅						
Mis, s	Nis								
Ale, xan, der	Alo, xan, der	∅	2						
des, cen, due	des, cen, que	∅	140				des, cen, que	19	1744
Pie, rre	Lie, rre	∅	∅						
nou, s	pou, s	48	67						
acc, uei, llo, ns	acu, sil, lon, s								

Figure E.14: Trigrams Page 3 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
som, mes	son, mes	8	61						
not, re							hot, re	272	∅
bur, eau	hur, eau	∅	∅						
foy, er	loy, er	∅	∅				Jov, er	11	∅
mod, est, e	moj, est, e	1	∅						
har, mon, ie	har, mou, ie	32	1						
Has, kel, l	Mas, tel, l	1 1	∅ 5						
Mur, ray	Muv, ray	16	∅						
Mai, s	bai, s	∅	∅						
asp, ira, tio, ns	ash, ira, tio, ns	∅	∅						
gén, é	sén, é	3	∅						
hum, ain, e	pum, ain, e	1	∅						
ain, si	sin, si	12	3						
çai, s	Çai, s	1	6						
sou, mis	son, mis	23	61						
Sei, ne	sei, ne	∅	3						
Mon, tag, ne	Hon, tag, ne	26	1						
Sai, nte	aai, nte	∅	∅						
Rie, n	hie, n	∅	5						
Quo, i	coi						Coo, i	17	∅
don, c	don, dt								
san, s	ans								
fav, eur							fuv, eur	26	∅
chr, éti, enn, es	chr, éti, ene, s								
Ori, ent	cri, ent	∅	6				ori, ent	4	65
su	au								
lsl, am	fel, as	∅	∅				lsl, an/, lsl, am		
san, g	san, s								
par	Ber	103	6						

Figure E.15: Trigrams Page 4 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
a	s								
Son	son	2	61						
ina, per, çu	ins, per, du	9	8				ina, per, du		
agi, tat, ion	asi, tat, ion	1	1						
éph, émè, res	Cph, émè, res	∅	∅				éph, éné, res	1	∅
com, mis, es	com, ise, s								
tou, jou, rs	tou, jon, s						tou, jor, rs	353	∅
con, ser, ve	cou, ser, ve	80	13						
Mur, ray	Mur, ay								
Sén, ate, ur	sén, ato, ur	4 ∅	∅ ∅						
EST, OUR, NEL, LES	Rsf, dUR, LIE, s			EST, OUR, NEL, ES					
CON, STA, NT	CoN, alA, N								
Bar, on	Mar, on	∅	1						
ADE, LSW, ARD	ADE, LSy, ARD	∅	∅						
Rec, teu, r	Bec, teu, r	∅	∅				Bec, teu, r	∅	∅
APP, ELL	Apl, NIT	∅ ∅	∅ ∅						
SHO, TWE, LL	SHC, TNP, ll	∅ ∅	∅ ∅	SuO, TVE, LL	1 1	∅ ∅	SHO, TUW, ELL		
SAN, GRO	SAN, Ch0	∅	∅	SAN, CRO	1	∅	SAN, CRO	1	1
OLA, NO	CIA, 440								
Pro, fes, seu, r	pro, fes, seu, r	1	29						
RED, LIC, H	BET, LIO, N	∅ ∅	∅ ∅						
Hel, lmu, th	Mel, lmu, th	∅	∅				Mel, lmu, th	1	4
GER, LAC, H	GER, LAC, N								
Pro, fes, seu, r	prc, fes, seu, r	1	∅						
FOE, RST, ER	FOF, RCT, RR	2 2	∅ ∅						
Pro, fes, seu, r	pro, fes, seu, r	1	29						
GID, E	GIL, R	2	∅						
EFR, EMO, FF	NEN, EMO, TT	∅	∅				EER, EMO, FP	∅	∅
Dép, uté	pép, uté	∅	∅				Ddé, put, é		

Figure E.16: Trigrams Page 5 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
Jus, tin	Dus, tin	∅	∅						
GOD, ART	dop, AR								
Sén, ate, ur	sén, ate, ur	4	∅						
LA	La								
FON, TAI, NE	FoN, LAI, RR	∅ ∅	∅ ∅	FON, TAT, NE	1	∅			
Pro, fes, seu, r	Fro, fes, seu, r	1	∅						
Hen, ri	Ben, ri	1	∅						
LIC, HTE, NBE, RGE, R	LIO, NTE, RER, dER						LIC, HTE, NRE, RGE, R	∅	∅
LEJ, EUN, E	LET, EUE, E	∅ ∅	7 ∅	LEJ, EUE			LET, EUE		
Rep, rés, ent, ant	Bep, rés, ent, ant	∅	∅						
Alb, ert	Alh, ert	∅	∅						
THO, MAS	Lac, ns						TRO, RAS	2 4	1 ∅
NIP, POL, D	NIP, PGL, R	∅	∅						
JAU, DON	TAN, DON	∅	∅						
CON, VER, SET	CoN, vER, SET	2 ∅	∅ ∅						
EST, OUR, NEL, LES	Esf, ouR, ETI, ES								
CON, STA, NT	CoR, sTA, NT	∅ ∅	∅ ∅						
Pro, fes, seu, r	Fro, fes, seu, r	1	∅						
Th	lh								
RUY, SSE, N	BUT, SsR, N	∅ ∅	19 ∅				BUY, SSE, N	5	∅
Pro, fes, seu, r	Pro, fes, eur								
PRU, DHO, MME, AUX	PRp, RON, MUR, AU			PUD, HOM, MEA, UX					
DAN, DIE, U	BAN, RIE, S	∅ ∅	∅ ∅						
LET, TRE				LET, E					
PAR, IS				PAa, IS	8	∅			
Jui, n							rui, n	16	70
ON				OE					
PAR, LE				PAU, LE	8	∅			
PLU, S	LUS								

Figure E.17: Trigrams Page 6 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
AFF, AIR, E				AFA, TE			APP, AIR, E	2	8
CAI, LLA, UX	CAI, LLA, UT			CAI, LTA, U			CAI, LAU		
CAV, ALL, INI	CAV, AI, INI	2	∅	SAV, ALI, EL					
VIE, NT				VIE, T					
ETR, E				ETS, E	∅	∅			
ACQ, UIT, TÉ				ACO, UIT, T					
aff, air, e	arf, air, e	18	∅						
Cai, lla, ux	cai, lla, ux	6	1						
app, art, ien, t	apb, art, ien, t	14	∅	app, art, len, t	46	20			
Son				Sen	2	∅	Sen	8	8
ACT, ION				AOT, TON	∅	∅			
FRA, NCA, ISE	PRA, NCA, ISN	1 1	∅ ∅						
par, le				paf, le	246	∅			
acc, usa, teu, rs				asg, usa, teu, rs	39	∅	ans, usa, teu, rs	138	140
Rom, e							Bom, e	1	∅
Pub, lic							pub, lic	∅	59
On				on					
PAR, IS				PAR, rS					
le				Ce			ee		
CON, TIN, UE	CON, TIN, UR			CON, THR, E					
IMP, ORT, UN	INT, ONT, UN	1 1	∅ ∅	LÉT, ORC, UE	∅ ∅	∅ ∅			
ET	NT			EX					
CON, TRE	GON, TRE	2	∅	CON, TE					
NAT, URE	NAT, RE			FAT, URE	∅	∅			
Sar, the	sar, the	3	∅						
pas	bas	56	2						
pri, nte, mps							bri, nte, mps	246	17
hum, ain, es	hnm, ain, es	1	∅						
lai, sse, rai, ent	lai, ssé, rai, ent	14	3	lai, ssé, rai, ent	32	18	lai, sss, rai, ent	248	∅

Figure E.18: Trigrams Page 7 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
mai, s				Mai, s	96	27			
non				nom	17	37	nun	135	∅
est							esi, t		
cep, end, ant							ceb, end, ant	33	∅
hau, te							hua, ute		
ten, ue				ten, que			ten, que		
inf, ini				inf, iri	9	∅	inf, iri	72	∅
çai, se	Cai, se	1	6	cai, se	1	11			
gue, rre	gue, rrs	38	∅						
pou, rqu, oi	pou, rqu, oi						pou, rçç, oi	27	∅
ape, rço, it	ape, rco, it	1	∅						
Cer, tes				Ger, tes	4	11	Cer, teo, s		
hom, mag, e				hom, mge					
Mur, ray							Mur, rey	128	∅
BUT, LER				EUT, LER	17	∅			
LET, TRE				LEV, TRR	7 7	1 ∅			
LE				VE			EN		
DÉS, ARM, EME, NT	DÉS, ARM, EMR, NT	1	∅	FÉS, ARM, EEN, T			DÉS, ARU, ERG, ENT		
ALL, EMA, GNE				ATL, ACR					
GER, LAC, H	GRL, ACN			DEU, LAC, S	2	∅	CER, LAG, H	3 6	3 1
FOE, RST, ER	FOE, RST, R			gOE, MCI, ER	1 1	∅ ∅	POE, RST, ER	7	∅
RÉP, ONS, E	BÉP, ONS, RE			RUF, ONS, R	∅	∅	RÉP, ONR, E	16	∅
à				A					
VOT, RE				VOR, E			VOT, RN		
LET, TRE	LET, ERE	8	∅	LUT, RE					
DÉC, EMB, RE	DÉR, SRE			FÉO, RÉE, RE	∅ 1	∅ ∅	DÉG, EMB, RE	3	∅
che, r				cue, r	68	∅			
But, ler							mut, ler	129	8
mor, al							aor, al	66	∅

Figure E.19: Trigrams Page 8 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
dés, arm, eme, nt	dés, rme, men, t			dés, aur, mem, ent					
All, ema, gne	All, ema, ge			All, esa, gue	12 7	ø 45			
nos				qos	42	ø			
pré, occ, upa, tio, ns	pré, occ, ups, tio, ns	7	ø	pré, occ, upa, tio, us			pré, eco, upa, tio, ns	34	ø
par, lé				par, té					
com, pat, rio, tes				com, pEt, rio, tes	10	ø			
Pro, fes, seu, r	Fro, fes, seu, r	1	ø						
Cen, tra, l							Gen, tra, l	5	10
Hom, me							hom, me	4	86
Kes, sle, r				Fes, sle, r	ø	ø			
Kes, sle, r				fes, sle, r	ø	ø	Nes, sle, r	ø	1
app, ris				apr, is					
rep, rés, ent, ant, s							rep, rés, ont, ant, s	1748	395
Fri, oul	Frj, oul	1	ø				Pri, oul	10	7
che, z	che, s								
pro, pri, été, s				pro, bri, été, s	48	4			
Loz, ère				Lom, ère	ø	ø			
fac, ile, men, t	fad, ile, men, t	2	ø						
Par, lem, ent	par, lem, ent	11	103						
Cha, mbr, un	Gha, mru, n								
Las, caz, es	las, cag, es	1	ø	las, cam, es	ø	17			
not, amm, ent				not, emm, ent	2	3	not, ame, nt		
Mon, sie, ur							Mom, sie, ur	325	ø
Nic, hol, as				Nic, hou, as	15	5			
BUT, LER				EUT, LER	17	ø			
let, tre, s							lat, tre, s	110	32
Yor, k							Nor, k	9	11
Cer, tai, nes							Ger, tai, nes	9	11
qua, nd				Cha, nd	30	20	dha, nd	168	ø

Figure E.20: Trigrams Page 9 (Token Error Analysis)



Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
vou, s	Vou, s	38	6						
Ver, dun	ver, dun	1	17				ver, dun	14	316
Ard, enn, es	Ard, ene, s								
Vos, ges	vos, ges	1	4				vos, ges	2	53
lut, tes							lue, tes	39	25
nob, le				noR, ke	5	∅	noè, re	21	∅
Déf, ens, e	péf, ens, e	2	∅				péf, ens, e	2	2
fra, nco	Fra, nco	8	25	gra, nco	42	43			
apo, thé, ose	apo, thé, cse	5	∅						
LET, TRE				AET, OE			RET, TRE	93	6
Cré, ans				cré, ans	8	2			
Aoû, t				Acû, t	1	∅			
EN	N			NE					
ALL, EMA, GNE	ALL, ERM, AGN, E			ALE, WAG, E			ALL, EMA, SNE	1	1
enc, our, age, ons	enc, our, age, cns	15	∅						
man, ife, sta, tio, n				man, ige, sta, tio, n	8	7			
Voi, ci	Voj, oi	5	∅	voi, ci	9	58			
pou, r	nou, r	67	48	mou, r	214	1			
déc, la				déC, la	20	∅			
gue, rre				que, rre	45	286			
là				là					
mis, ère, s	Bis, ère, st								
Est	Rst	1	∅	Hst	13	∅			
dix, ièm, e				dix, ièm, e	∅	∅			
inv, ito, ns				inv, ito, nē					
cim, eti, ère	cim, etj, ère	2	∅	cim, etd, ère	1	∅			
mat, in							mnt, in	69	∅
rép, udi, ati, on				rép, adi, ati, on	2	∅			
pou, sse, nt				pou, sce, nt	32	∅			

Figure E.21: Trigrams Page 10 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
gue, rre	sue, rre	36	∅	Sie, rre	45	∅	Sue, rre	513	∅
fra, nça, is				fra, nca, is	17	4			
Ora, teu, rs				ora, teu, rs	∅	10	ora, teu, rs	∅	57
Mur, ray	MMu, rra, y								

Figure E.22: Trigrams Page 11 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
tran, quil, lité	tran, quil, aité	1	∅	tran, quim, lité	4	∅	tran, quil, bité	34	∅
Tout	Lout	3	∅						
tanc, es							ttan, ces		
de	ss						qa		
base	pase	∅	∅						
Cham, vres							cham, /res	18 2	43 ∅
barr, icad, es	parr, icad, es	∅	∅						
brus, quem, ent	brus, quen, ent	∅	∅						
Dépu, tés	pépu, tés	∅	∅						
hens, ible	nens, ible	∅	∅						
rell, emen, t	rell, omen, t	11	∅						
méco, nten, teme, nt	méco, nten, tene, nt	2	∅						
biza, rre	biga, rre	∅	∅						
expl, oité, e							expl, otté, e	1	1
part, isan, s	pari, sans								
Gran, d	Cran, d	∅	∅						
Poin, cari, ste	poin, cari, ste	1	5						
Clém, enci, ste							Clém, ençi, ste	3	∅
béné, fice	péné, fice	∅	∅						
circ, ulai, re	ciro, ulai, re	1	∅						
accu, sant	accu, saut	1	2						
chac, un	Chao, un	∅	∅						
cond, amna, tion	Cond, amna, tion	3	∅						
mani, fest, emen, t	mani, este, ment								
Rest, ait	Best, ait	∅	∅						
ici	joi								
bonh, eur	bonp, eur	∅	∅						
succ, ès	sucè, s								
Que	que								

Figure E.23: Tetragrams Page 1 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
tôt	tdt								
souh, aita, ient	soun, aita, ient	∅	∅						
titr, e	titf, e	1	∅						
parl, er	Parl, er	18	1						
Sept, em	sept, en	∅	1						
Octo, bre	octo, bre	1	∅				octo, bre	10	9
Moi	Noi								
dois	cois	∅	∅						
mond, e	nond, e	9	∅						
reco, nnaî, t	reco, nnat, t	1	∅	reco, nnat, t	∅	∅			
ceux	Ceux	4	∅						
même, s	mêne, s	21	∅						
gard, er	Pard, e						pard, e		
prem, ier	pren, ten	9	4						
carr, ière	carr, isre	5	∅						
fil	fila	∅	∅						
Bolo							Rolo	∅	∅
ces	Ces								
Mess, ieur, s	sess, ieur, s	∅	1						
fui	rui								
Mais	sais	∅	9						
fall, u	fail, u	∅	∅				fall		
Répo, ndre	hépo, ndre	∅	∅						
infâ, mant, e	infê, mant, e	∅	∅				infa, mant, e	3	6
Un	In								
déba, t	dépa, t	2	1						
fût	ôt								
préf, éré	prér, éré	∅	∅						
yeux	veux	2	∅						

Figure E.24: Tetragrams Page 2 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
beau	neau	5	∅						
rôle	pôle	∅	∅						
PARI, S	PARI, ô								
Nove, mbre	Noye, mbre	1	∅						
MISS	MIôS	∅	∅						
MARG, ARET	Mand, enNr	∅ ∅	∅ ∅				MARC, ARET	∅	∅
ALEX, ANDE, R	ALET, espR, R	∅ ∅	∅ ∅	ALEW, ANDE, R	1	∅			
PARI, S	PaRI			PAul, C	7	∅			
cher				chef	24	6			
nouv, eau	nouv, ebu								
Alba, nie	Alau, nie	∅	∅						
où	ou								
Croi, x	Cryi, x	∅	∅	croi, x	3	11			
Roug, e	moug, e	∅	∅						
amér, icai, ne							amér, icai, ns		
pens, ez	pams, ez	8	∅						
dû	dû								
jeun, esse	joun, esse	10	∅						
faut	fut								
et	e								
Je	de								
reco, mman, de	Loco, mman, de	8	∅						
isol, ée	jscl, ée	1	∅						
Miss	Nis								
Alex, ande, r	Alox, ande, r	∅	∅						
desc, endu, e	desc, enqu, e	∅	∅				desc, enqu, e	13	13
Pier, re	Lier, re	∅	∅						
nous	pous	43	2						
accu, eill, ons	acus, illo, ns								

Figure E.25: Tetragrams Page 3 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
somm, es	sonm, es	7	∅						
notr, e							hotr, e	243	∅
bure, au	hure, au	∅	1						
foye, r	loye, r	∅	∅				Jove, r	11	∅
mode, ste	moje, ste	1	∅						
harm, onie	harm, ouie	∅	∅						
Hask, ell	Mast, ell	∅	∅						
Murr, ay	Muvr, ay	16	∅						
Mais	bais	11	∅						
aspi, rati, ons	ashi, rati, ons	∅	∅						
généré	séné	3	∅						
huma, ine	puma, ine	1	∅						
ains, i	sins, i	3	∅						
çais	Cais	13	∅						
soum, is	sonm, is	∅	∅						
Sein, e	sein, e	∅	∅						
Mont, agne	Hont, agne	∅	∅						
Sain, te	aaïn, te	∅	∅						
Rien	hien	∅	∅						
Quoi	coi						Cooi	17	∅
donc	dond, t								
sans	ans								
fave, ur							fuve, ur	14	∅
chré, tien, nes	chré, tien, es								
Orie, nt	crie, nt	∅	∅				orie, nt	5	23
su	au								
Isla, m	fela, s	∅	∅				Isla, n/Is, lam		
sang	sans	∅	18						
par	Ber								

Figure E.26: Tetragrams Page 4 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
a	s								
Son	son								
inap, erçu	insp, erdu	∅ ∅	2 ∅				inap, erqu	∅	1
agit, atio, n	asit, atio, n	1	∅						
éphé, mère, s	Cphé, mère, s	∅	∅				éphé, nère, s	22	∅
comm, ises	comi, ses								
touj, ours	touj, ons						touj, orrs	139	∅
cons, erve	cous, erve	16	∅						
Murr, ay	Mura, y								
Séna, teur	séna, tour	4 1	∅ ∅						
ESTO, URNE, LLES	Rsfd, URLI, Es			ESTO, URNE, LES					
CONS, TANT	CoNa, IAN								
Baro, n	Maro, n	∅	∅						
ADEL, SWAR, D	ADEL, SyAR, D	∅	∅						
Rect, eur	Bect, eur	∅	∅				Bect, eur	∅	∅
APPE, LL	ApLN, IT	∅	∅						
SHOT, WELL	SHCT, NPII	∅ ∅	∅ ∅	SuOT, VELL	1 1	∅ ∅	SHOT, UWEL, L		
SANG, RO	SANC, h0	∅	∅	SANC, RO	1	∅	SANC, RO	2	2
OLAN, O	CIA4, 40								
Prof, esse, ur	prof, esse, ur	1	7						
REDL, ICH	BETL, ION	∅ ∅	∅ ∅						
Hell, muth	Mell, muth	∅	∅				Mell, muth	1	4
GERL, ACH	GERL, ACN								
Prof, esse, ur	prcf, esse, ur	1	∅						
FOER, STER	FOFR, CTRR	2 2	∅ ∅						
Prof, esse, ur	prof, esse, ur	1	7						
GIDE	GILR	2	∅						
EFRE, MOFF	NENE, MOTT	∅ ∅	∅ ∅				EERE, MOFP	∅ ∅	∅ ∅
Dépu, té	pépu, té	∅	∅				Ddép, uté		

Figure E.27: Tetragrams Page 5 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
Just, in	Dust, in	∅	∅						
GODA, RT	dopA, R								
Séna, teur	séna, teur	4	∅						
LA	La								
FONT, AINE	FoNL, AIRR	∅ ∅	∅ ∅	FONT, ATNE	1	∅			
Prof, esse, ur	Frof, esse, ur	1	∅						
Henr, i	Benr, i	1	∅						
LICH, TENB, ERGE, R	LION, TERE, RdER						LICH, TENR, ERGE, R	∅	∅
LEJE, UNE	LETE, UEE	∅ 1	∅ ∅	LEJE, UE			LETE, UE		
Repr, ésen, tant	Bepr, ésen, tant	∅	∅						
Albe, rt	Alhe, rt	∅	∅						
THOM, AS	Lacn, s						TROR, AS	2	∅
NIPP, OLD	NIPP, GLR								
JAUD, ON	TAND, ON	∅	∅						
CONV, ERSE, T	CoNv, ERSE, T	∅	∅						
ESTO, URNE, LLES	Esfo, uRET, IES								
CONS, TANT	CoRs, TANT	∅	∅						
Prof, esse, ur	Frof, esse, ur	1	∅						
Th	lh								
RUYS, SEN	BUTS, sRN	∅	∅				BUYS, SEN	5	∅
Prof, esse, ur	Prof, eseu, r								
PRUD, HOMM, EAUX	PRpR, ONMU, RAU			PUDH, OMME, AUX					
DAND, IEU	BANR, IES	∅	∅						
LETT, RE				LETE					
PARI, S				PAal, S	7	∅			
Juin							ruin	4	38
ON				OE					
PARL, E				PAUL, E	∅	∅			
PLUS	LUS								

Figure E.28: Tetragrams Page 6 (Token Error Analysis)



Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
AFFA, IRE				AFAT, E			APPA, IRE	∅	∅
CAIL, LAUX	CAIL, LAUT	1	∅	CAIL, TAU			CAIL, AU		
CAVA, LLIN, I	CAVA, IIN, I	1	∅	SAVA, LIEL					
VIEN, T				VIET					
ETRE				ETSE	∅	∅			
ACQU, ITTÉ				ACOU, ITT					
affa, ire	arfa, ire	3	∅						
Cail, laux	cail, laux	6	∅						
appa, rtie, nt	apba, rtie, nt	1	∅	appa, rtie, nt	∅	∅			
Son				Sen			Sen		
ACTI, ON				AOTT, ON	∅	∅			
FRAN, CAIS, E	PRAN, CAIS, N	1	∅						
parl, e				pafi, e	17	∅			
accu, sate, urs				asgu, sate, urs	13	∅	ansu, sate, urs	44	∅
Rome							Bome	∅	∅
Publ, ic							publ, ic	∅	59
On				on					
PARI, S				PARr, S	7	∅			
le				Ce			ee		
CONT, INUE	CONT, INUR	1	∅	CONT, HRE					
IMPO, RTUN	INTO, NTUN	1 1	∅ ∅	LÉTO, RCUE	∅ ∅	∅ ∅			
ET	NT			EX					
CONT, RE	GONT, RE	2	∅	CONT, E					
NATU, RE	NATR, E			FATU, RE	∅	∅			
Sart, he	sart, he	3	∅						
pas	bas								
prin, temp, s							brin, temp, s	34	∅
huma, ines	hnma, ines	1	∅						
lais, sera, ient	lais, séra, ient	11	∅	lais, séra, ient	21	∅	lais, ssra, ient	194	∅

Figure E.29: Tetragrams Page 7 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
mais				Mais	69	22			
non				nom			nun		
est							esit		
cepe, ndan, t							cebe, ndan, t	28	∅
haut, e							huau, te		
tenu, e				tenq, ue			tenq, ue		
infi, ni				infi, ri			infi, ri		
çais, e	Cais, e	13	∅	çais, e	18	∅			
guer, re	guer, rs								
pour, quoi	pour, quci	6	∅				pour, ççoi	48	∅
aper, çoit	aper, coit	1	∅						
Cert, es				Gert, es	1	∅	Cert, eos		
homm, age				homm, ge					
Murr, ay							Murr, ey		
BUTL, ER				EUTL, ER	17	∅			
LETT, RE				LEVt, RR	7	∅			
LE				VE			EN		
DÉSA, RMEM, ENT	DÉSA, RMEM, RNT			FÉSA, RMEE, NT			DÉSA, RUER, GENT		
ALLE, MAGN, E				ATLA, CR					
GERL, ACH	GRLA, CN			DEUL, ACS	1	∅	CERL, AGH	∅	∅
FOER, STER	FOER, STR			gOEM, CIER	1 1	∅ ∅	POER, STER	7	∅
RÉPO, NSE	BÉPO, NSRE			RUFO, NSR	∅	∅	RÉPO, NRE		
à				A					
VOTR, E				VORE			VOTR, N		
LETT, RE	LETE, RE	6	∅	LUTR, E					
DÉCE, MBRE	DÉRS, RE			FÉOR, ÉERE	∅ ∅	∅ ∅	DÉGE, MBRE	1	∅
cher				cuer	24	∅			
Butl, er							mutl, er	128	∅
mora, l							aora, l	27	∅

Figure E.30: Tetragrams Page 8 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
désa, rmem, ent	désr, meme, nt			désa, urme, ment					
Alle, magn, e	Alle, mage			Alle, sagu, e	6	∅			
nos				qos					
préo, ccup, atio, ns	préo, ccup, stio, ns	15	∅	préo, ccup, atio, us			prée, coup, atio, ns	7 5	∅ 97
parl, é				part, é	17	41			
comp, atri, otes				comp, Etri, otes	∅	∅			
Prof, esse, ur	Prof, esse, ur	1	∅						
Cent, ral							Gent, ral	5	1
Hommm, e							hommm, e	7	86
Kess, ler				Fess, ler	∅	∅			
Kess, ler				Fess, ler	∅	∅	Ness, ler	∅	∅
appr, is				apri, s					
repr, ésen, tant, s							repr, éson, tant, s	44	∅
Frio, ul	Frjo, ul	1	∅				Prio, ul	∅	∅
chez	ches	1	2						
prop, riét, és				prob, riét, és	18	2			
Lozè, re				Lomè, re	∅	∅			
faci, leme, nt	fadi, leme, nt	2	∅						
Parl, emen, t	parl, emen, t	1	18						
Cham, brun	Gham, run								
Lasc, azes	lasc, ages	1 1	∅ ∅	lasc, ames	∅ ∅	∅ ∅			
nota, mmen, t				note, mmen, t	2	2	nota, ment		
Mons, leur							Moms, leur	163	∅
Nich, olas				Nich, ouas	13	∅			
BUTL, ER				EUTL, ER	17	∅			
lett, res							latt, res	73	∅
York							Nork	20	∅
Cert, aine, s							Gert, aine, s	6	∅
quan, d				Chan, d	17	3	dhan, d	99	∅

Figure E.31: Tetragrams Page 9 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
vous	Vous	33	6						
Verd, un	verd, un	1	∅				verd, un	9	2
Arde, nnes	Arde, nes								
Vosg, es	vosg, es	1	∅				vosg, es	2	∅
lutt, es							luet, es	34	∅
nobl, e				noRk, e	5	∅	noèr, e	19	∅
Défe, nse	péfe, nse	1	∅				péfe, nse	2	∅
fran, co	Fran, co	7	25	gran, co	33	36			
apot, hécs, e	apot, hécs, e	1	∅						
LETT, RE				AETO, E			RETT, RE	94	∅
Créa, ns				créa, ns	8	∅			
Août				Acût	1	∅			
EN	N			NE					
ALLE, MAGN, E	ALLE, RMAG, NE			ALEW, AGE			ALLE, MASN, E	1	∅
enco, urag, eons	enco, urag, ecns	1	∅						
mani, fest, atio, n				mani, gest, atio, n	8	∅			
Voic, i	Vojo, i	2	∅	voic, i	3	8			
pour	nour	59	∅	mour	194	∅			
décl, a				déCl, a	3	∅			
guer, re				quer, re	38	5			
là				là					
misè, res	Bisè, rest								
Est	Rst			Hst					
dixi, ème				dixi, ème					
invi, tons				invi, tonè	∅	∅			
cime, tièr, e	cime, tjèr, e	2	∅	cime, tdèr, e	∅	∅			
mati, n							mnti, n	35	∅
répu, diat, ion				répa, diat, ion	16	6			
pous, sent				pous, cent	9	9			

Figure E.32: Tetragrams Page 10 (Token Error Analysis)

Correct transcription	Model War	Nb occ CT	Nb occ MW	Model Other	Nb occ CT	Nb occ MO	Model GT	Nb occ CT	Nb occ MGT
guer, re	suer, re	36	∅	Sier, re	38	∅	Suer, re	469	∅
fran, çais				fran, cais	18	∅			
Orat, eurs				orat, eurs	∅	∅	orat, eurs	∅	3
Murr, ay	MMur, ray								

Figure E.33: Tetragrams Page 11 (Token Error Analysis)

# ALPHABET ANALYSIS (MULTILINGUAL MODEL)

---

## F.1 Distribution of the Alphabet By Characters

*This heatmap displays, from white to bright red cells, the distribution, in percentages, of the characters across the various languages of the multilingual dataset.*

## F.2 Distribution of the Alphabet By Language

*This heatmap displays, from white to bright red cells, the distribution, in percentages, of the alphabet in the multilingual dataset, language by language.*

## F.3 General Distribution of the Alphabet

*This heatmap displays, from white to bright red cells, the general distribution, in numbers, of the alphabet in the multilingual dataset.*

Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
e	35,23%	22,07%	12,60%	12,01%	12,99%	2,85%	2,24%	51838
a	22,90%	21,98%	18,84%	16,25%	9,02%	6,70%	4,32%	29256
t	22,29%	27,67%	14,62%	21,46%	9,12%	2,69%	2,15%	27908
i	33,59%	22,89%	14,46%	8,66%	9,52%	7,73%	3,15%	25096
r	32,45%	22,91%	13,42%	9,72%	14,48%	3,92%	3,10%	24196
s	30,96%	23,25%	16,77%	12,46%	10,17%	3,43%	2,96%	22779
o	12,03%	29,14%	27,82%	11,16%	7,36%	6,67%	5,82%	22304
l	20,44%	18,00%	22,89%	20,27%	12,11%	2,98%	3,32%	17026
n	0,95%	36,06%	31,96%	0,36%	17,95%	6,66%	6,05%	16173
N	72,64%	0,71%	0,77%	24,95%	0,48%	0,34%	0,11%	16168
d	4,36%	26,03%	26,50%	9,05%	23,15%	6,29%	4,63%	12006
m	23,45%	18,70%	20,77%	17,70%	9,70%	5,88%	3,79%	11474
u	41,22%	18,14%	19,84%	9,24%	4,25%	4,29%	3,02%	11449
k	11,81%	4,36%	26,86%	36,23%	9,74%	6,24%	4,77%	10241
c	37,85%	24,53%	20,48%	4,34%	0,90%	8,77%	3,13%	9564
h	58,83%	0,96%	19,29%	9,65%	5,32%	2,57%	3,39%	9120
g	35,28%	15,98%	2,86%	24,39%	17,67%	3,19%	0,63%	8525
v	10,58%	11,40%	42,37%	17,43%	10,28%	0,00%	7,94%	7167
b	27,93%	16,86%	21,47%	18,84%	6,36%	5,09%	3,45%	7111
z	18,06%	2,17%	24,09%	33,14%	0,05%	19,31%	3,19%	6588
p	7,40%	27,11%	36,81%	9,49%	4,70%	8,09%	6,41%	6257
y	0,87%	21,71%	32,55%	21,71%	3,68%	16,43%	3,07%	5546
D	91,50%	1,72%	2,21%	0,87%	2,80%	0,44%	0,46%	5422
f	26,45%	40,53%	3,40%	8,08%	20,20%	0,68%	0,68%	5184
H	4,94%	89,42%	2,19%	0,84%	2,01%	0,36%	0,24%	5020
w	40,46%	40,03%	0,58%	0,68%	0,08%	18,13%	0,04%	4849
á	0,10%	0,00%	47,46%	46,97%	0,00%	0,00%	5,46%	3843
é	0,30%	0,11%	29,58%	64,51%	0,00%	0,00%	5,50%	2637
í	0,12%	0,00%	83,58%	7,84%	0,00%	0,00%	8,46%	2588

Figure F.1: Distribution of the alphabet by character Page 1 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
J	6,51%	17,94%	63,05%	1,52%	9,19%	1,29%	0,49%	2241
j	13,30%	3,15%	4,29%	28,18%	17,09%	22,76%	11,22%	1586
ě	0,07%	0,00%	99,53%	0,00%	0,00%	0,00%	0,40%	1489
S	37,47%	17,41%	18,45%	6,41%	12,74%	4,67%	2,86%	1436
A	39,06%	19,82%	7,18%	16,60%	14,37%	1,32%	1,65%	1211
ü	43,61%	0,00%	3,46%	51,24%	0,09%	0,62%	0,98%	1126
ž	0,18%	0,00%	87,20%	0,00%	0,00%	0,00%	12,62%	1125
B	34,48%	14,85%	17,72%	8,52%	17,24%	2,49%	4,69%	1044
ř	0,10%	0,00%	99,60%	0,00%	0,00%	0,00%	0,30%	1009
T	31,03%	34,28%	18,82%	2,24%	10,07%	2,95%	0,61%	983
I	29,97%	24,76%	17,43%	7,97%	15,73%	2,87%	1,28%	941
M	41,13%	16,97%	12,84%	8,38%	15,02%	3,48%	2,18%	919
ö	22,97%	0,00%	1,41%	73,97%	1,18%	0,00%	0,47%	849
č	0,25%	0,00%	73,90%	0,00%	0,00%	0,00%	25,86%	816
š	0,13%	0,00%	87,52%	0,00%	0,00%	0,00%	12,36%	777
ý	0,00%	0,00%	79,95%	0,00%	0,00%	0,00%	20,05%	763
P	22,15%	11,97%	32,32%	3,58%	15,54%	9,22%	5,23%	727
ó	0,00%	0,00%	0,00%	65,88%	0,00%	34,12%	0,00%	680
G	51,99%	20,32%	4,12%	6,33%	13,11%	2,06%	2,06%	679
F	36,03%	18,78%	4,43%	4,27%	34,81%	1,07%	0,61%	655
K	44,67%	3,50%	17,81%	12,40%	12,72%	4,13%	4,77%	629
E	38,89%	18,30%	8,50%	19,44%	13,56%	0,49%	0,82%	612
R	24,38%	25,54%	20,40%	3,65%	18,24%	1,82%	5,97%	603
L	49,46%	15,40%	10,14%	6,16%	15,40%	2,36%	1,09%	552
V	35,33%	3,99%	38,04%	7,61%	11,05%	0,00%	3,99%	552
ø	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	536
W	63,21%	19,43%	0,94%	3,21%	1,13%	10,57%	1,51%	530
ł	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	525
ő	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	485
ä	87,47%	0,23%	5,69%	1,82%	1,59%	0,00%	3,19%	439

Figure F.2: Distribution of the alphabet by character Page 2 (Multilingual model)



Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
O	16,35%	19,91%	19,67%	9,95%	21,56%	9,00%	3,55%	422
ú	0,00%	0,00%	25,81%	37,97%	0,00%	0,00%	36,23%	403
æ	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	397
ů	0,29%	0,00%	99,13%	0,00%	0,00%	0,00%	0,58%	344
C	10,98%	61,59%	12,50%	7,01%	2,44%	4,57%	0,91%	328
Z	50,48%	2,86%	27,94%	5,40%	0,32%	9,21%	3,81%	315
x	11,74%	51,34%	29,19%	5,70%	0,34%	1,01%	0,67%	298
ę	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	262
U	41,20%	12,80%	6,00%	3,20%	29,20%	6,40%	1,20%	250
ś	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	188
ą	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	163
ż	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	138
ň	0,00%	0,00%	62,11%	0,00%	0,00%	0,00%	37,89%	95
č	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	78
q	9,21%	77,63%	11,84%	0,00%	0,00%	1,32%	0,00%	76
ť	0,00%	0,00%	36,11%	0,00%	0,00%	0,00%	63,89%	72
ď	0,00%	0,00%	42,11%	0,00%	0,00%	0,00%	57,89%	57
X	43,64%	32,73%	21,82%	0,00%	1,82%	0,00%	0,00%	55
ı	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	53
ű	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	47
Č	0,00%	0,00%	75,00%	0,00%	0,00%	0,00%	25,00%	44
Ž	0,00%	0,00%	92,50%	0,00%	0,00%	0,00%	7,50%	40
ň	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	37
ž	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	37
Y	26,67%	30,00%	6,67%	33,33%	3,33%	0,00%	0,00%	30
Æ	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	23
É	0,00%	0,00%	0,00%	95,24%	0,00%	0,00%	4,76%	21
Š	0,00%	0,00%	65,00%	0,00%	0,00%	0,00%	35,00%	20
Ž	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	20
Ö	5,88%	0,00%	0,00%	94,12%	0,00%	0,00%	0,00%	17

Figure F.3: Distribution of the alphabet by character Page 3 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
Q	64,29%	28,57%	0,00%	7,14%	0,00%	0,00%	0,00%	14
Á	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	11
ô	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	11
Ú	0,00%	0,00%	36,36%	54,55%	0,00%	0,00%	9,09%	11
Û	81,82%	0,00%	0,00%	18,18%	0,00%	0,00%	0,00%	11
Ŕ	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	8
Í	0,00%	0,00%	0,00%	57,14%	0,00%	0,00%	42,86%	7
Ä	100,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	6
õ	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	6
Ś	0,00%	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	5
Ø	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	4
Ř	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	0,00%	4
ê	100,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	2
à	0,00%	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	1
â	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	0,00%	1
ë	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	1
Ñ	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	0,00%	1
Ó	0,00%	0,00%	0,00%	100,00%	0,00%	0,00%	0,00%	1
ß	100,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1

Figure F.4: Distribution of the alphabet by character Page 4 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak
A	0,4179%	0,2784%	0,1061%	0,3326%	0,4279%	0,0745%	0,1392%
a	5,9191%	7,4590%	6,7236%	7,8659%	6,4872%	9,1315%	8,7928%
Á	0,0000%	0,0000%	0,0000%	0,0182%	0,0000%	0,0000%	0,0000%
á	0,0035%	0,0000%	2,2249%	2,9872%	0,0000%	0,0000%	1,4620%
À	0,0000%	0,0000%	0,0000%	0,0000%	0,0025%	0,0000%	0,0000%
à	0,0000%	0,0000%	0,0012%	0,0000%	0,0000%	0,0000%	0,0000%
Ä	0,0053%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%
ä	0,3393%	0,0012%	0,0305%	0,0132%	0,0172%	0,0000%	0,0975%
ą	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,7590%	0,0000%
Æ	0,0000%	0,0000%	0,0000%	0,0000%	0,0566%	0,0000%	0,0000%
æ	0,0000%	0,0000%	0,0000%	0,0000%	0,9763%	0,0000%	0,0000%
B	0,3181%	0,1798%	0,2257%	0,1473%	0,4426%	0,1211%	0,3411%
b	1,7548%	1,3909%	1,8626%	2,2176%	1,1115%	1,6857%	1,7057%
C	0,0318%	0,2343%	0,0500%	0,0381%	0,0197%	0,0698%	0,0209%
c	3,1986%	2,7214%	2,3896%	0,6868%	0,2115%	3,9069%	2,0816%
č	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,3632%	0,0000%
Č	0,0000%	0,0000%	0,0403%	0,0000%	0,0000%	0,0000%	0,0766%
ĉ	0,0018%	0,0000%	0,7355%	0,0000%	0,0000%	0,0000%	1,4690%
D	4,3834%	0,1079%	0,1464%	0,0778%	0,3738%	0,1118%	0,1740%
d	0,4630%	3,6251%	3,8802%	1,7973%	6,8339%	3,5157%	3,8708%
d'	0,0000%	0,0000%	0,0293%	0,0000%	0,0000%	0,0000%	0,2297%
E	0,2103%	0,1299%	0,0634%	0,1969%	0,2041%	0,0140%	0,0348%
e	16,1377%	13,2742%	7,9690%	10,3037%	16,5622%	6,8685%	8,0897%
É	0,0000%	0,0000%	0,0000%	0,0331%	0,0000%	0,0000%	0,0070%
é	0,0071%	0,0035%	0,9515%	2,8151%	0,0000%	0,0000%	1,0095%
ê	0,0018%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%
ë	0,0009%	0,0000%	1,8078%	0,0000%	0,0000%	0,0000%	0,0418%
ë	0,0000%	0,0000%	0,0000%	0,0017%	0,0000%	0,0000%	0,0000%
ę	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	1,2200%	0,0000%

Figure F.5: Distribution of the alphabet by language Page 1 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak
F	0,2085%	0,1427%	0,0354%	0,0463%	0,5607%	0,0326%	0,0278%
f	1,2114%	2,4372%	0,2147%	0,6934%	2,5747%	0,1630%	0,2437%
G	0,3119%	0,1601%	0,0342%	0,0712%	0,2189%	0,0652%	0,0975%
g	2,6578%	1,5800%	0,2976%	3,4406%	3,7034%	1,2666%	0,3759%
H	0,2191%	5,2074%	0,1342%	0,0695%	0,2484%	0,0838%	0,0835%
h	4,7404%	0,1021%	2,1456%	1,4564%	1,1927%	1,0896%	2,1512%
I	0,2492%	0,2703%	0,2000%	0,1241%	0,3639%	0,1257%	0,0835%
i	7,4486%	6,6632%	4,4279%	3,5978%	5,8724%	9,0291%	5,5068%
í	0,0000%	0,0000%	0,0000%	0,0066%	0,0000%	0,0000%	0,0209%
í	0,0027%	0,0000%	2,6384%	0,3360%	0,0000%	0,0000%	1,5246%
J	0,1290%	0,4663%	1,7236%	0,0563%	0,5066%	0,1350%	0,0766%
j	0,1864%	0,0580%	0,0829%	0,7398%	0,6664%	1,6810%	1,2392%
K	0,2483%	0,0255%	0,1366%	0,1291%	0,1967%	0,1211%	0,2089%
k	1,0682%	0,5185%	3,3557%	6,1398%	2,4517%	2,9756%	3,3974%
L	0,2412%	0,0986%	0,0683%	0,0563%	0,2090%	0,0605%	0,0418%
l	3,0749%	3,5543%	4,7536%	5,7129%	5,0682%	2,3609%	3,9334%
ĺ	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,3690%
ł	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	2,4447%	0,0000%
M	0,3340%	0,1810%	0,1439%	0,1274%	0,3394%	0,1490%	0,1392%
m	2,3777%	2,4894%	2,9068%	3,3612%	2,7370%	3,1432%	3,0284%
N	10,3776%	0,1334%	0,1513%	6,6760%	0,1894%	0,2561%	0,1253%
n	0,1361%	6,7653%	6,3052%	0,0976%	7,1388%	5,0151%	6,8157%
ň	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,1723%	0,0000%
Ñ	0,0000%	0,0000%	0,0012%	0,0000%	0,0000%	0,0000%	0,0000%
ň	0,0000%	0,0000%	0,0720%	0,0000%	0,0000%	0,0000%	0,2506%

Figure F.6: Distribution of the alphabet by language Page 2 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak
O	0,0610%	0,0974%	0,1012%	0,0695%	0,2238%	0,1769%	0,1044%
o	2,3706%	7,5390%	7,5701%	4,1192%	4,0379%	6,9290%	9,0295%
Ó	0,0000%	0,0000%	0,0000%	0,0017%	0,0000%	0,0000%	0,0000%
ó	0,0000%	0,0000%	0,0000%	0,7414%	0,0000%	1,0803%	0,0000%
ô	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0766%
Ö	0,0009%	0,0000%	0,0000%	0,0265%	0,0000%	0,0000%	0,0000%
ö	0,1723%	0,0000%	0,0146%	1,0393%	0,0246%	0,0000%	0,0278%
Ő	0,0000%	0,0000%	0,0000%	0,0132%	0,0000%	0,0000%	0,0000%
ő	0,0000%	0,0000%	0,0000%	0,8026%	0,0000%	0,0000%	0,0000%
Ø	0,0000%	0,0000%	0,0000%	0,0000%	0,0098%	0,0000%	0,0000%
ø	0,0000%	0,0000%	0,0000%	0,0000%	1,3181%	0,0000%	0,0000%
õ	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0418%
P	0,1423%	0,1009%	0,2867%	0,0430%	0,2779%	0,3120%	0,2646%
p	0,4091%	1,9674%	2,8092%	0,9830%	0,7230%	2,3562%	2,7917%
Q	0,0080%	0,0046%	0,0000%	0,0017%	0,0000%	0,0000%	0,0000%
q	0,0062%	0,0684%	0,0110%	0,0000%	0,0000%	0,0047%	0,0000%
R	0,1299%	0,1786%	0,1500%	0,0364%	0,2705%	0,0512%	0,2506%
r	6,9370%	6,4312%	3,9607%	3,8941%	8,6143%	4,4191%	5,2144%
Ř	0,0000%	0,0000%	0,0049%	0,0000%	0,0000%	0,0000%	0,0000%
ř	0,0009%	0,0000%	1,2259%	0,0000%	0,0000%	0,0000%	0,0209%
S	0,4754%	0,2900%	0,3232%	0,1523%	0,4500%	0,3120%	0,2854%
s	6,2310%	6,1435%	4,6585%	4,6984%	5,6953%	3,6414%	4,6992%
Ś	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0233%	0,0000%
ś	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,8754%	0,0000%
Š	0,0000%	0,0000%	0,0159%	0,0000%	0,0000%	0,0000%	0,0487%
š	0,0009%	0,0000%	0,8295%	0,0000%	0,0000%	0,0000%	0,6683%
ß	0,0009%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%

Figure F.7: Distribution of the alphabet by language Page 3 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak
T	0,2695%	0,3909%	0,2257%	0,0364%	0,2435%	0,1350%	0,0418%
t	5,4967%	8,9566%	4,9756%	9,9098%	6,2609%	3,5017%	4,1841%
ť	0,0000%	0,0000%	0,0317%	0,0000%	0,0000%	0,0000%	0,3202%
U	0,0910%	0,0371%	0,0183%	0,0132%	0,1795%	0,0745%	0,0209%
u	4,1696%	2,4094%	2,7702%	1,7509%	1,1976%	2,2864%	2,4088%
Ú	0,0000%	0,0000%	0,0049%	0,0099%	0,0000%	0,0000%	0,0070%
ú	0,0000%	0,0000%	0,1269%	0,2532%	0,0000%	0,0000%	1,0164%
ů	0,0009%	0,0000%	0,4160%	0,0000%	0,0000%	0,0000%	0,0139%
Ů	0,0080%	0,0000%	0,0000%	0,0033%	0,0000%	0,0000%	0,0000%
ů	0,4338%	0,0000%	0,0476%	0,9549%	0,0025%	0,0326%	0,0766%
ű	0,0000%	0,0000%	0,0000%	0,0778%	0,0000%	0,0000%	0,0000%
V	0,1723%	0,0255%	0,2562%	0,0695%	0,1500%	0,0000%	0,1532%
v	0,6698%	0,9477%	3,7046%	2,0670%	1,8124%	0,0000%	3,9613%
W	0,2960%	0,1195%	0,0061%	0,0281%	0,0148%	0,2608%	0,0557%
w	1,7336%	2,2516%	0,0342%	0,0546%	0,0098%	4,0931%	0,0139%
X	0,0212%	0,0209%	0,0146%	0,0000%	0,0025%	0,0000%	0,0000%
x	0,0309%	0,1775%	0,1061%	0,0281%	0,0025%	0,0140%	0,0139%
Y	0,0071%	0,0104%	0,0024%	0,0165%	0,0025%	0,0000%	0,0000%
y	0,0424%	1,3967%	2,2018%	1,9926%	0,5017%	4,2421%	1,1835%
ý	0,0000%	0,0000%	0,7441%	0,0000%	0,0000%	0,0000%	1,0652%
Z	0,1405%	0,0104%	0,1073%	0,0281%	0,0025%	0,1350%	0,0835%
z	1,0515%	0,1659%	1,9358%	3,6127%	0,0074%	5,9232%	1,4620%
ž	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,1723%	0,0000%
Ž	0,0000%	0,0000%	0,0451%	0,0000%	0,0000%	0,0000%	0,0209%
ž	0,0018%	0,0000%	1,1966%	0,0000%	0,0000%	0,0000%	0,9886%
Ž	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,0931%	0,0000%
ž	0,0000%	0,0000%	0,0000%	0,0000%	0,0000%	0,6426%	0,0000%
Total	113176	86205	81980	60425	40665	21475	14364

Figure F.8: Distribution of the alphabet by language Page 4 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
e	18264	11443	6533	6226	6735	1475	1162	51838
a	6699	6430	5512	4753	2638	1961	1263	29256
t	6221	7721	4079	5988	2546	752	601	27908
i	8430	5744	3630	2174	2388	1939	791	25096
r	7851	5544	3247	2353	3503	949	749	24196
s	7052	5296	3819	2839	2316	782	675	22779
o	2683	6499	6206	2489	1642	1488	1297	22304
l	3480	3064	3897	3452	2061	507	565	17026
n	154	5832	5169	59	2903	1077	979	16173
N	11745	115	124	4034	77	55	18	16168
d	524	3125	3181	1086	2779	755	556	12006
m	2691	2146	2383	2031	1113	675	435	11474
u	4719	2077	2271	1058	487	491	346	11449
k	1209	447	2751	3710	997	639	488	10241
c	3620	2346	1959	415	86	839	299	9564
h	5365	88	1759	880	485	234	309	9120
g	3008	1362	244	2079	1506	272	54	8525
v	758	817	3037	1249	737	0	569	7167
b	1986	1199	1527	1340	452	362	245	7111
z	1190	143	1587	2183	3	1272	210	6588
p	463	1696	2303	594	294	506	401	6257
y	48	1204	1805	1204	204	911	170	5546
D	4961	93	120	47	152	24	25	5422
f	1371	2101	176	419	1047	35	35	5184
H	248	4489	110	42	101	18	12	5020
w	1962	1941	28	33	4	879	2	4849
á	4	0	1824	1805	0	0	210	3843
é	8	3	780	1701	0	0	145	2637
í	3	0	2163	203	0	0	219	2588
J	146	402	1413	34	206	29	11	2241
j	211	50	68	447	271	361	178	1586
ě	1	0	1482	0	0	0	6	1489
S	538	250	265	92	183	67	41	1436
A	473	240	87	201	174	16	20	1211
ü	491	0	39	577	1	7	11	1126
ž	2	0	981	0	0	0	142	1125
B	360	155	185	89	180	26	49	1044
ř	1	0	1005	0	0	0	3	1009
T	305	337	185	22	99	29	6	983
l	282	233	164	75	148	27	12	941
M	378	156	118	77	138	32	20	919
ö	195	0	12	628	10	0	4	849
č	2	0	603	0	0	0	211	816
š	1	0	680	0	0	0	96	777
ý	0	0	610	0	0	0	153	763

Figure F.9: General distribution of the alphabet Page 1 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
P	161	87	235	26	113	67	38	727
ó	0	0	0	448	0	232	0	680
G	353	138	28	43	89	14	14	679
F	236	123	29	28	228	7	4	655
K	281	22	112	78	80	26	30	629
E	238	112	52	119	83	3	5	612
R	147	154	123	22	110	11	36	603
L	273	85	56	34	85	13	6	552
V	195	22	210	42	61	0	22	552
ø	0	0	0	0	536	0	0	536
W	335	103	5	17	6	56	8	530
ł	0	0	0	0	0	525	0	525
ő	0	0	0	485	0	0	0	485
ä	384	1	25	8	7	0	14	439
O	69	84	83	42	91	38	15	422
ú	0	0	104	153	0	0	146	403
æ	0	0	0	0	397	0	0	397
ů	1	0	341	0	0	0	2	344
C	36	202	41	23	8	15	3	328
Z	159	9	88	17	1	29	12	315
x	35	153	87	17	1	3	2	298
ę	0	0	0	0	0	262	0	262
U	103	32	15	8	73	16	3	250
ś	0	0	0	0	0	188	0	188
ą	0	0	0	0	0	163	0	163
ż	0	0	0	0	0	138	0	138
ň	0	0	59	0	0	0	36	95
ć	0	0	0	0	0	78	0	78
q	7	59	9	0	0	1	0	76
ť	0	0	26	0	0	0	46	72
ď	0	0	24	0	0	0	33	57
X	24	18	12	0	1	0	0	55
ı	0	0	0	0	0	0	53	53
ű	0	0	0	47	0	0	0	47
Č	0	0	33	0	0	0	11	44
Ž	0	0	37	0	0	0	3	40
ň	0	0	0	0	0	37	0	37
ž	0	0	0	0	0	37	0	37
Y	8	9	2	10	1	0	0	30
Æ	0	0	0	0	23	0	0	23
É	0	0	0	20	0	0	1	21
Š	0	0	13	0	0	0	7	20
Ž	0	0	0	0	0	20	0	20
Ö	1	0	0	16	0	0	0	17
Q	9	4	0	1	0	0	0	14

Figure F.10: General distribution of the alphabet Page 2 (Multilingual model)

Alphabet	german	english	czech	hungarian	danish	polish	slovak	Total
Á	0	0	0	11	0	0	0	11
ô	0	0	0	0	0	0	11	11
Ú	0	0	4	6	0	0	1	11
Û	9	0	0	2	0	0	0	11
Ů	0	0	0	8	0	0	0	8
Í	0	0	0	4	0	0	3	7
Ä	6	0	0	0	0	0	0	6
ō	0	0	0	0	0	0	6	6
Ś	0	0	0	0	0	5	0	5
Ø	0	0	0	0	4	0	0	4
Ř	0	0	4	0	0	0	0	4
ê	2	0	0	0	0	0	0	2
à	0	0	0	0	1	0	0	1
å	0	0	1	0	0	0	0	1
ë	0	0	0	1	0	0	0	1
Ň	0	0	1	0	0	0	0	1
Ó	0	0	0	1	0	0	0	1
ß	1	0	0	0	0	0	0	1
Total	113176	86205	81980	60425	40665	21475	14364	418290

Figure F.11: General distribution of the alphabet Page 3 (Multilingual model)

## METRICS (MULTILINGUAL MODEL)

---

*Those tables display the metrics obtained after the application of the model EHRI on the test set of the EHRI dataset. It is available in two versions: one version where the comparison was done while taking everything in reference and prediction into account, and one where the comparison was done while ignoring the punctuation.*

Table G.1: Comparative results for the EHRI NFC model (normal)

	EN 1	EN 2	DA 1	DA 2	SK 1	SK 2	FR 1	FR 2	IT 1	IT 2
Levenshtein Distance (Char.)	48	10	106	14	11	11	67	94	98	21
Levenshtein Distance (Words)	33	9	73	19	14	13	62	94	75	21
WER in %	9.85	2.325	39.673	8.636	10.37	6.046	13.596	20.042	20.215	11.351
CER in %	2.51	0.421	8.811	0.893	1.199	0.812	2.433	3.196	4.224	1.76
Wacc in %	90.149	97.674	60.326	91.363	86.629	93.953	86.403	79.957	79.784	88.648
Hits	1867	2363	1107	1557	906	1343	2689	2850	2226	1173
Substitutions	44	7	86	8	7	7	57	83	77	19
Deletions	1	2	10	2	4	4	7	8	17	1
Insertions	3	1	10	4	0	0	3	3	4	1
Total char. in reference	1912	2372	1203	1567	917	1354	2753	2941	2320	1193
Total char. in prediction	1914	2371	1203	1569	913	1350	2749	2936	2307	1193



Table G.3: Comparative results for the EHRI NFC model (no punctuation)

	EN 1	EN 2	DA 1	DA 2	SK 1	SK 2	FR 1	FR 2	IT 1	IT 2
Levenshtein Distance (Char.)	39	9	106	11	10	10	62	90	90	18
Levenshtein Distance (Words)	24	9	73	16	11	13	58	90	68	19
WER in %	7.185	2.362	39.673	7.339	8.208	6.103	12.803	19.313	18.428	10.439
CER in %	2.108	0.389	9.298	0.726	1.129	0.759	2.307	3.132	3.956	1.534
Wacc in %	92.814	97.637	60.326	92.66	91.791	93.896	87.196	80.686	81.571	89.56
Hits	1814	2308	1946	1509	875	1307	2629	2785	2190	1160
Substitutions	31	3	84	5	6	7	46	61	69	12
Deletions	5	2	10	1	4	3	12	27	16	1
Insertions	3	4	12	5	0	0	4	2	5	5
Total char. in reference	1850	2313	1140	1515	885	1317	2687	2873	2275	1173
Total char. in prediction	1848	2315	1142	1519	881	1314	2679	2848	2264	1177

# COMPARISON TRANSCRIPTION (MULTILINGUAL MODEL)

*This table details the differences observed on some lines of the specific pages selected with the application of the model EHRI on the test set, compared to the manual transcription of the pages. The first column indicates the page from which the errors were retrieved. The last column indicates when there was a specific situation within the facsimile that might have prompted the prediction error.*

Document	Line	Correct Transcription	Model EHRI	Facsimile Particularity
English 1	1. 1	THE	EE	
English 1	1. 1	BURGENLAND	BURGENIANp	
English 1	1. 1	JEWS	IEys	
English 1	1. 3	BOAT	POAT	
English 1	1. 5	On	on	
English 1	1. 5	1938	l0z8	
English 1	1. 5	Jewish	rewieb	
English 1	1. 5	inhabitants	inhebitante	
English 1	1. 5	of	og	
English 1	1. 5	Kittsee	Kittee	
English 1	1. 5	and	ann	
English 1	1. 6	AUSTRIA	AUSTBLA	
English 1	1. 10	from	fron	
English 1	1. 14	starved	staryed	
English 1	1. 15	bayonets	bavonets	
English 1	1. 25	Danube	Danuhe	
English 1	1. 25	Royka	Royvka	
English 1	1. 26	years	vears	
English 1	1. 29	Jewish	Jewiwh	two-character overlay

English 2	l. 2	approached	apnroached	
English 2	l. 2	saving	saying	
English 2	l. 20	lutely	utely	two-character overlay
English 2	l. 22	Refugees	Refuges	
English 2	l. 35	£100	f100	
English 2	l. 37	fifty	fiftyv	
Danish 1	l. 1	No. 56.	Ho. 26.	
Danish 1	l. 2	Jødelovgivningen	Jødelovivmingem	
Danish 1	l. 2	Bulgarien	mulgsrien	
Danish 1	l. 3	16.2.1943.	19.2.1943.	
Danish 1	l. 4	Ges	ces	
Danish 1	l. 4	No	Ho	
Danish 1	l. 4	143	139	
Danish 1	l. 5	4	3	
Danish 1	l. 5	Gennemslag	Genaemslag	
Danish 1	l. 6	Henvisning	Rerviemning	
Danish 1	l. 6	til	ti	
Danish 1	l. 6	tidligere	tidligære	
Danish 1	l. 6	Indberetninger	Dudberetminger	
Danish 1	l. 6	Konsulatet	konsulatév	
Danish 1	l. 7	ifølge	ifelge	
Danish 1	l. 7	Forordning	Fororchning	
Danish 1	l. 8	rigsministeren	rigeministeren	
Danish 1	l. 8	Presse	Fresse	
Danish 1	l. 9	Provinsen	Provingem	
Danish 1	l. 9	fremgik	frengik	
Danish 1	l. 10	Konsulatets	Konzulatets	
Danish 1	l. 10	Indberetning	Indberetaing	
Danish 1	l. 10	94	3	
Danish 1	l. 10	arbejdslø-	artejdsle-	
Danish 1	l. 11	kun	hun	
Danish 1	l. 11	faa	fa	
Danish 1	l. 12	Sofia	Gofia	
Danish 1	l. 12	gennemfører	gememférer	
Danish 1	l. 12	Kommissariatet	Kommiesariatet	

Danish 1	l. 12	Jødeanliggender	Jødeanliggender	
Danish 1	l. 12	en	on	
Danish 1	l. 13	udvisning	udvisnine	
Danish 1	l. 13	Hver	Bwer	
Danish 1	l. 13	Familie	Fanilie	
Danish 1	l. 13	en	em	
Danish 1	l. 13	Frist	Friet	
Danish 1	l. 14	at	et	
Danish 1	l. 14	forlade	forlæde	
Danish 1	l. 14	meddeles	medeles	
Danish 1	l. 14	hvilken	hvilkon	
Danish 1	l. 15	Flytningen	Plytningen	
Danish 1	l. 15	Jøder	Jeder	
Danish 1	l. 15	ikke	ihle	
Danish 1	l. 15	opfylder	ogfylder	
Danish 1	l. 15	denne	demme	
Danish 1	l. 15	Forordning	Fororduign	
Danish 1	l. 16	udvist	udwist	
Danish 1	l. 17	23.000	23.00	
Danish 1	l. 17	denne	demne	
Danish 1	l. 17	Forordning	Forordming	
Danish 1	l. 18	25.000	23.00	
Danish 1	l. 18	kun	hun	
Danish 1	l. 18	Statsborgere	Stataborgere	
Danish 1	l. 19	kan	kam	
Danish 1	l. 20	41.000	31.000	
Danish 1	l. 21	10.000	10.00	
Danish 1	l. 21	civilmobiliseret	eivilmobiliseret	
Danish 1	l. 21	Størsteparten	Sterstepartem	
Danish 1	l. 22	Vejbygningsarbejder	Vejboremingmarbejder	
Danish 1	l. 25	Gunnar	Cunmar	
Danish 1	l. 26	kgl	kol	
Danish 1	l. 26	Gesandtskab	Gezematsekab	
Danish 1	l. 27	Bukarest	Zakarest	
Danish 2	l. 1	Ved	ved	

Danish 2	l. 1	Beslaglæggelsen	Beslaglaggelsen	
Danish 2	l. 9	foretaget	forataget	
Danish 2	l. 12	paahviler	pashviler	
Danish 2	l. 26	Ejendom	Eiendom	
French 1	l. 2	à	N	
French 1	l. 2	à	A	
French 1	l. 2	près	prës	
French 1	l. 2	moitié	moitis	
French 1	l. 3	emmeés	emmeás	
French 1	l. 3	paraît	paraft	
French 1	l. 5	remontèrent	remontörent	
French 1	l. 5	où	ob	
French 1	l. 5	y	v	
French 1	l. 5	à	A	
French 1	l. 8	Il	D	
French 1	l. 9	voyage	vovage	
French 1	l. 9	arrêta	arreta	
French 1	l. 9	à	a	
French 1	l. 10	a/à	sn	
French 1	l. 11	âgés	Ægés	
French 1	l. 11	à	3	
French 1	l. 11	répondirent	rápondirent	
French 1	l. 12	déclarèrent	déclarörent	
French 1	l. 13	voyageaient	vovageaient	
French 1	l. 15	arrivèrent	arriverent	
French 1	l. 16	au	am	two-character overlay
French 1	l. 25	environs	envírons	
French 1	l. 26	français	francais	
French 1	l. 29	baraques	barac hes	
French 1	l. 29	mêmes	měmes	
French 1	l. 29	que	due	
French 1	l. 32	Le	le	
French 1	l. 32	très	trös	
French 1	l. 32	pénible	pánible	
French 1	l. 36	lesquels	lesduels	

French 1	l. 38	qu	du	
French 1	l. 39	à	a	
French 1	l. 42	mètres	mötres	
French 1	l. 42	être	Stre	
French 1	l. 44	paraît	paraft	
French 1	l. 44	5ème	S9me	
French 1	l. 44	que	due	
French 2	l. 1	Quant	Ouant	
French 2	l. 1	à	4	
French 2	l. 1	recevaient	recevsient	
French 2	l. 1	pain	hain	
French 2	l. 1	900	0	
French 2	l. 4	noirâtre	noirätre	
French 2	l. 6	Il	I1	
French 2	l. 8	très	trös	
French 2	l. 9	ayant	avant	
French 2	l. 10	quelqu	quelou	
French 2	l. 11	Ayant	Avant	
French 2	l. 11	à	A	
French 2	l. 11	réfugié	réfugís	
French 2	l. 12	choisi	chdisi	two-character overlay
French 2	l. 12	l'	1	
French 2	l. 15	25ans	2Sans	
French 2	l. 16	déjà	déja	
French 2	l. 16	frères	freres	
French 2	l. 16	à	a	
French 2	l. 17	derrière	derriere	
French 2	l. 18	complètement	completement	
French 2	l. 18	dégoûté	dégođté	
French 2	l. 22	réfugié	røfugig	
French 2	l. 23	qu	du	
French 2	l. 28	empêcher	empScher	
French 2	l. 29	l'	1	
French 2	l. 30	qu	cu	
French 2	l. 31	chaque	chadue	

French 2	l. 33	l'	l	
French 2	l. 33	indiqua	indioua	
French 2	l. 34	être	Stre	
French 2	l. 34	qu	du	
French 2	l. 35	vécut	vácut	
French 2	l. 35	une	nne	two-character overlay
French 2	l. 37	annonça	annonca	
French 2	l. 37	très	trös	
French 2	l. 41	même	meme	
French 2	l. 43	être	etre	
French 2	l. 44	portés	pprtés	
French 2	l. 45	Après	Apres	
French 2	l. 45	fût	füt	
French 2	l. 45	dûment	düment	
French 2	l. 46	à	a	
Italian 1	l. 1	4I7/92	4II/2	
Italian 1	l. 1	I6	16	
Italian 1	l. 1	marzo	margo	
Italian 1	l. 1	I943	L943	
Italian 1	l. 1	XXI	rII	
Italian 1	l. 2	portato	portsto	
Italian 1	l. 2	oro	orv	
Italian 1	l. 2	valuta	valnta	
Italian 1	l. 2	sulle	sule	
Italian 1	l. 2	persone	Persone	
Italian 1	l. 6	sono	eno	
Italian 1	l. 6	disgraziati	6lsgrasisati	
Italian 1	l. 6	quali	cali	
Italian 1	l. 6	può	pa	
Italian 1	l. 6	veramente	versæente	
Italian 1	l. 6	dire	Güire	
Italian 1	l. 6	han	her	
Italian 1	l. 7	occhi	ochi	
Italian 1	l. 8	preoccupazioni	preocupazioni	
Italian 1	l. 8	9.000	5.000	

Italian 1	l. 8	deportandi	deportand	
Italian 1	l. 10	sofferenze	soferenze	
Italian 1	l. 10	vengono	vensono	two-character overlay
Italian 1	l. 11	brutalità	brutalita	
Italian 1	l. 11	più	pih	
Italian 1	l. 13	modo	medo	
Italian 1	l. 13	è	5	
Italian 1	l. 14	anche	gncha	
Italian 1	l. 14	mesi	masi	
Italian 1	l. 14	veniva	kemiva	
Italian 1	l. 15	senza pane	gænzansne	
Italian 1	l. 15	guardie	gnardie	
Italian 1	l. 15	adoperano	adoperanc	
Italian 1	l. 16	ogni	cgni	
Italian 1	l. 16	fruste	fræste	
Italian 1	l. 17	promiscuità	promiscuith	
Italian 1	l. 18	impossibile	impossibile	
Italian 1	l. 18	ciò	cib	
Italian 1	l. 18	sono	somo	two-character overlay
Italian 1	l. 19	suicidi	saicidi	two-character overlay
Italian 1	l. 19	secondo	seconco	
Italian 1	l. 20	donne	donme	
Italian 1	l. 21	pazzia	pazia	
Italian 1	l. 22	qui	oni	
Italian 1	l. 23	Macedonia	Nacedonia	
Italian 1	l. 24	è	6	
Italian 1	l. 24	però	perb	
Italian 1	l. 25	grosso	srosso	
Italian 1	l. 25	che	cohe	
Italian 1	l. 25	eseguirlo	esesnirla	
Italian 1	l. 25	essi	si	
Italian 1	l. 25	più	pic	
Italian 1	l. 26	più	pih	
Italian 1	l. 27	I'll	1.11	
Italian 1	l. 28	vedendo	vedezdo	manual addition of characters



Italian 1	l. 31	autorità	autorith	two-character overlay
Italian 1	l. 32	israelita	isrselita	
Italian 1	l. 33	campi	csmpto	
Italian 1	l. 34	9	O	
Italian 1	l. 34	già	gih	
Italian 1	l. 35	409	408	
Italian 1	l. 35	ulterior	ulterion	
Italian 2	l. 3	decisione	decisions	writing slightly faded
Italian 2	l. 3	Rappresentante	Rapresentante	
Italian 2	l. 8	Il	I1	
Italian 2	l. 13	già	gia	
Italian 2	l. 16	Sicurezza	Siourezza	
Italian 2	l. 17	è	e	
Italian 2	l. 19	1°	19	
Italian 2	l. 20	2°	26	
Italian 2	l. 21	ebrei	sbrei	
Italian 2	l. 21	Reich	Feich	
Italian 2	l. 21	a	s	
Italian 2	l. 22	3°	36	
Italian 2	l. 22	città	citta	
Slovak 1	l. 1	Prezídium	Prezýdiu	
Slovak 1	l. 1	ministerstva	minierterstva	
Slovak 1	l. 1	vnútra	vnutra	
Slovak 1	l. 7	1891	1801	
Slovak 1	l. 16	minút	minut	
Slovak 1	l. 18	odsú	odsu	
Slovak 2	l. 2	Fuchsa	Ruchsa	
Slovak 2	l. 11	budú	budû	
Slovak 2	l. 12	zpät	zpät	
Slovak 2	l. 13	dôsledku	dêsledku	
Slovak 2	l. 17	budú	budû	
Slovak 2	l. 26	ďalšie	Galšie	

Table H.1: Prediction errors from the model EHRI NFC

# TABLES OF N-GRAMS (MULTILINGUAL MODEL)

---

*Created following the same rules as those of the tables in Annex E, those tables render, line by line, the errors retrieved from the test set of the EHRI dataset, to which the model EHRI (MEHRI) was applied.*

*This table only contains one set of three columns. The first column show the n-gram(s) wrongly predicted. The second column renders the number of occurrences of the n-gram of the reference in the training data of the model. The third column renders the number of occurrences of the n-gram of the prediction in the training data of the model.*

*As for the colours filling the cells, blue indicates that reference and prediction do not have the same number of characters, grey that the n-gram is not the right unit of characters for the table, and green that the prediction occurrence was greater than that of the reference.*

Correct Transcription	Model EHRI	Nb occ CT	Nb occ MEHRI
TH, E	EE		
BU, RG, EN, LA, ND	BU, RG, EN, IA, Np	2 3	1 ∅
JE, WS	IE, ys	7 ∅	2 54
BO, AT	PO, AT	∅	1
On	on	32	1087
Je, wi, sh	re, wi, eb	436 300	2130 182
in, ha, bi, ta, nt, s	in, he, bi, ta, nt, e	1220	963
of	og	970	196
Ki, tt, se, e	Ki, tt, ee, e	1697	85
an, d	an, n		
AU, ST, RI, A	AU, ST, BL, A	1	1
fr, om	fr, on	428	1087
st, ar, ve, d	st, ar, ye, d	1215	90
ba, yo, ne, ts	ba, vo, ne, ts	77	916
Da, nu, be	Da, nu, he	1915	963
Ro, yk, a	Ro, yv, ka		
ye, ar, s	ve, ar, s	90	1215
Je, wi, sh	Je, wi, wh	300	292
ap, pr, oa, ch, ed	ap, nr, oa, ch, ed	1028	22
sa, vi, ng	sa, yi, ng	489	28
lu, te, ly	ut, el, y		
Re, fu, ge, es	Re, fu, ge, s		
fi, ft, y	fi, ft, yv		
Jø, de, lo, vg, iv, ni, ng, en	Jø, de, lo, vi, vm, in, ge, m		
Bu, lg, ar, ie, n	mu, lg, sr, ie, n	154 792	404 45
Ge, s	ce, s	306	515
No	Ho	73	98
Ge, nn, em, sl, ag	Ge, na, em, sl, ag	625	1486
He, nv, is, ni, ng	Re, rv, ie, mn, in, g		
ti, l	ti		
ti, dl, ig, er, e	ti, dl, ig, æe, re		
In, db, er, et, ni, ng, er	Du, db, er, et, mi, ng, er	219 1254	85 1055
Ko, ns, ul, at, et	ko, ns, ul, at, év	110 976	907 16

Figure I.1: Bigrams Page 1 (Multilingual model)

if, øl, ge	if, el, ge	11	1303
Fo, ro, rd, ni, ng	Fo, ro, rc, hn, in, g		
ri, gs, mi, ni, st, er, en	ri, ge, mi, ni, st, er, en	133	1744
Pr, es, se	Fr, es, se	220	209
Pr, ov, in, se, n	Pr, ov, in, ge, m	1697	1744
fr, em, gi, k	fr, en, gi, k	820	3865
Ko, ns, ul, at, et, s	Ko, nz, ul, at, et, s	598	127
In, db, er, et, ni, ng	In, db, er, et, ai, ng	1254	141
ar, be, jd, sl, ø-	ar, te, jd, sl, e-	1915	2348
ku, n	hu, n	245	144
fa, a	fa		
So, fi, a	Go, fi, a	110	68
ge, nn, em, fø, re, r	ge, me, mf, ér, er		
Ko, mm, is, sa, ri, at, et	Ko, mm, ie, sa, ri, at, et	1145	707
Jø, de, an, li, gg, en, de, r	Jø, de, an, li, ge, nd, er		
en	on	3865	1087
ud, vi, sn, in, g	ud, vi, sn, in, e		
Hv, er	Bw, er	2	ø
Fa, mi, li, e	Fa, ni, li, e	1055	1254
en	em	3865	820
Fr, is, t	Fr, ie, t	1145	707
at	et	1286	976
fo, rl, ad, e	fo, rl, æd, e	473	13
me, dd, el, es	me, de, le, s		
hv, il, ke, n	hv, il, ko, n	817	907
Fl, yt, ni, ng, en	Pl, yt, ni, ng, en	25	16
Jø, de, r	Je, de, r	167	436
ik, ke	ih, le	210 817	157 1604
op, fy, ld, er	og, fy, ld, er	283	196
de, nn, e	de, mm, e	625	354
Fo, ro, rd, ni, ng	Fo, ro, rd, ui, gg, n		
ud, vi, st	ud, wi, st	489	935
de, nn, e	de, mn, e	625	69
Fo, ro, rd, ni, ng	Fo, ro, rd, mi, ng	1254	1055

Figure I.2: Bigrams Page 2 (Multilingual model)

ku, n	hu, n	245	144
St, at, sb, or, ge, re	St, at, ab, or, ge, re	61	391
ka, n	ka, m		
ci, vi, lm, ob, il, is, er, et	ei, vi, lm, ob, il, is, er, et	430	1139
St, ør, st, ep, ar, te, n	St, er, st, ep, ar, te, m	57	3349
Ve, jb, yg, ni, ng, sa, rb, ej, de, r	Ve, jb, or, em, im, gm, ar, be, jd, er		
Gu, nn, ar	Cu, nm, ar	17 625	5 31
kg, l	ko, l	15	907
Ge, sa, nd, ts, ka, b	Ge, ze, ma, ts, ek, ab		
Bu, ka, re, st	Za, ka, re, st	154	40
Ve, d	ve, d	134	1215
Be, sl, ag, læ, gg, el, se, n	Be, sl, ag, la, gg, el, se, n	16	837
fo, re, ta, ge, t	fo, ra, ta, ge, t	2130	1017
pa, ah, vi, le, r	pa, sh, vi, le, r	91	300
Ej, en, do, m	Ei, en, do, m	30	51
à	N		
à	A		
pr, ès	pr, ěs	∅	12
mo, it, ié	mo, it, is	1	1145
em, me, és	em, me, ás	421	123
pa, ra, ît	pa, ra, ft	∅	152
re, mo, nt, èr, en, t	re, mo, nt, ör, en, t	∅	38
où	ob	∅	358
y	v		
à	A		
ll	D		
vo, ya, ge	vo, va, ge	46	510
ar, rê, ta	ar, re, ta	∅	2130
à	a		
a/, à	sn		
âg, és	Æg, és	∅	20
à	3		
ré, po, nd, ir, en, t	rá, po, nd, ir, en, t	44	245
dé, cl, ar, èr, en, t	dé, cl, ar, ör, en, t	∅	38

Figure I.3: Bigrams Page 3 (Multilingual model)

vo, ya, ga, ie, nt	vo, <b>va</b> , ga, ie, nt	46	510
ar, ri, vè, re, nt	ar, ri, <b>ve</b> , re, nt	∅	1215
au	<b>am</b>	709	369
en, vi, ro, ns	en, <b>vi</b> , ro, ns	489	75
fr, an, ça, is	fr, an, <b>ca</b> , is	∅	334
ba, ra, qu, es	<b>ba</b> , ra, c , he, s		
mê, me, s	<b>mě</b> , me, s	∅	120
qu, e	<b>du</b> , e	56	217
Le	<b>le</b>	163	1604
tr, ès	tr, <b>ös</b>	∅	48
pé, ni, bl, e	<b>pá</b> , ni, bl, e	18	27
le, sq, ue, ls	le, <b>sd</b> , ue, ls	1	68
qu	<b>du</b>	56	217
à	a		
mè, tr, es	<b>mö</b> , tr, es	∅	15
êt, re	<b>St</b> , re	∅	249
pa, ra, ît	pa, ra, <b>ft</b>	∅	152
qu, e	<b>du</b> , e	56	217
Qu, an, t	<b>Ou</b> , an, t	11	∅
à	4		
re, ce, va, ie, nt	re, ce, <b>vs</b> , ie, nt	510	49
pa, in	<b>ha</b> , in	425	1220
no, ir, ât, re	no, ir, <b>ät</b> , re	∅	15
ll	<b>l1</b>	8	∅
tr, ès	tr, <b>ös</b>	∅	48
ay, an, t	<b>av</b> , an, t	20	200
qu, el, qu	qu, el, <b>ou</b>	56	431
Ay, an, t	<b>Av</b> , an, t	∅	1
à	A		
ré, fu, gi, é	ré, fu, gi, <b>ś</b>		
ch, oi, si	ch, <b>di</b> , si	10	1371
l'	<b>1</b>		
dé, jà	dé, <b>ja</b>	∅	186
fr, èr, es	fr, <b>er</b> , es	∅	3349

Figure I.4: Bigrams Page 4 (Multilingual model)

à	a		
de, rr, iè, re	de, rr, ie, re	∅	707
co, mp, lè, te, me, nt	co, mp, le, te, me, nt	∅	1604
dé, go, ût, é	dé, go, dt, é	∅	∅
ré, fu, gi, é	rø, fu, gi, g	44	13
qu	du	56	217
em, pê, ch, er	em, pS, ch, er	∅	∅
l'	1		
qu	cu	44	48
ch, aq, ue	ch, ad, ue	∅	473
l'	1		
in, di, qu, a	in, di, ou, a	56	431
êt, re	St, re	∅	249
qu	du	56	217
vé, cu, t	vá, cu, t	142	161
un, e	nn, e	1664	625
an, no, nç, a	an, no, nc, a	∅	192
tr, ès	tr, ös	∅	48
mê, me	me, me	∅	1797
êt, re	et, re	∅	976
po, rt, és	pp, rt, és	1101	87
Ap, rè, s	Ap, re, s	∅	2130
fû, t	fü, t	∅	133
dû, me, nt	dü, me, nt	∅	8
à	a		
ma, rz, o	ma, rg, o	140	217
XX, l	rl, l	11	∅
po, rt, at, o	po, rt, st, o	1286	2274
or, o	or, v		
va, lu, ta	va, ln, ta	158	146
su, ll, e	su, le		
pe, rs, on, e	Pe, rs, on, e	379	108
so, no	en, o		
di, sg, ra, zi, at, i	6l, sg, ra, si, sa, ti		

Figure I.5: Bigrams Page 5 (Multilingual model)

qu, al, i	ca, li		
pu, ò	pa		
ve, ra, me, nt, e	ve, rs, æe, nt, e	1797	1
di, re	Gü, ir, e		
ha, n	he, r	1120	963
oc, ch, i	oc, hi		
pr, eo, cc, up, az, io, ni	pr, eo, cu, pa, zi, on, i		
de, po, rt, an, di	de, po, rt, an, d		
so, ff, er, en, ze	so, fe, re, nz, e		
ve, ng, on, o	ve, ns, on, o	869	598
br, ut, al, it, à	br, ut, al, it, a		
pi, ù	pi, h		
mo, do	me, do	458	1797
è	5		
an, ch, e	gn, ch, a	2131	51
me, si	ma, si	1797	915
ve, ni, va	ke, mi, va	1215   1254	817   1055
gu, ar, di, e	gn, ar, di, e	151	51
ad, op, er, an, o	ad, op, er, an, c		
og, ni	cg, ni	196	∅
fr, us, te	fr, æs, te	461	7
pr, om, is, cu, it, à	pr, om, is, cu, it, h		
im, po, ss, ib, le	im, po, ss, ib, il, e		
ci, ò	ci, b		
so, no	so, mo	719	458
su, ic, id, i	sa, ic, id, i	260	512
se, co, nd, o	se, co, nc, o	1043	192
do, nn, e	do, nm, e	625	31
pa, zz, ia	pa, zi, a		
qu, i	on, i	56	1087
Ma, ce, do, ni, a	Na, ce, do, ni, a	258	200
è	6		
pe, rò	pe, rb	17	162
gr, os, so	sr, os, so	216	45

Figure I.6: Bigrams Page 6 (Multilingual model)



ch, e	co, he		
es, eg, ui, rl, a	es, es, ni, rl, a	416 13	1351 1254
es, si	si		
pi, ù	pi, c		
as, so, lu, to	ss, ol, ut, o		
pi, ù	pi, h		
ve, de, nd, o	ve, de, zd, o	1043	114
au, to, ri, tà	au, to, ri, th	∅	2264
is, ra, el, it, a	is, rs, el, it, a	1017	516
ca, mp, i	cs, mp, o	334	175
gi, à	gi, h		
ul, te, ri, or	ul, te, ri, on	877	1087
de, ci, si, on, e	de, ci, si, on, s		
Ra, pp, re, se, nt, an, te	Ra, pr, es, en, ta, nt, e		
ll	ll	8	∅
gi, à	gi, a		
Si, cu, re, zz, a	Si, ou, re, zz, a	48	431
è	e		
eb, re, i	sb, re, i	182	61
Re, ic, h	Fe, ic, h	153	61
a	s		
ci, tt, à	ci, tt, a		
Pr, ez, íd, iu	Pr, ez, ýd, iu	25	∅
mi, ni, st, er, st, va	mi, ni, et, er, st, va	2274	976
vn, út, ra	vn, ut, ra	43	418
mi, nú, t	mi, nu, t	9	279
od, sú	od, su	19	260
Fu, ch, sa	Ru, ch, sa	25	105
bu, dú	bu, dŭ	3	49
zp, äť	zp, ät	4	15
dô, sl, ed, ku	dě, sl, ed, ku	6	157
bu, dú	bu, dŭ	3	49
ďa, lš, ie	Ga, lš, ie	27	38

Figure I.7: Bigrams Page 7 (Multilingual model)

Correct Transcription	Model EHRI	Nb occ CT	Nb occ MEHRI
THE	EE		
BUR, GEN, LAN, D	BUR, GEN, IAN, p	∅	∅
JEW, S	Iey, s	2	∅
BOA, T	POA, T	∅	∅
On	on		
Jew, ish	rew, ieb	365   164	4   25
inh, abi, tan, ts	inh, ebi, tan, te	8	∅
of	og		
Kit, tse, e	Kit, tee, e	4	37
and	ann	602	66
AUS, TRI, A	AUS, TBL, A	∅	∅
fro, m	fro, n		
sta, rve, d	sta, rye, d	7	∅
bay, one, ts	bav, one, ts	∅	1
Dan, ube	Dan, uhe	32	4
Roy, ka	Roy, vka		
yea, rs	vea, rs	20	∅
Jew, ish	Jew, iwh	164	∅
app, roa, che, d	apn, roa, che, d	69	∅
sav, ing	say, ing	8	4
lut, ely	ute, ly		
Ref, uge, es	Ref, uge, s		
fif, ty	fif, tyv		
Jød, elo, vgi, vni, nge, n	Jød, elo, viv, min, gem		
Bul, gar, ien	mul, gsr, ien	40   73	6   1
Ges	ces	94	48
No	Ho		
Gen, nem, sla, g	Gen, aem, sla, g	243	1
Hen, vis, nin, g	Rer, vie, mni, ng		
til	ti		
tid, lig, ere	tid, lig, æer, e		
Ind, ber, etn, ing, er	Dud, ber, etm, ing, er	52   10	∅   ∅
Kon, sul, ate, t	kon, sul, até, v	50   92	159   1

Figure I.8: Trigrams Page 1 (Multilingual model)

ifø, lge	ife, lge	2	10
For, ord, nin, g	For, orc, hni, ng		
rig, smi, nis, ter, en	rig, emi, nis, ter, en	13	25
Pre, sse	Fre, sse	27	32
Pro, vin, sen	Pro, vin, gem	255	60
fre, mgi, k	fre, ngi, k	∅	∅
Kon, sul, ate, ts	Kon, zul, ate, ts	21	15
Ind, ber, etn, ing	Ind, ber, eta, ing	10	36
arb, ejd, slø, -	art, ejd, sle, -	23   ∅	64   15
kun	hun	33	14
faa	fa		
Sof, ia	Gof, ia	14	∅
gen, nem, før, er	gem, emf, ére, r		
Kom, mis, sar, iat, et	Kom, mie, sar, iat, et	48	67
Jød, ean, lig, gen, der	Jød, ean, lig, end, er		
en	on		
udv, isn, ing	udv, isn, ine	241	92
Hve, r	Bwe, r	∅	∅
Fam, ili, e	Fan, ili, e	18	3
en	em		
Fri, st	Fri, et		
at	et		
for, lad, e	for, læd, e	45	1
med, del, es	med, ele, s		
hvi, lke, n	hvi, lko, n	18	27
Fly, tni, nge, n	Ply, tni, nge, n	1	1
Jød, er	Jed, er	165	21
ikk, e	ihl, e	60	1
opf, yld, er	ogf, yld, er	3	∅
den, ne	dem, me	759	175
For, ord, nin, g	For, ord, uig, gn		
udv, ist	udw, ist	5	∅
den, ne	dem, ne	759	175
For, ord, nin, g	For, ord, min, g	90	240

Figure I.9: Trigrams Page 2 (Multilingual model)

kun	hun	33	14
Sta, tsb, org, ere	Sta, tab, org, ere	20	5
kan	kam	93	100
civ, ilm, obi, lis, ere, t	eiv, ilm, obi, lis, ere, t	13	25
Stø, rst, epa, rte, n	Ste, rst, epa, rte, m	ø	39
Vej, byg, nin, gsa, rbe, jde, r	Vej, bor, emi, mgm, arb, ejd, er		
Gun, nar	Cun, mar	3	ø
kgI	kol	2	24
Ges, and, tsk, ab	Gez, ema, tse, kab		
Buk, are, st	Zak, are, st	5	2
Ved	ved	21	99
Bes, lag, læg, gel, sen	Bes, lag, lag, gel, sen	17	66
for, eta, get	for, ata, get	36	12
paa, hvi, ler	pas, hvi, ler	79	39
Eje, ndo, m	Eie, ndo, m	29	1
à	N		
à	A		
prè, s	prě, s	ø	ø
moi, tié	moi, tis	ø	84
emm, eés	emm, eás	ø	ø
par, aît	par, aft	ø	49
rem, ont, ère, nt	rem, ont, öre, nt	ø	5
où	ob		
y	v		
à	A		
ll	D		
voy, age	vov, age	ø	6
arr, êta	arr, eta	ø	36
à	a		
a/à	sn		
âgé, s	Ægé, s	ø	ø
à	3		
rép, ond, ire, nt	ráp, ond, ire, nt	4	ø
déc, lar, ère, nt	déc, lar, öre, nt	ø	5

Figure I.10: Trigrams Page 3 (Multilingual model)

voy, aga, ien, t	vov, aga, ien, t	∅	6
arr, ivè, ren, t	arr, ive, ren, t	∅	39
au	am		
env, iro, ns	env, íro, ns	6	∅
fra, nça, is	fra, nca, is	∅	1
bar, aqu, es	bar, ac, hes		
mêm, es	měm, es	∅	∅
que	due	14	10
Le	le		
trè, s	trö, s	∅	∅
pén, ibl, e	pán, ibl, e	9	6
les, que, ls	les, due, ls	14	10
qu	du		
à	a		
mèt, res	möt, res	∅	∅
êtr, e	Str, e	∅	46
par, aît	par, aft	∅	49
que	due	14	10
Qua, nt	Oua, nt	9	∅
à	4		
rec, eva, ien, t	rec, evs, ien, t	16	1
pai, n	hai, n	6	4
noi, rât, re	noi, rät, re	∅	1
ll	ll		
trè, s	trö, s	∅	∅
aya, nt	ava, nt	∅	25
que, lqu	que, lou	∅	13
Aya, nt	Ava, nt	∅	∅
à	A		
réf, ugi, é	réf, ugi, ś		
cho, isi	chd, isi	66	∅
l'	1		
déj, à	déj, a		
frè, res	fre, res	∅	50

Figure I.11: Trigrams Page 4 (Multilingual model)

à	a		
der, riè, re	der, rie, re	∅	66
com, plè, tem, ent	com, ple, tem, ent	∅	88
dég, oût, é	dég, odt, é	∅	∅
réf, ugi, é	røf, ugi, g	∅	∅
qu	du		
emp, êch, er	emp, Sch, er	∅	193
l'	1		
qu	cu		
cha, que	cha, due	14	10
l'	1		
ind, iqu, a	ind, iou, a	2	11
êtr, e	Str, e	∅	46
qu	du		
véc, ut	vác, ut	∅	1
une	nne	14	25
ann, onç, a	ann, onc, a	∅	9
trè, s	trö, s	∅	∅
mêm, e	mem, e	∅	32
êtr, e	etr, e	∅	5
por, tés	ppr, tés	71	∅
Apr, ès	Apr, es		
fût	füt	∅	∅
dûm, ent	düm, ent	∅	∅
à	a		
mar, zo	mar, go		
XXI	rll	∅	∅
por, tat, o	por, tst, o	37	8
oro	orv	25	3
val, uta	val, nta	23	22
sul, le	sul, e		
per, son, e	Per, son, e	131	72
son, o	eno		
dīs, gra, zia, ti	6ls, gra, sis, ati		

Figure I.12: Trigrams Page 5 (Multilingual model)

qua, li	cal, i		
può	pa		
ver, ame, nte	ver, sæe, nte	51	∅
dir, e	Güi, re		
han	her	57	188
occ, hi	och, i		
pre, occ, upa, zio, ni	pre, ocu, paz, ion, i		
dep, ort, and, i	dep, ort, and		
sof, fer, enz, e	sof, ere, nze		
ven, gon, o	ven, son, o	25	155
bru, tal, ità	bru, tal, ita	∅	36
più	pih	∅	2
mod, o	med, o	32	103
è	5		
anc, he	gnc, ha	35	∅
mes, i	mas, i	21	15
ven, iva	kem, iva	60	21
gua, rdi, e	gna, rdi, e	17	1
ado, per, ano	ado, per, anc	21	35
ogn, i	cgn, i	2	∅
fru, ste	fræ, ste	2	∅
pro, mis, cui, tà	pro, mis, cui, th		
imp, oss, ibl, e	imp, oss, ibi, le		
ciò	cib	∅	∅
son, o	som, o	155	143
sui, cid, i	sai, cid, i	1	18
sec, ond, o	sec, onc, o	31	9
don, ne	don, me		
paz, zia	paz, ia		
qui	oni	5	21
Mac, edo, nia	Nac, edo, nia	3	52
è	6		
per, ò	per, b		
gro, sso	sro, sso	52	1

Figure I.13: Trigrams Page 6 (Multilingual model)

che	coh, e		
ese, gui, rla	ese, <b>sni</b> , rla	3	1
ess, i	<b>si</b>		
più	<b>pic</b>	∅	∅
ass, olu, to	<b>sso, lut, o</b>		
più	<b>pih</b>	∅	2
ved, end, o	ved, <b>ezd</b> , o	162	9
aut, ori, tà	aut, ori, <b>th</b>		
isr, ael, ita	isr, <b>sel</b> , ita	19	97
cam, pi	<b>csm, po</b>	68	∅
già	<b>gih</b>	∅	∅
ult, eri, or	ult, eri, <b>on</b>		
dec, isi, one	dec, isi, <b>ons</b>	92	54
Rap, pre, sen, tan, te	<b>Rap, res, ent, ant, e</b>		
ll	<b>l1</b>		
già	<b>gia</b>	∅	∅
Sic, ure, zza	<b>Sio</b> , ure, zza	10	2
è	<b>e</b>		
ebr, ei	<b>sbr</b> , ei	5	2
Rei, ch	<b>Fei</b> , ch	31	3
a	<b>s</b>		
cit, tà	cit, <b>ta</b>		
Pre, zíd, iu	Pre, <b>zýd</b> , iu	6	∅
min, ist, ers, tva	min, <b>iet</b> , ers, tva	291	18
vnú, tra	<b>vnu</b> , tra	1	3
min, út	min, <b>ut</b>		
ods, ú	ods, <b>u</b>		
Fuc, hsa	<b>Ruc</b> , hsa	5	∅
bud, ú	bud, <b>û</b>		
zpä, t	zpä, <b>t</b>		
dôs, led, ku	<b>dës</b> , led, ku	1	∅
bud, ú	bud, <b>û</b>		
d'al, šie	<b>Gal</b> , šie	4	4

Figure I.14: Trigrams Page 7 (Multilingual model)



Correct Transcription	Model EHRI	Nb occ CT	Nb occ MEHRI
THE	EE		
BURG, ENLA, ND	BURG, ENIA, Np	∅	∅
JEWS	IEys	∅	∅
BOAT	POAT	∅	∅
On	on		
Jewi, sh	rewi, eb	149	∅
inha, bita, nts	inhe, bita, nte	5	∅
of	og		
Kitt, see	Kitt, eee		
and	ann		
AUST, RIA	AUST, BLA		
from	fron	122	15
star, ved	star, yed		
bayo, nets	bavo, nets	∅	∅
Danu, be	Danu, he		
Royk, a	Royv, ka		
year, s	vear, s	19	∅
Jewi, sh	Jewi, wh		
appr, oach, ed	apnr, oach, ed	13	∅
savi, ng	sayi, ng	1	1
lute, ly	utel, y		
Refu, gees	Refu, ges		
fift, y	fift, yv		
Jøde, lovg, ivni, ngen	Jøde, lovi, vmin, gem		
Bulg, arie, n	mulg, srie, n	39   1	∅   ∅
Ges	ces		
No	Ho		
Genn, emsl, ag	Gena, emsl, ag	4	∅
Henv, isni, ng	Rerv, iem n, ing		
til	ti		
tidl, iger, e	tidl, igæe, re		
Indb, eret, ning, er	Dudb, eret, ming, er	13   25	∅   4
Kons, ulat, et	kons, ulat, év	14	5

Figure I.15: Tetragrams Page 1 (Multilingual model)

iføl, ge	ifel, ge	2	∅
Foro, rdni, ng	Foro, rchn, ing		
rigs, mini, ster, en	rige, mini, ster, en	12	15
Pres, se	Fres, se	20	1
Prov, inse, n	Prov, inge, m	2	17
frem, gik	fren, gik	13	∅
Kons, ulat, ets	Konz, ulat, ets	14	5
Indb, eret, ning	Indb, eret, aing	25	∅
arbe, jdsl, ∅-	arte, jdsl, e-	17	3
kun	hun		
faa	fa		
Sofi, a	Gofi, a	14	∅
genn, emfø, rer	geme, mfér, er		
Komm, issa, riat, et	Komm, iessa, riat, et	9	∅
Jøde, anli, ggen, der	Jøde, anli, gend, er		
en	on		
udvi, snin, g	udvi, snin, e		
Hver	Bwer	∅	∅
Fami, lie	Fani, lie	18	∅
en	em		
Fris, t	Frie, t	2	8
at	et		
forl, ade	forl, æede		
medd, eles	mede, les		
hvil, ken	hvil, kon		
Flyt, ning, en	Plyt, ning, en	∅	∅
Jøde, r	Jede, r	156	13
ikke	ihle	60	∅
opfy, lder	ogfy, lder	1	∅
denn, e	demm, e	54	∅
Foro, rdni, ng	Foro, rdni, ggn		
udvi, st	udwi, st	3	∅
denn, e	demn, e	54	∅
Foro, rdni, ng	Foro, rdmi, ng	38	∅

Figure I.16: Tetragrams Page 2 (Multilingual model)

kun	hun		
Stat, sbor, gere	Stat, <b>abor</b> , gere	22	2
kan	kam		
civi, lmob, ilis, eret	<b>eivi</b> , lmob, ilis, eret	13	∅
Stør, step, arte, n	<b>Ster</b> , step, arte, <b>m</b>	∅	6
Vejb, ygni, ngsa, rbej, der	<b>Vejb</b> , orem, imgm, arbe, jder		
Gunn, ar	<b>Cunm</b> , ar	∅	∅
kgl	kol		
Gesa, ndts, kab	<b>Geze</b> , mats, ekab		
Buka, rest	<b>Zaka</b> , rest	5	1
Ved	ved		
Besl, aglæ, ggel, sen	Besl, <b>agla</b> , ggel, sen	6	1
fore, tage, t	<b>fora</b> , tage, t	33	2
paah, vile, r	<b>pash</b> , vile, r	4	∅
Ejen, dom	<b>Eien</b> , dom	23	∅
à	N		
à	A		
près	près		
moit, ié	moit, <b>is</b>		
emme, és	emme, <b>ás</b>		
para, ît	para, <b>ft</b>		
remo, ntèr, ent	remo, <b>ntör</b> , ent	∅	∅
où	ob		
y	v		
à	A		
ll	<b>D</b>		
voya, ge	<b>vova</b> , ge	∅	3
arré, ta	<b>arre</b> , ta	∅	5
à	a		
a/à	<b>sn</b>		
âgés	<b>Ægés</b>	∅	∅
à	3		
répo, ndir, ent	<b>rápo</b> , ndir, ent	∅	∅
décl, arèr, ent	décl, <b>arör</b> , ent	∅	∅

Figure I.17: Tetragrams Page 3 (Multilingual model)

voya, gaie, nt	vova, gaie, nt	∅	3
arri, vère, nt	arri, vere, nt	∅	5
au	am		
envi, rons	enví, rons	1	∅
fran, çais	fran, cais	∅	1
bara, ques	bara, c, hes		
même, s	měme, s	∅	∅
que	due		
Le	le		
très	trös	∅	∅
péni, ble	páni, ble	∅	5
lesq, uels	lesd, uels	∅	∅
qu	du		
à	a		
mètr, es	mötr, es	∅	∅
être	Stre	∅	∅
para, ît	para, ft		
que	due		
Quan, t	Ouan, t	∅	∅
à	4		
rece, vaie, nt	rece, vsie, nt	∅	∅
pain	hain	4	∅
noir, âtre	noir, ätre	∅	∅
ll	l1		
très	trös	∅	∅
ayan, t	avan, t	∅	∅
quel, qu	quel, ou		
Ayan, t	Avan, t	∅	∅
à	A		
réfu, gié	réfu, gís		
choi, si	chdi, si	1	∅
l'	1		
déjà	déja	∅	∅
frèr, es	frer, es	∅	∅

Figure I.18: Tetragrams Page 4 (Multilingual model)

à	a		
derr, ière	derr, iere	∅	8
comp, lète, ment	comp, lete, ment	∅	19
dégo, ûté	dégo, dté		
réfu, gié	røfu, gig	∅	∅
qu	du		
empê, cher	empS, cher	∅	∅
l'	1		
qu	cu		
chaq, ue	chad, ue	∅	∅
l'	1		
indi, qua	indi, oua		
être	Stre	∅	∅
qu	du		
vécu, t	vácu, t	∅	∅
une	nne		
anno, nça	anno, nca		
très	trös	∅	∅
même	meme	∅	∅
être	etre	∅	1
port, és	pprt, és	18	∅
Aprè, s	Aprè, s	∅	∅
fût	füt		
dûme, nt	düme, nt	∅	∅
à	a		
marz, o	marg, o	∅	9
XXI	rll		
port, ato	port, sto		
oro	orv		
valu, ta	valn, ta	8	∅
sull, e	sule		
pers, one	Pers, one	44	60
sono	eno		
disg, razi, ati	6lsg, rasi, sati		

Figure I.19: Tetragrams Page 5 (Multilingual model)

qual, i	cali		
può	pa		
vera, ment, e	vers, æent, e	10 72	64 ∅
dire	Güir, e		
han	her		
occh, i	ochi		
preo, ccup, azio, ni	preo, cupa, zion, i		
depo, rtan, di	depo, rtan, d		
soff, eren, ze	sofe, renz, e		
veng, ono	vens, ono	∅	19
brut, alit, à	brut, alit, a		
più	pih		
modo	medo	∅	∅
è	5		
anch, e	gnch, a	2	∅
mesi	masi	∅	∅
veni, va	kemi, va	∅	∅
guar, die	gnar, die	10	∅
adop, eran, o	adop, eran, c		
ogni	cgni	∅	∅
frus, te	fræs, te	∅	∅
prom, iscu, ità	prom, iscu, ith		
impo, ssib, le	impo, ssib, ile		
ciò	cib		
sono	somo	∅	∅
suic, idi	saic, idi	1	∅
seco, ndo	seco, nco		
donn, e	donm, e	∅	∅
pazz, ia	pazi, a		
qui	oni		
Mace, doni, a	Nace, doni, a	∅	∅
è	6		
però	perb	∅	∅
gros, so	sros, so	37	3

Figure I.20: Tetragrams Page 6 (Multilingual model)

che	cohe		
eseg, uirl, a	eses, nirl, a	∅ ∅	5 ∅
essi	si		
più	pic		
asso, luto	ssol, uto		
più	pih		
vede, ndo	vede, zdo		
auto, rità	auto, rith	∅	∅
isra, elit, a	isrs, elit, a	2	∅
camp, i	csmp, o	62	∅
già	gih		
ulte, rior	ulte, rion	7	∅
deci, sion, e	deci, sion, s		
Rapp, rese, ntan, te	Rapr, esen, tant, e		
ll	l1		
già	gia		
Sícu, rezz, a	Siou, rezz, a	∅	∅
è	e		
ebre, i	sbre, i	3	∅
Reic, h	Feic, h	14	∅
a	s		
citt, à	citt, a		
Prez, ídiu	Prez, ýdiu	3	∅
mini, ster, stva	mini, eter, stva	50	7
vnút, ra	vnut, ra	1	∅
minú, t	minu, t	1	13
odsú	odsu	1	2
Fuch, sa	Ruch, sa	5	∅
budú	budû	3	∅
zpět	zpät	3	∅
dôsl, edku	děsl, edku	1	∅
budú	budû	3	∅
ďalš, ie	Galš, ie	2	∅

Figure I.21: Tetragrams Page 7 (Multilingual model)

ANNEX II  
FIGURES





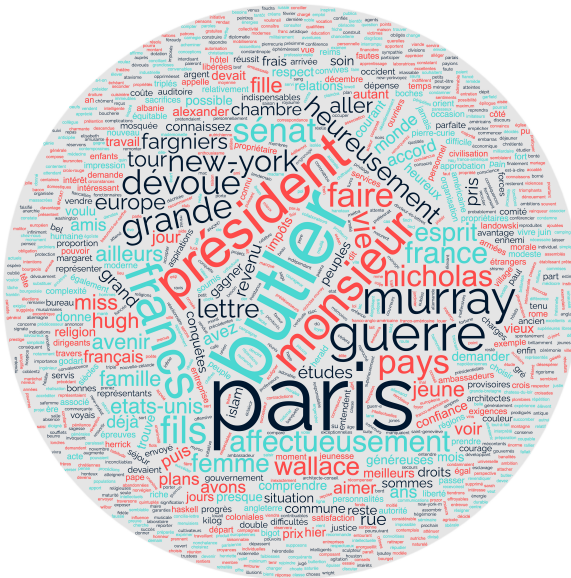


(a) Source test - 43 pages

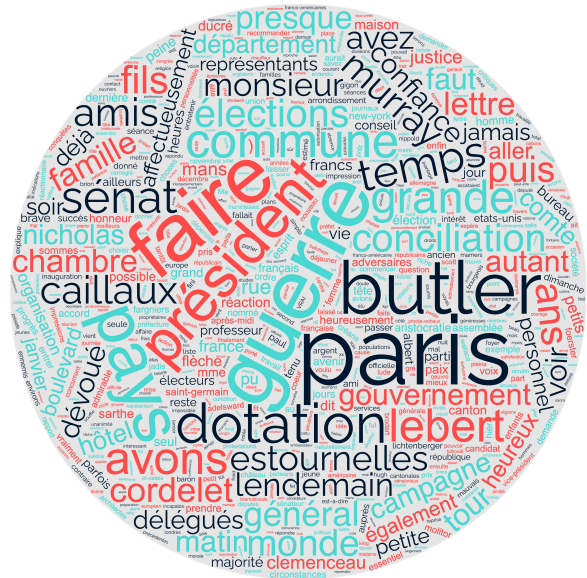


(b) Source test - 107 pages

Figure J.2: Complete list of words

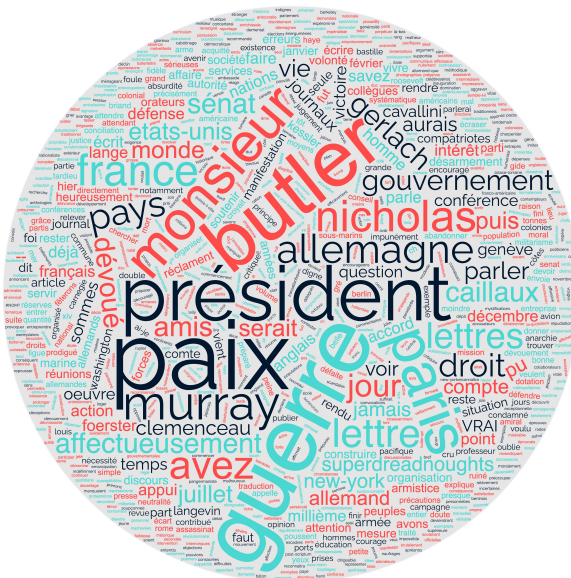


(a) Source test - 43 pages



(b) Source test - 107 pages

Figure J.3: List of words from the set Other



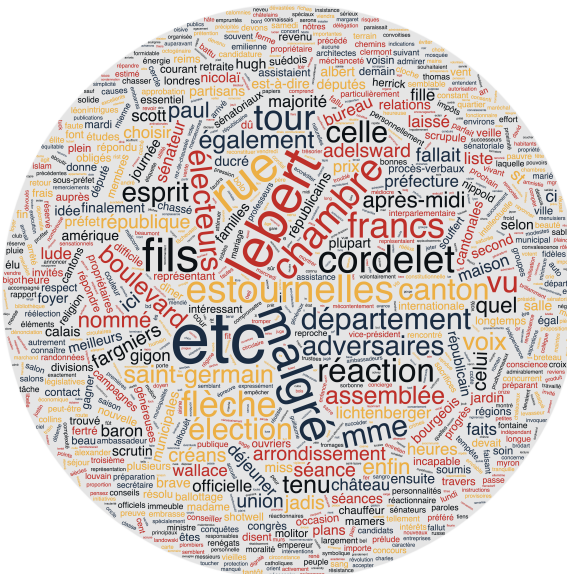
(a) Source test - 43 pages



(b) Source test - 107 pages

Figure J.4: List of words from the set War



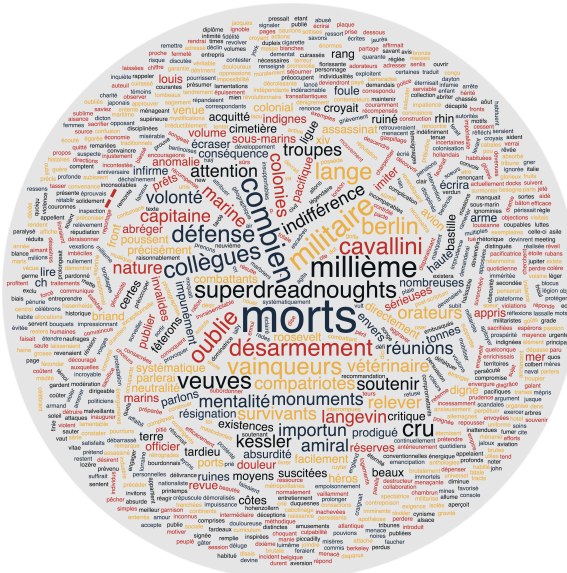


(a) Tokens

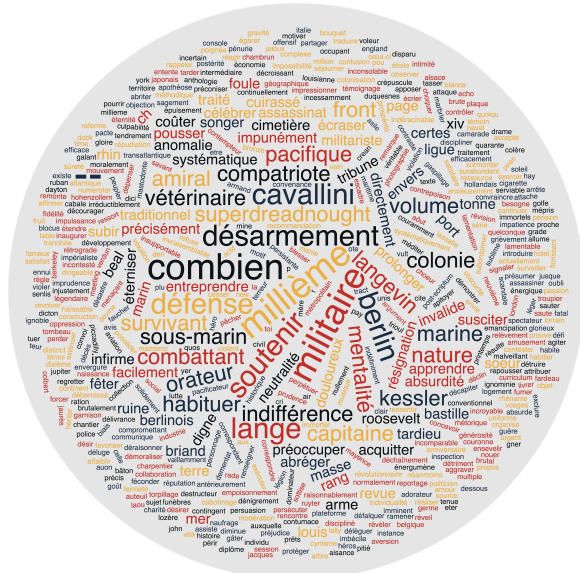


(b) Lemmas

Figure K.2: Unique to the set Other



(a) Tokens



(b) Lemmas

Figure K.3: Unique to the set War

## DIAGRAMS (CONTENT ANALYSIS)

---

*Produced for the Content analysis experiment, those diagrams present the division of the sets Other and War, by their part-of-speech tags. The set division exists in two styles: either the set has been tagged how it was, i.e. its tokens, or the set was reduced a bit by transforming all the tokens in their lemma format, i.e. a round-up of all the alternating and inflected forms of a same token. Lastly, the first diagrams present the division of the whole set, but the last two are different: the sets have been compared to extract the tokens/lemmas that were unique to the sets, and those that they had in common. The last three sets of diagrams present the distribution of those new sets.*

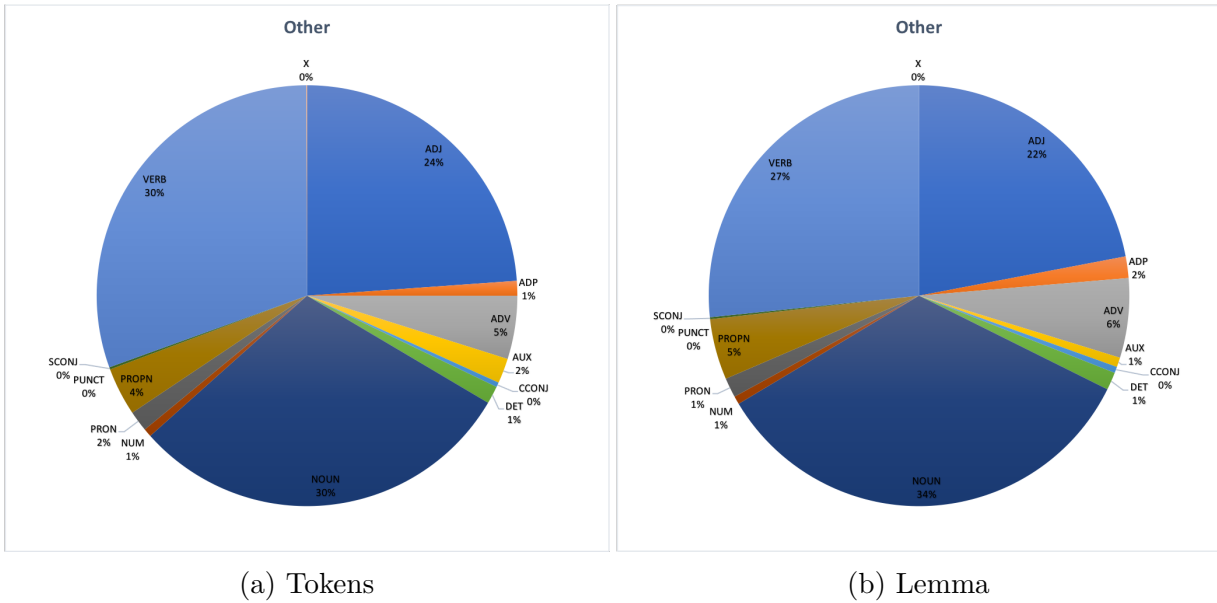


Figure L.1: Part-of-speech division from the set Other

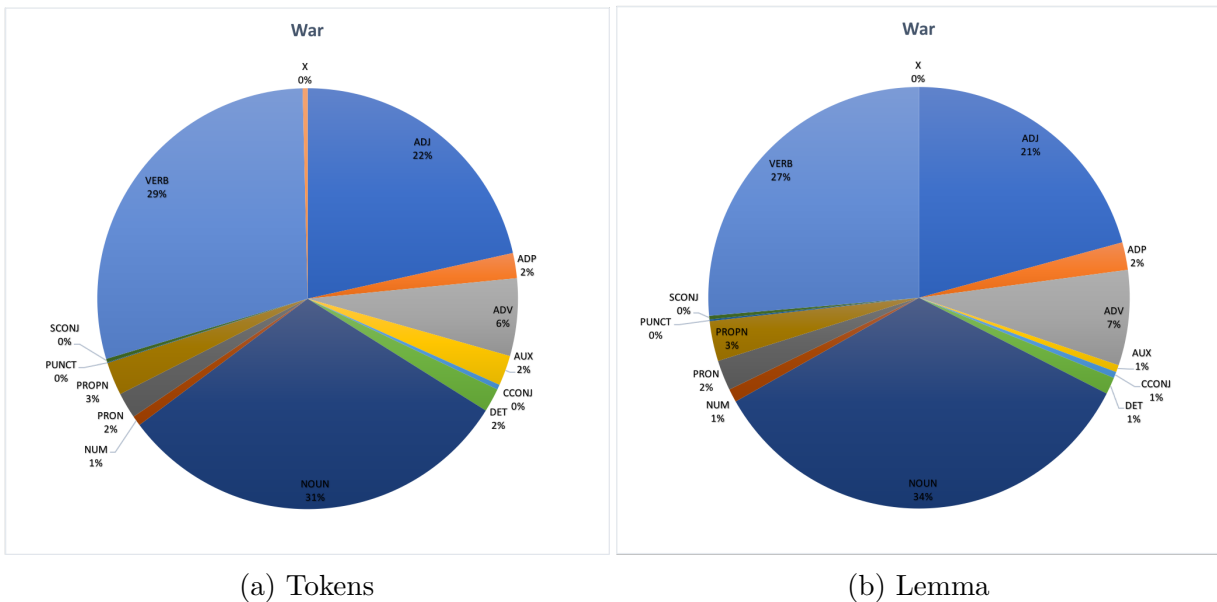


Figure L.2: Part-of-speech division from the set War

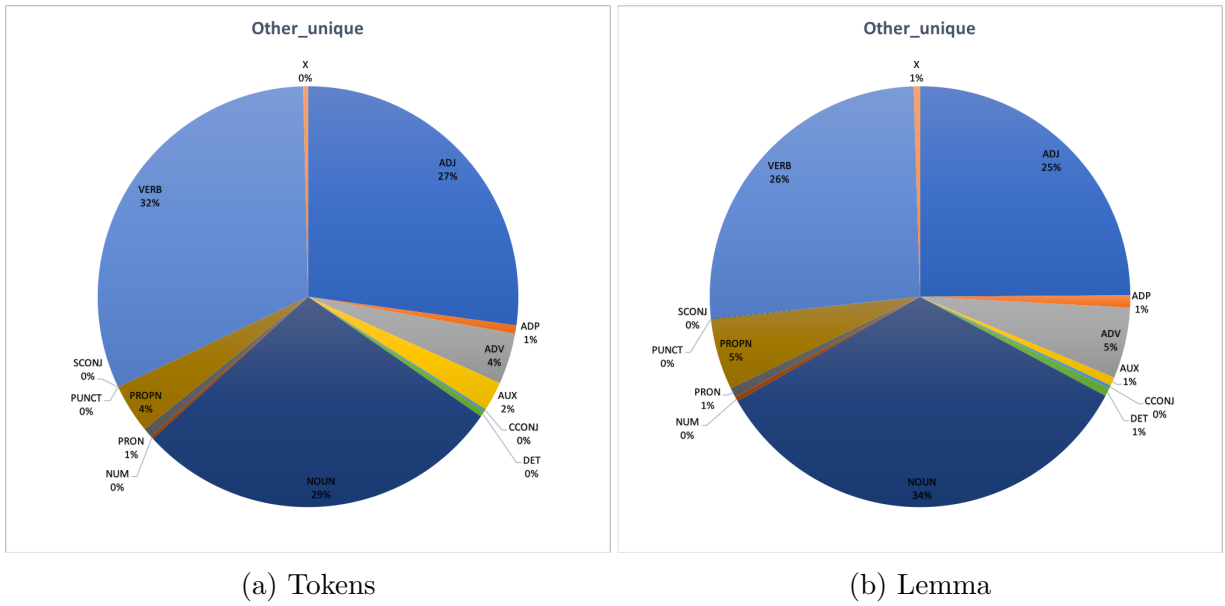


Figure L.3: Part-of-speech division of unique elements from the set Other (no commonality with the set War)

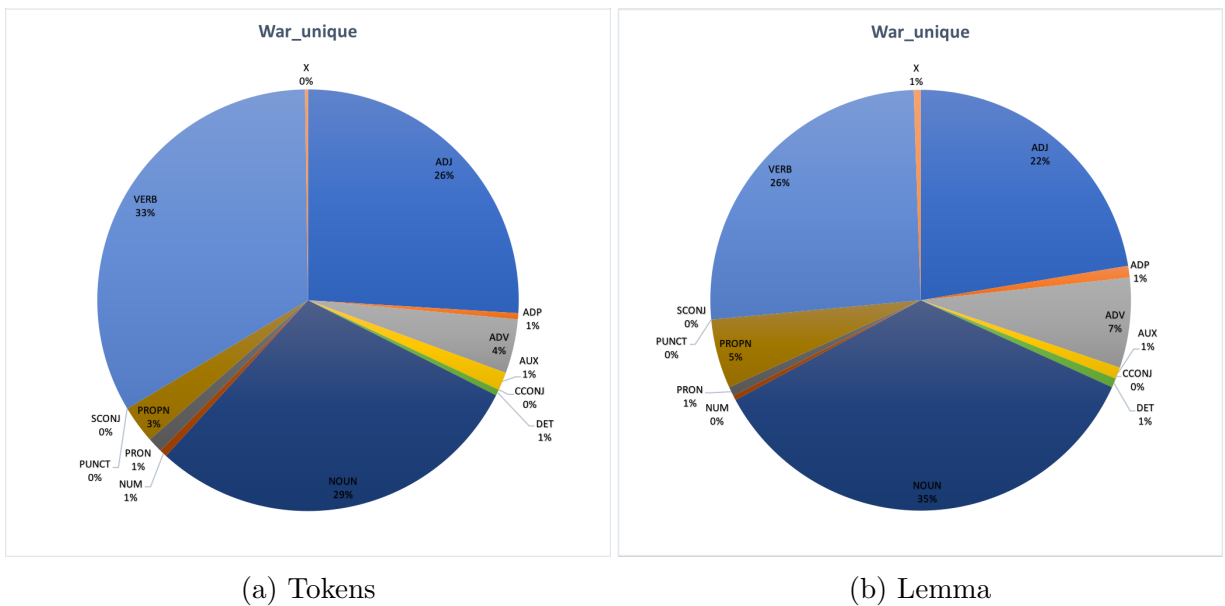
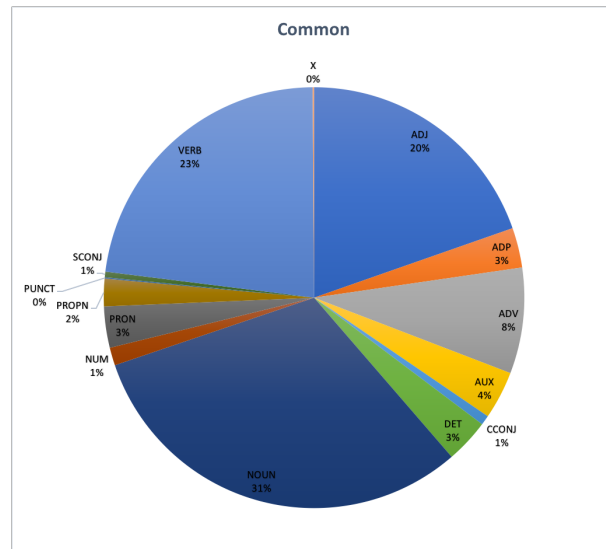
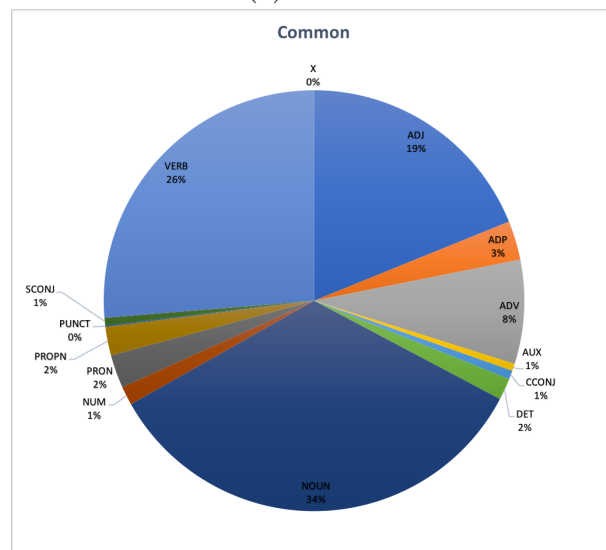


Figure L.4: Part-of-speech division of unique elements from the set War (no commonality with the set Other)





(a) Tokens



(b) Lemma

Figure L.5: Part-of-speech division of common elements between the sets Other and War

## BAR CHARTS (TOKEN ANALYSIS)

Created as part of the Token analysis experiment, those bar charts display the distribution of  $n$ -gram by number range. There are three types of bar charts: All Caps, that present the distribution of  $n$ -gram entirely made of uppercases tokens, Initials, that present the distribution of  $n$ -gram made of the first letter in uppercase and the following in lowercases, and Lowercases, that present the distribution of  $n$ -gram made entirely of lowercases tokens. Each type has three tables, one for each set of training data that were studied: Other, War, and Ground Truth. The number range varies, depending on the type of the bar chart, but also the set that is studied. For each number range, the  $n$ -grams are divided in colours: green for the bigrams, blue for the trigrams, and yellow for the tetragrams.

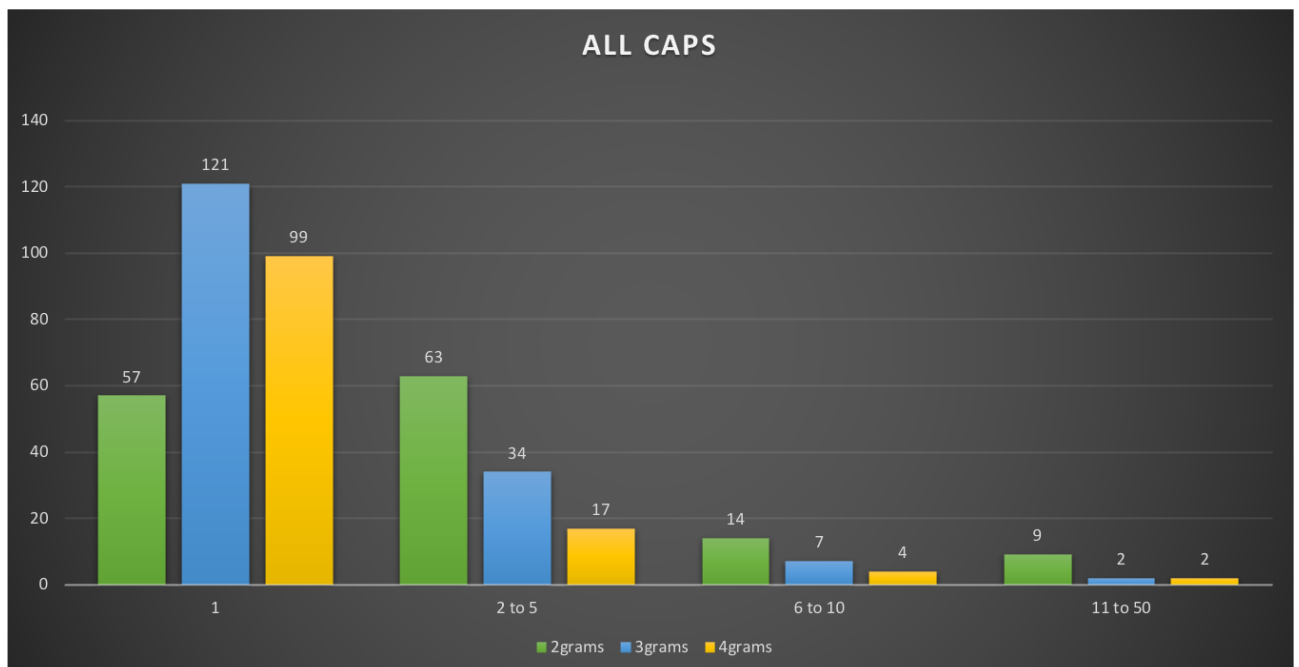


Figure M.1: All caps - set Other

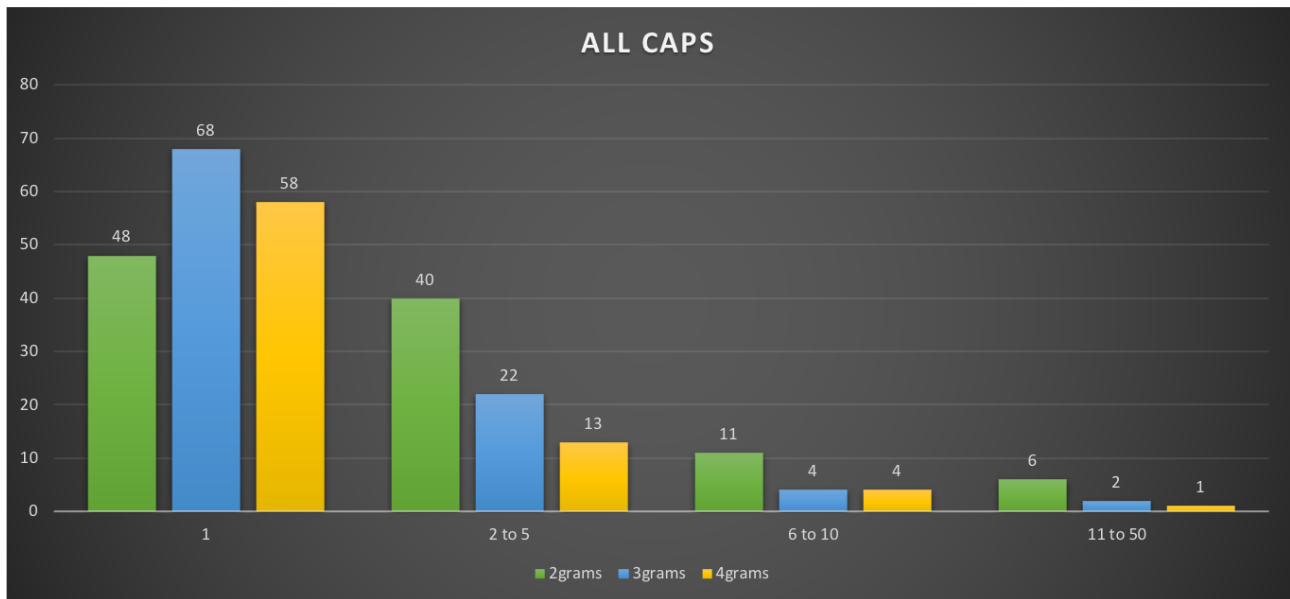


Figure M.2: All caps - set War

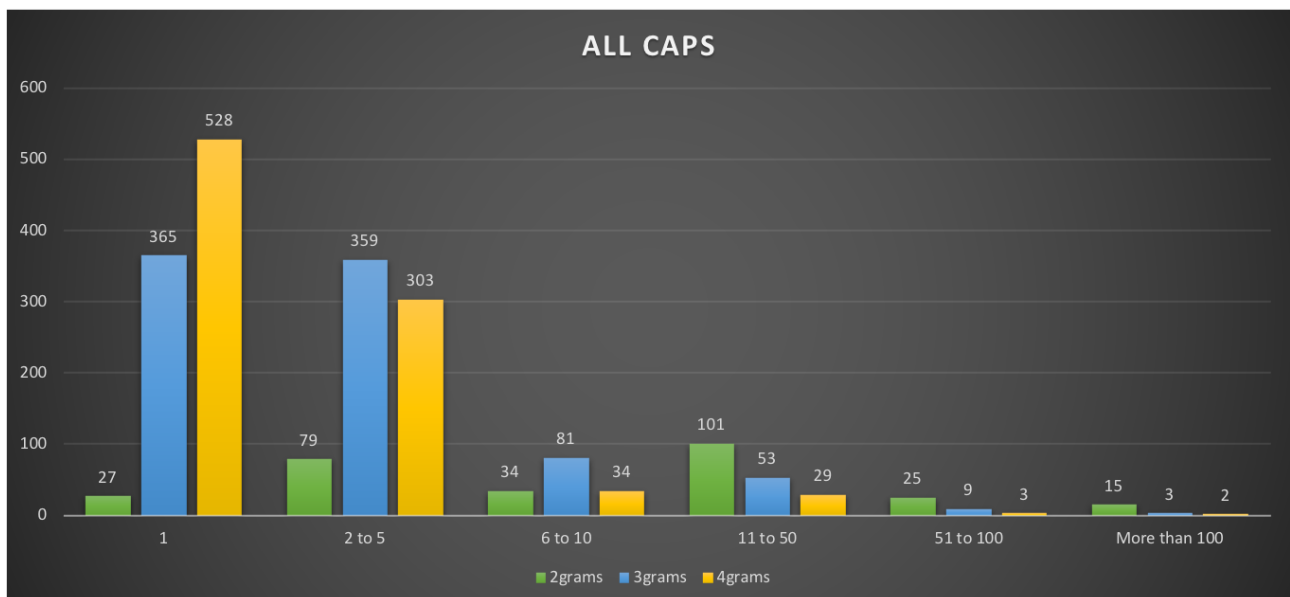


Figure M.3: All caps - set Ground Truth

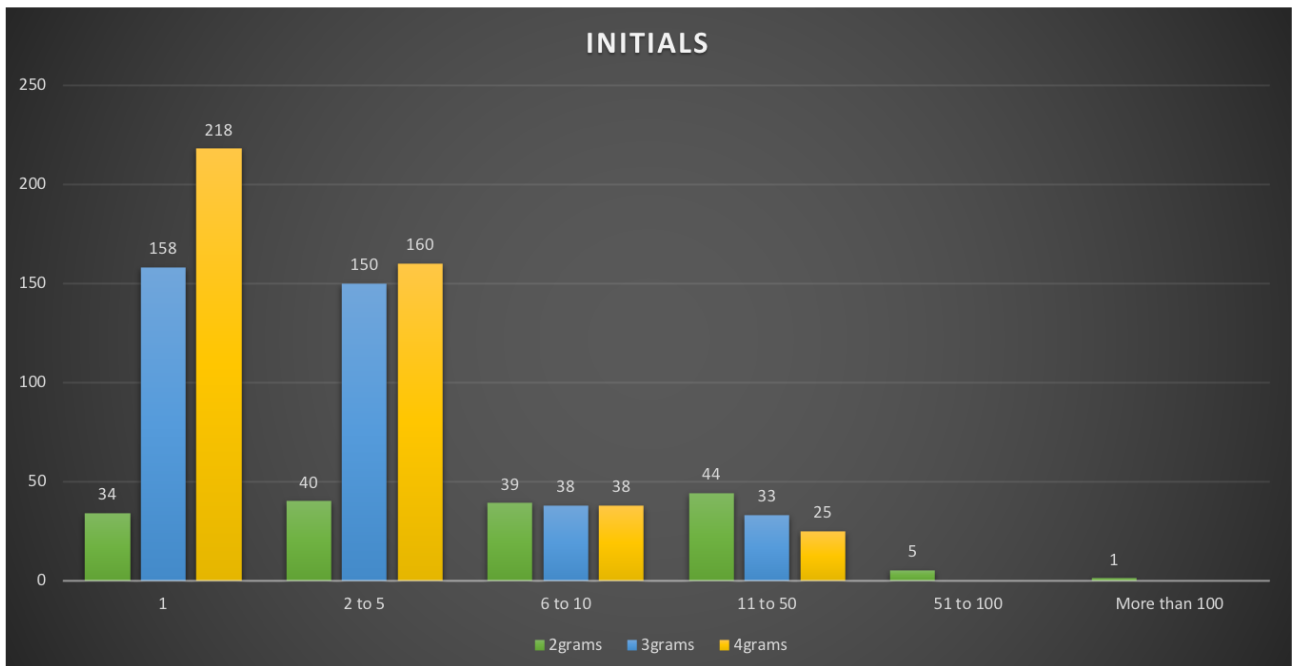


Figure M.4: Initials - set Other

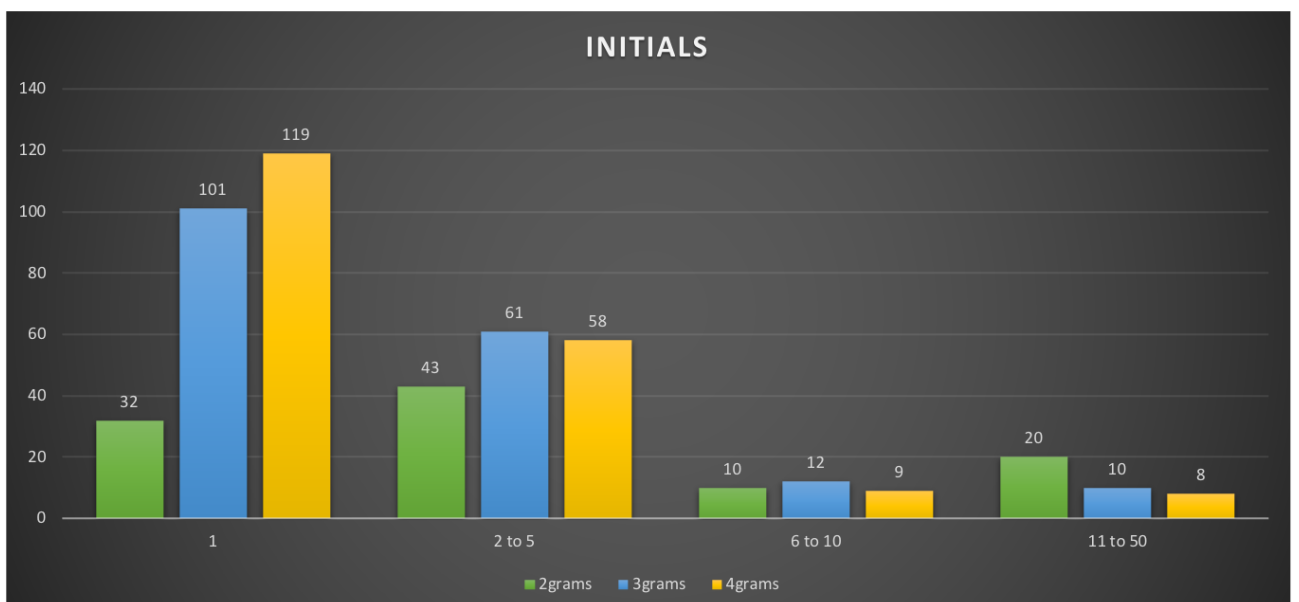


Figure M.5: Initials - set War

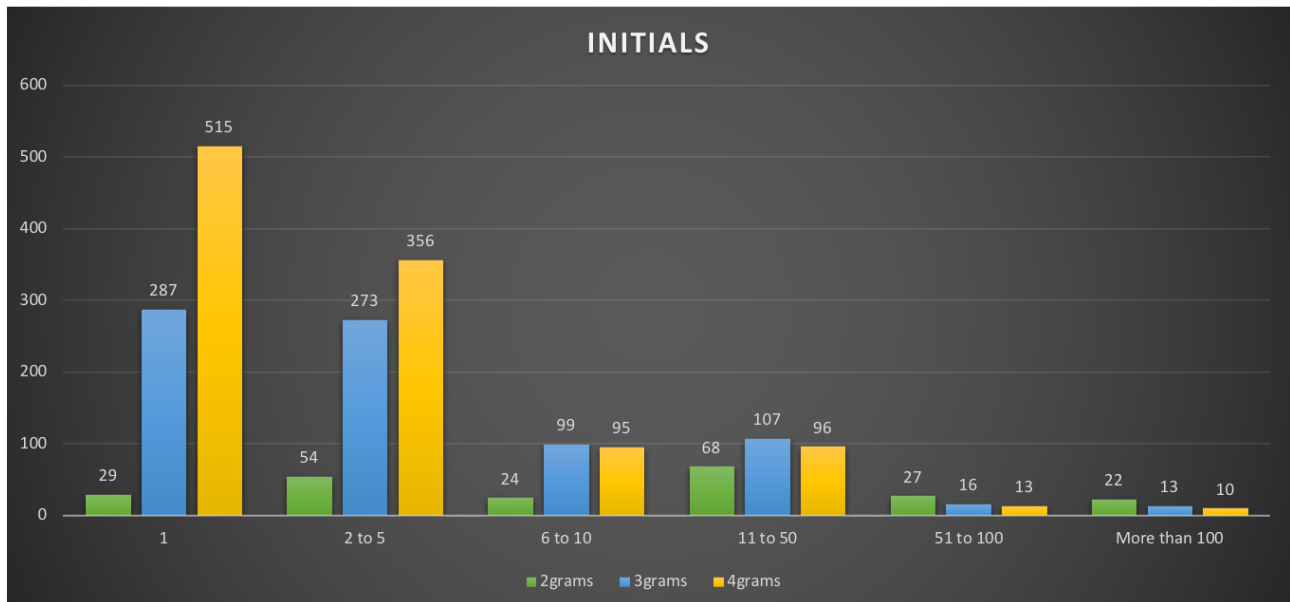


Figure M.6: Initials - set Ground Truth

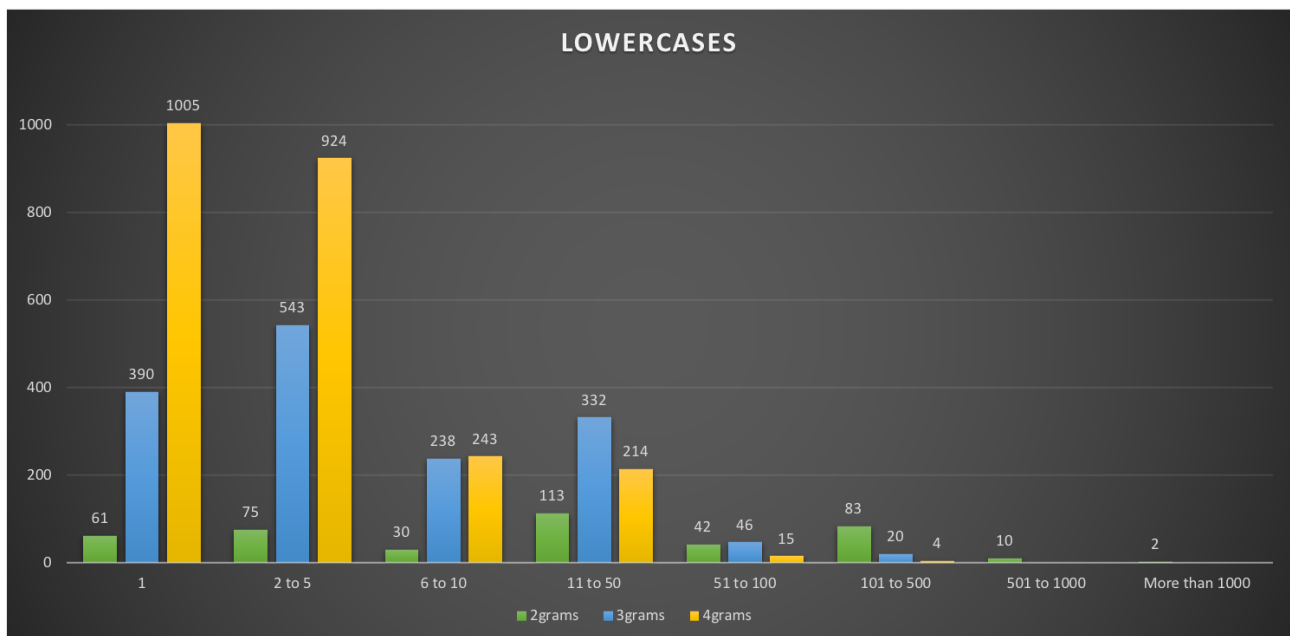


Figure M.7: Lowercases - set Other

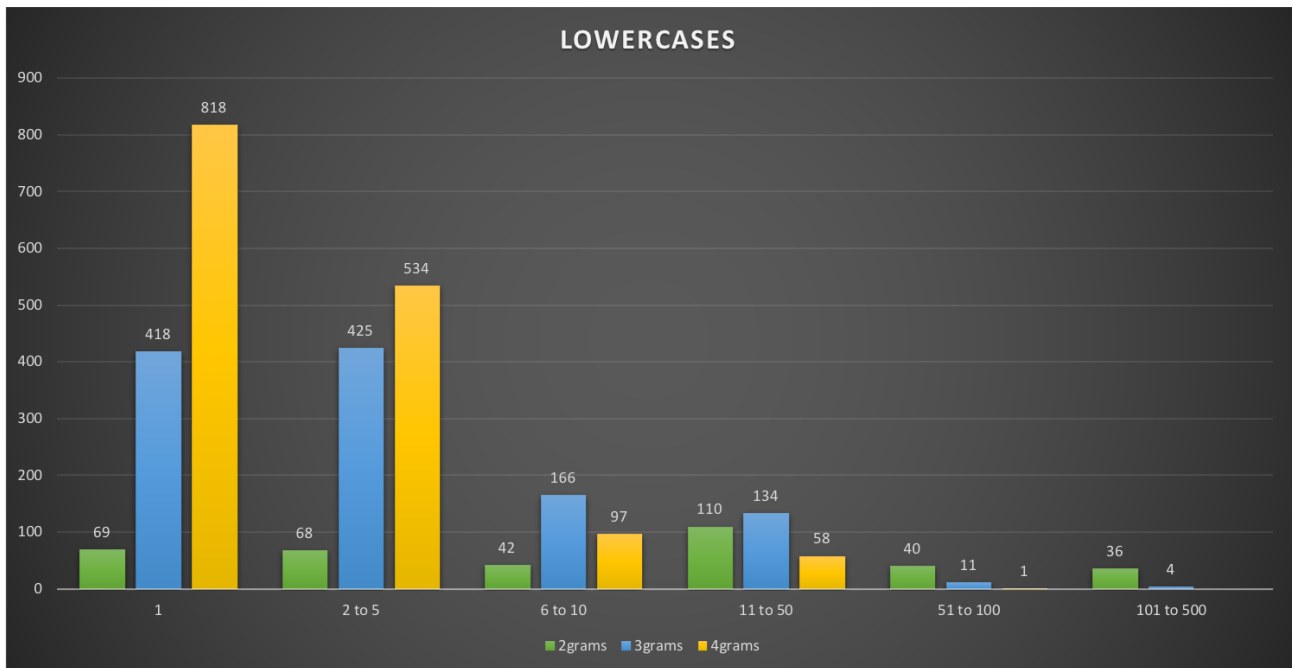


Figure M.8: Lowercases - set War

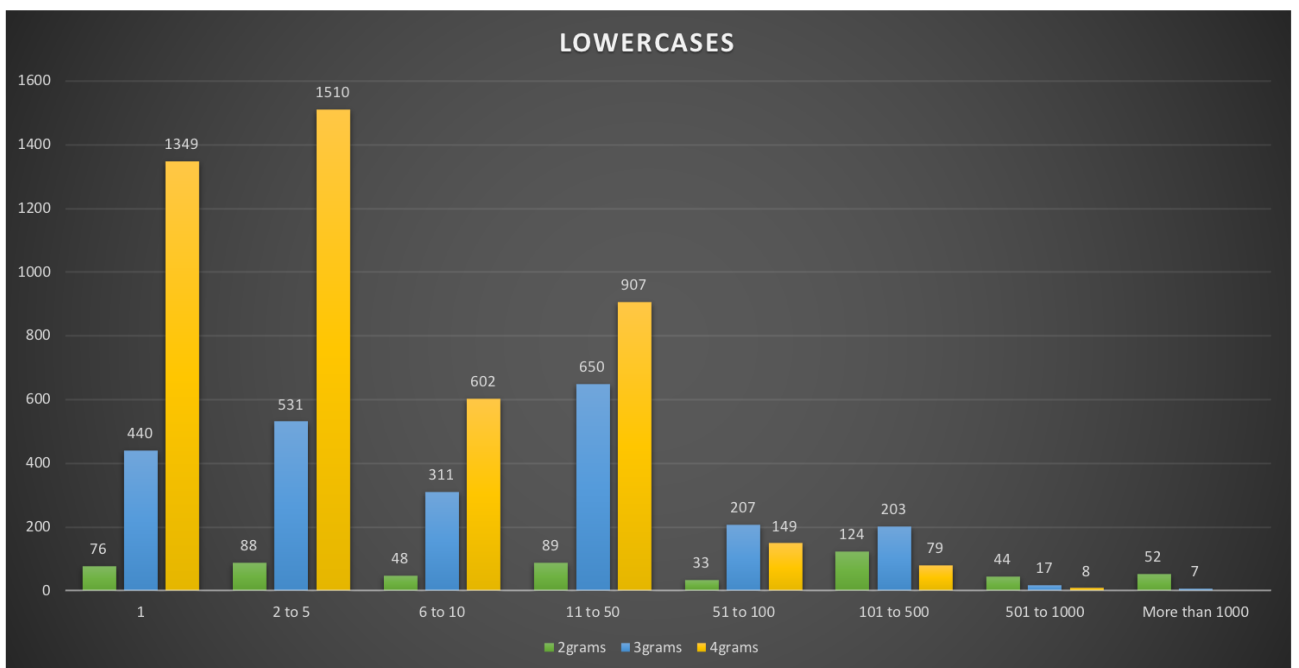


Figure M.9: Lowercases - set Ground Truth



# ANNEX III

## DATASETS



# PAUL D'ESTOURNELLES DE CONSTANT'

## CORRESPONDENCE

---

### N.1 Composition of the Set War (SW)

This set is available at [https://github.com/FloChiff/phd/tree/main/dataset/pec/set\\_war](https://github.com/FloChiff/phd/tree/main/dataset/pec/set_war)

Document	Number of pages	Date	Topics
Letter 617	3	1920-02-04	Treaties and war actions
Letter 678	2	1920-06-03	Trials and war
Letter 844	5	1921-06-18	War memorials and ceremonies
Letter 927	4	1921-12-09	Military occupation
Letter 948	4	1921-12-20	War memorials and ceremonies
Letter 957	4	1921-12-25	Armaments of war
Letter 1000	5	1922-01-23	Summary of the war correspondence
Letter 1364	2	1923-07-30	German sentiment
Letter 1367	2	1922-08-03	Post-war Germany

Table N.1: Composition of the set War

### N.2 Composition of the Set Other (SO)

This set is available at [https://github.com/FloChiff/phd/tree/main/dataset/pec/set\\_other](https://github.com/FloChiff/phd/tree/main/dataset/pec/set_other)

Document	Number of pages	Date	Topics
Letter 607	38	1920-01-12/13	Elections
Letter 722	2	1920-11-18	Colleague life
Letter 753	3	1920-12-23	Family marriage
Letter 846	3	1921-06-20	Diplomacy and theater
Letter 1029	3	1922-02-20	Carnegie project
Letter 1103	2	1922-06-12	Family relations
Letter 1170	4	1922-10-21	Religion and populations
Letter 1217	4	1922-12-14	Cost of post-war life
Letter 1358	17	1923-07-18	Work obligations

Table N.2: Composition of the set Other

### N.3 Composition of the Test Set

This set is available at [https://github.com/FloChiff/phd/tree/main/dataset/pec/specific\\_pages\\_studied](https://github.com/FloChiff/phd/tree/main/dataset/pec/specific_pages_studied)

Document	Set	Specificities
Letter 607 Page 3	Other	None
Letter 607 Page 17	Other	Narrow lines of text
Letter 678 Page 1	War	None
Letter 722 Page 1	Other	None
Letter 844 Page 1	War	Narrow lines of text
Letter 948 Page 1	War	None
Letter 1000 Page 3	War	None
Letter 1170 Page 3	Other	None
Letter 1358 Page 4	Other	Uppercases letters
Letter 1367 Page 1	War	Narrow lines of text

Table N.3: Composition of the test set

### N.4 Composition of the Ground Truth

The ground truth is available at <https://github.com/HTR-United/dahncorpus>

#	Name	Nb of images	GT for segmenter?	GT for recognizer?	Description
0	batch-00	(48)	y	n	Manual segmentation on pages with straight and regular lines
1	batch-01	(258)	y	y	Long letters, many lines per page, mostly straight lines but also narrow tight lines. Several pages contain lists, tables, and many capital letters words.
2	batch-02	(59)	y	y	Long letters, many lines per page, mostly straight lines but also narrow tight lines. Approximately ten letters with handwritten texts.
3	batch-03	(84)	y	y	Manual segmentation and complete transcription of letters. The letters sometimes differ in quality, writing colors, etc.
4	batch-04	(97)	n	y	Segmentation and transcription of chunks of texts or unique words to help recognition form specificities: capital letters, numbers, titles, recurring elements, handwritten elements, narrow tight parts of texts

Table N.4: Constitution of the Ground Truth

# EUROPEAN HOLOCAUST RESEARCH INFRASTRUCTURE EDITIONS

---

## O.1 List of the EHRI Online Editions Used to Create the Dataset

- "Early Holocaust Testimonies" (EHT): Written or transcribed oral testimonies on the persecution of the Jews in Nazi Germany;
- "Von Wien ins Nirgendwo: Die Nisko-Deportationen 1939" (Nisko): Testimonies and letters documenting the Nisko Plan, which aimed at creating a Jewish reservation, built by the Jews themselves, in Nisko and Lublin (Poland);
- "BeGrentze Flucht" (BF): Testimonies on the forced emigration of the Jewish population of Austria after its annexation in March 1938, focusing mostly on the situation at the Czechoslovakian border;
- "Diplomatic Reports" (DR): Reports written by foreign diplomats stationed in Nazi Germany to their respective Ministry of Foreign Affairs.

## O.2 Distribution of the Dataset

This dataset is fully (images and texts) available at <https://github.com/FloChiff/ehri-dataset>

Language	Collection	Documents	Lines
German	BF; Nisko; EHT	56	2287
English	BF; EHT; DR	54	1989
Czech	BF; EHT	46	1713
Danish	DR	36	1007
Hungarian	EHT	30	1334
Polish	EHT	15	468
Slovak	BF	15	395
Multilingual	BF; Nisko; DR; EHT	252	9193

Table O.1: Distribution of the EHRI training data

### O.3 Distribution of the Test Set

This test set is fully (images and texts) available at <https://github.com/FloChiff/phd/tree/main/dataset/ehri>

Image	Origin	Date	Lines	Link
Danish 1	DR	1943-05-28	27	<a href="#">Link</a>
Danish 2	DR	1942-06-15	27	<a href="#">Link</a>
English 1	BF	1938-08-05	33	<a href="#">Link</a>
English 2	BF	1938-05-09	42	<a href="#">Link</a>
French 1	EHT	1942-10-10	45	<a href="#">Link</a>
French 2	EHT	1942-10-10	46	<a href="#">Link</a>
Italian 1	DR	1943-03-16	36	<a href="#">Link</a>
Italian 2	DR	1941-10-07	22	<a href="#">Link</a>
Slovak 1	BF	1938-09-06	18	<a href="#">Link</a>
Slovak 2	BF	1938-04-21	26	<a href="#">Link</a>

Table O.2: Distribution of the EHRI test set

ANNEX IV  
RESOURCES

# PYTHON SCRIPTS

---

*This annex presents the Python scripts that were used during the various experiments of the PhD, in order to obtain data and results.*

## P.1 Lexicon Analysis

### P.1.1 [clean\\_groundtruth.py](#)

Cleaning of a transcription text by removing the punctuation and the transcription elements (symbols for handwriting, deletions, etc.), putting the entire text in lowercases, and deleting all the stop words, then outputting the input in a frequency list

### P.1.2 [groundtruth.py](#)

Retrieval of the text of a transcription from an XML output, in PAGE or ALTO

## P.2 Content Analysis

### P.2.1 [counting.py](#)

Count of the number of occurrences of each word in the text

### P.2.2 [creating\\_list\\_of\\_tokens](#)

Creation of lists of tokens from sets of words, presenting the common and unique elements between the sets

### P.2.3 [producing\\_part\\_of\\_speech\\_tagging.py](#)

Retrieval of the part-of-speech category of each element of given lists

### **P.2.4 [producing\\_tokens\\_and\\_lemmas.py](#)**

Cleaning of a transcription text by removing the punctuation and parts of text that are not relevant to a content analysis, and putting the entire text in lowercases, then outputting the input in two versions, the tokens and the lemmas

### **P.2.5 [removing\\_incorrect\\_tokens.py](#)**

Cleaning of lists of characters sequences to remove inexistent words after verification in a dictionary

## **P.3 Token Analysis**

### **P.3.1 [creating\\_list\\_of\\_tokens\\_from\\_ngrams.py](#)**

Creation of lists from sets of most and least popular n-grams, presenting the common and unique elements between the sets

### **P.3.2 [creating\\_list\\_of\\_tokens\\_from\\_texts.py](#)**

Transformation of a text into a list of tokens

### **P.3.3 [producing\\_dictionary\\_of\\_ngrams\\_occurrences.py](#)**

Transformation of lists of tokens into dictionaries of occurrences

### **P.3.4 [producing\\_list\\_of\\_ngrams.py](#)**

Production of lists of n-grams of various sizes, by wrapping each token from lists in sequences of two, three, or four characters

## **P.4 Token Error Analysis**

### **P.4.1 [producing\\_list\\_of\\_ngrams\\_from\\_several\\_models.py](#)**

Production of lists of n-grams of various sizes, by wrapping each token from lists in sequences of two, three, or four characters, from various lists



## **P.5 Multilingual Model EHRI**

### **P.5.1 [count\\_characters.py](#)**

Count of the number of occurrences of each character in the text

### **P.5.2 [producing\\_list\\_of\\_ngrams\\_from\\_several\\_models.py](#)**

Production of lists of n-grams of various sizes, by wrapping each token from lists in sequences of two, three, or four characters, from various lists (adapted to that experiment)

### **P.5.3 [creating\\_list\\_of\\_tokens\\_from\\_ngrams.py](#)**

Creation of lists from sets of most and least popular n-grams, presenting the common and unique elements between the sets (adapted to that experiment)

### **P.5.4 [producing\\_dictionary\\_of\\_ngrams\\_occurrences.py](#)**

Transformation of lists of tokens into dictionaries of occurrences (adapted to that experiment)

### **P.5.5 [producing\\_list\\_of\\_ngrams.py](#)**

Production of lists of n-grams of various sizes, by wrapping each token from lists in sequences of two, three, or four characters (adapted to that experiment)

# GLOSSARY

---

*This glossary has been made from the [glossary](#) of the *Harmonising ATR* project, as well as from some books referenced in the bibliography.*

**Accuracy** Score to measure the performance of an automatic text recognition model.

**Automatic Text Recognition (ATR)** Process of acquiring automatically, usually with machine learning technologies, digital textual data from a digitized analogue document.

**Automatic Speech Recognition (ASR)** Process of mapping any waveform into its appropriate string of characters.

**Deep Learning** Subpart of machine learning, in which artificial neural network, algorithms build to operate like the human brain, learns from large quantity of data.

**Character Error Rate (CER)** Count of the minimum number of character-level operations required to transform the ground truth text into the OCR output.

Formula:  $CER = \frac{\text{Substitution}(s) + \text{Insertion}(s) + \text{Deletion}(s)}{\text{Number of characters in the GT}}$

The lower the CER value (with 0 being a perfect score), the better the performance of the OCR model.

**Epoch** One entire passing of training data through the algorithm.

**FAIR Principles** Quality criteria developed in the context of data management, emphasising the importance of making data Findable, Accessible, Interoperable and Reusable.

**Fine-Tune** Technique allowing a pre-trained Machine Learning model to be specialized on a specific task.

**"Gold" Corpus** Data exclusively created and verified by humans, to obtain a perfect transcription.

**GPU** Graphics Processing Unit.

Specialized electronic circuit designed to accelerate calculations

**Ground Truth (GT)** (Perfect) transcription of a text (usually human-made), that will later be used to train model(s) fitted for the automatic transcription of a corpus/corpora.

**Handwritten Text Recognition (HTR)** Ability of a computer or device to take as input handwriting from sources such as printed physical documents, pictures and other devices, or to use handwriting as a direct input to a touchscreen and then interpret this as text.

**Lemma** Set of lexical forms having the same stem, the same major part-of-speech, and the same word sense.

**Levenshtein Distance** Metric for measuring the difference between two sequences. The Levenshtein distance between two words is the minimum number of single-character edits (insertions, deletions, or substitutions) required to change one word into the other.

**Machine Learning** Part of artificial intelligence that enables systems to learn from experience and improve without explicit instructions included in programming. Machine learning uses algorithms capable of analysing the data and making predictions or decisions based on patterns and trends within them.

**N-Gram** Contiguous sequence of  $n$  items from a given sample of text or speech.

**Neural Network** Type of machine learning model that is compositionally built from small units and that is typically designed to transform a set of numerical input values into a set of numerical output values. Each unit has one or more parameters that can be changed during model training. Combined, the parameters of the model form a specific memory which can represent features of the training data with the goal of improving desired output on novel data after training.

**Online/Offline Handwriting Recognition** Both functions pretty much the same, as it follows the definition for HTR given above, with the difference, for the Online HTR that, by writing on a Personal Digital Assistant, the recognition is helped by a sensor that picks up the pen-tip movements as well as pen-up/pen-down switching.

**Open Science** Movement that promotes accessibility, transparency, and collaboration in all areas of scientific research. The aim is to reach openness for scientific knowledge, data, methods, and publications so that they can be used at large, especially by other researchers.

**Optical Character Recognition (OCR)** Electronic or mechanical conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document or a scene-photo.

**Overfitting** Situation where a neural network ends up not only learning, but also memorizing the training data. It can be problematic, especially if noise is present in the training data.

**Part-Of-Speech (POS)** Nature of the word in a given sequence, which can help understand sentence structure and meaning.

**Part-Of-Speech (POS) Tagging** Process of assigning a part-of-speech to each word in a text.

**"Silver" Corpus** Data acquired by the prediction of the model made from the gold corpus.

**Token** Any contiguous sequence of alphanumeric characters, beginning with a letter and occurring between spaces, slashes, brackets, braces, parentheses, quotation marks, and punctuation marks.

**Transformers** Type of deep learning model in AI.

Transformer models use a mechanism called ‘self-attention’ for being able to process sequential data. They have outstanding performance in tasks such as Natural Language Processing and Machine Translation, given their ability to catch complex relationships in the input sequences.

**VGSL** Variable-size Graph Specification Language.

Specification of different network architectures for image processing purposes using a short definition string, that consists of an input block, one or more layers, and an output block.

**Word Error Rate (WER)** Count of the minimum number of word-level operations required to transform the ground truth text into the OCR output.

Formula:  $WER = \text{Word substitution(s)} + \text{Word insertion(s)} + \text{Word deletion(s)} / \text{Number of words in the GT}$

The value of the WER is usually higher than the CER value.

# WEB RESOURCES

---

*This annex gathers the links to the various web resources that were mentioned throughout this thesis.*

## ATR Software :

- ABBYY FineReader: <https://pdf.abbyy.com/fr/>
- Calamari: <https://github.com/Calamari-OCR>
- eScriptorium: <https://escriptorium.inria.fr/>
- Kraken: <https://kraken.re>
- OCRopus: <https://github.com/ocropus-archive/DUP-ocropy>
- Tesseract: <https://github.com/tesseract-ocr/tesseract>
- Transkribus: <https://www.transkribus.org/>
- Transym: <https://transym.com/>

## Documentation :

- Digital Intellectuals: <https://digitalintellectuals.hypotheses.org/category/dh-projects/dahn>
- eScriptorium documentation: <https://escriptorium.readthedocs.io/>
- HarmonizingATR: <https://harmoniseatr.hypotheses.org>

## Digital Scholarly Editions' Tools :

- Edition Visual Technology (EVT): <http://evt.labcd.unipi.it/>
- Gallica: <https://gallica.bnf.fr>

- HTR/OCR models: [https://zenodo.org/communities/ocr\\_models/](https://zenodo.org/communities/ocr_models/)
- HTR-United: <https://htr-united.github.io/index.html>
- IIIF: <https://iiif.io/>
- Nakala: <https://nakala.fr/>
- Omeka: <https://omeka.org/>
- SegmOnto: <https://segmonto.github.io/>
- SpaCy: <https://spacy.io/>
- TEI: <https://tei-c.org/>
- TEI Guidelines: <https://tei-c.org/release/doc/tei-p5-doc/en/html/index.html>
- TEI Publisher: <https://teipublisher.com/>

### European Holocaust Research Infrastructure (EHRI) :

- Dataset : <https://github.com/FloChiff/ehri-dataset>
- DiScholEd Edition: <https://discholed.huma-num.fr/exist/apps/discoled/index.html?collection=ehri>
- Online Editions: <https://www.ehri-project.eu/ehri-online-editions>
- Online Edition “BeGrenzte Flucht”: <https://begrenzte-flucht.ehri-project.eu/>
- Online Edition “Diplomatic Reports”: <https://diplomatic-reports.ehri-project.eu>
- Online Edition "Early Holocaust Testimonies": <https://early-testimony.ehri-project.eu/>
- Online Edition “Von Wien ins Nirgendwo: Die Nisko-Deportationen 1939”: <https://nisko-transports.ehri-project.eu/>
- Website: <https://www.ehri-project.eu/>

**Evaluation Tools** :

- CERberus: <https://github.com/WHaverals/CERberus>
- dinglehopper: <https://github.com/qurator-spk/dinglehopper>
- KaMI app: <https://huggingface.co/spaces/lterriell/kami-app>
- ocrevalUAtion: <https://github.com/impactcentre/ocrevalUAtion>

**Institutions** :

- Archives Départementales de la Sarthe: <https://archives.sarthe.fr/>
- Huma-Num: <https://www.huma-num.fr/>
- Mémorial de la Shoah (France): <https://www.memorialdelashoah.org/en>
- READ COOP: <https://readcoop.eu/>
- United States Holocaust Memorial Museum: <https://www.ushmm.org/>
- Yad Vashem's International Institute for Holocaust Research (Israel): <https://www.yadvashem.org/research.html>

**Paul d'Estournelles de Constant' Correspondence** :

- Edition: <https://discholed.huma-num.fr/exist/apps/discholed/index.html?collection=pec>
- Ground Truth: <https://github.com/HTR-United/dahncorpus>
- Images: <https://nakala.fr/collection/10.34847/nkl.adeb801d>
- Project DAHN: <https://github.com/FloChiff/DAHNPject>

**PhD** :

- Experiments: <https://github.com/FloChiff/phd/tree/main/experiments>
- Website: <https://flochiff.github.io/phd/>

ANNEX V  
PHD ACTIVITIES



# CHRONOLOGY

---

*This annex is a chronological retailing of my PhD, pointing out the meetings that I had, the courses that I followed, and some key dates.*

**October 14th, 2021:** Beginning of the thesis

**June 21st-22nd, 2022:** Seminar NLP meets DH, made of exchanges with other NLP/DH PhD students, presentation of my thesis subject and my progress and session of questions/answers on what to do next

**June 29th, 2022:** First-year meeting with the *Comité de Suivi Individuel (CSI)*, Elena Pierazzo and Jean-Philippe Magué, after sending them a report of what was done during this first year (experiments, readings, struggles, etc.)

**November 28th, 2022:** Progress meeting with my thesis supervisors, Anne Baillot and Laurent Romary, after sending them a report of what was done during this first year (experiments, readings, struggles, etc.)

**December 6th, 13th and 20th, 2022:** Training “How to write and publish your paper”, 18-hours course to help me improve in scientific english

**February 28th, 2023:** Training “*Découvrir la diffusion de la culture scientifique et technique - Module 1*”, 4-hours course to prepare me for an exercise of sharing and presenting my thesis subject to a larger audience

**June 3rd, 2023:** Progress meeting with my thesis supervisors, Anne Baillot and Laurent Romary, after sending them a report of what was done during this second year (experiments, readings, struggles, etc.), in order to determine the direction to take to continue my research

**June 14th, 2023:** Second-year meeting with the *Comité de Suivi Individuel (CSI)*, Elena Pierazzo and Jean-Philippe Magué, after sending them a report of what was done during this second year (experiments, readings, struggles, etc.)

**October 12th and 13th, 2023:** Training "Team Management", 12-hours course to learn about the notions and techniques of team management and leadership

**October 30th, 2023:** Progress meeting with my thesis supervisors, Anne Baillet and Laurent Romary, in order to determine the direction to take for the next year

**January 10th and 11th, 2024:** Training "Project Management", 12-hours course to learn about the notions and techniques of project management

**November 10th, 2023 to February 19th, 2024:** Training "*Ma thèse en 180 secondes*", introductory meeting about the training, followed by one-on-one meetings with a journalist helping with the writing of the prompt and days of stage training to prepare for the qualification, that happened on February 19th, 2024

**April 15th and 22th, 2024:** Progress meeting with my thesis supervisors, Anne Baillet and Laurent Romary, in order to validate my redaction plan, as well as answer some questions about the thesis and the defence

**November 20th, 2024:** PhD defense with my thesis supervisors, Anne Baillet and Laurent Romary, the rapporteurs, Ioana Galleron and Antoine Doucet, and the examiners, Elena Pierazzo and Jean-Philippe Magué

# CONFERENCES

---

*This annex lists the conferences that I attended and/or where I presented a paper during my PhD.*

## T.1 Next Gen TEI 2021

25-27 October 2021, Online

Presentation: *A TEI-based publication pipeline for historical egodocuments - the DAHN project*

## T.2 EVEILLE 2022

Day 4: 8 April 2022, R comme Réutilisable, Online

Program: <https://eveille2.sciencesconf.org/resource/page/id/19>

Intervention during the workshop *Pourquoi refaire, recréer ou fermer ?*

## T.3 Humanistica 2022

19-21 May 2022, Carrefour des arts et des sciences de l'Université de Montréal and Online

Website: <https://humanistica2022.sciencesconf.org/>

Program: <https://humanistica2022.sciencesconf.org/program>

Presentation: *Penser la réutilisabilité patrimoniale : présentation de la pipeline d'édition numérique de documents d'archives du projet DAHN*

## T.4 Documents anciens et reconnaissance automatique des écritures manuscrites

23-24 June 2022, École nationale des chartes, Paris

Website: <https://dahtr.sciencesconf.org/>

Member of the Organising Committee with Ariane Pinche.

Animation of the conference during two half-days and management of the recording.

## **T.5 Digital Humanities 2022 Responding to Asian Diversity**

25-29 July 2022, Toshi Center Hotel, Tokyo, Japan and Fully Online (Zoom)

Website: <https://dh2022.adho.org/>

Program: <https://dh2022.adho.org/program/presentations>

Presentation: *Take a sip of TEI and relax: a proposition for an end-to-end workflow to enrich and publish data created with automatic text recognition*

## **T.6 TEI 2022**

12-16 September 2022, Newcastle University, Newcastle, England

Website: <https://conferences.ncl.ac.uk/tei2022/>

Program: <https://conferences.ncl.ac.uk/tei2022/programme/>

Workshop: *From a collection of documents to a published edition : how to use an end-to-end publication pipeline [Full Day]*

## **T.7 DH2023**

July 10-14 2023, Graz, Austria

Website: <https://dh2023.adho.org/>

Simple attendance

## **T.8 From Source to Full Text: Workshop on Using Automatic Text Recognition (ATR)**

7-8 September 2023, German Historical Institute Paris, Paris, France

Member of the Organising Committee with Anne Baillet, Mareike König, Pauline Sychala and Olivier Richard.

Presentations:

- *Setting Up eScriptorium, Training Data, Models (and Where to Find Them), Predictions.*  
Introduction and Practical Exercises
- *Text Recognition and Correction*
- *Choice of Output Format and Re-Use*

## T.9 CHR 2023

December 6-8 2023, École pour l'informatique et les techniques avancées, Paris

Website: <https://2023.computational-humanities-research.org/>

Simple attendance

## T.10 Natural Language Processing Meets Holocaust Archives

March 27-28 2024, Charles University, Prague, République tchèque

Website: <https://www.clarin.eu/event/2024/natural-language-processing-meets-holocaust-archives>

Presentation : *Leveraging EHRI Online Editions for training automated edition tools*

Breakout session : <https://github.com/FloChiff/workshop-nlp-ehri>

## T.11 Workshop LREC-COLING 2024: Holocaust Testimonies as Language Resources

May 21 2024, Lingotto Conference Centre, Turin, Italy

Website: <https://www.clarin.eu/HTRes2024>

Online attendance

Presentation : *TEI Specifications for a Sustainable Management of Digitized Holocaust Testimonies*

## T.12 EHRI Academic Conference 2024

June 18th 2024, Polish Academy of Sciences, Warsaw, Poland

Website: <https://www.ehri-project.eu/ehri-academic-conference-researching-holocaust-digital-age>

---

Presentation: *Streamlining the Creation of Holocaust-related Digital Editions with Automatic Tools*

## **T.13 DH2024 Reinvention & Responsibility**

August 6-9 2024, George Mason University (GMU), Washington, United States of America

Website: <https://dh2024.adho.org/>

Presentation: *Collaboration and Transparency: A User-Generated Documentation for eScriptorium*

# TRAINING

---

*This annex lists the courses and MOOC that I attended on site or online during my PhD.*

## U.1 Ethique de la recherche

Date: April 15th, 2022

Duration: 6 hours

Subject: Le MOOC (Massive Open Online Course) proposé par l'Université de Lyon, centré sur l'éthique de la recherche, s'adresse prioritairement aux étudiants en thèse, mais concernent tous les chercheurs et citoyens qui souhaitent réfléchir aux transformations et implications contemporaines de la recherche, et aux nouveaux enjeux éthiques qu'elles soulèvent.

Link: [ethique-de-la-recherche/](#)

## U.2 How to write and publish your paper

Date: December 6th, 13th and 20th, 2022

Duration: 18 hours

Subject: This course will introduce the concepts of scientific writing and the procedures to publish scientific papers in international journals. PhD students will get practical advice and guidance on how to structure a research article, write it and get it published; explaining every step of the process, from choosing a suitable journal for your work, to presenting the results and citing references. We will work on your writing skills and discuss ethical issues important in scientific publishing.

Link: [formation/2325](#)

## U.3 Découvrir la diffusion de la culture scientifique et technique - Module 1

Date: February 28th, 2023

Duration: 4 hours

Subject: Ce module représente un module introductif à l'ensemble du programme de formations dédiées à la diffusion de la culture scientifique et technique. La participation à ce module sera un pré-requis pour prendre part aux autres formations de diffusion de la culture scientifique et technique (Communication orale, construire un atelier, la médiation scientifique avec des scolaires, Ma thèse en 180 secondes).

Link: [formation/2339](#)

## U.4 La science ouverte

Date: April 12th, 2023

Duration: 2 hours

Subject: Ce MOOC permet de se former à son rythme aux enjeux et aux pratiques de la science ouverte. Il rassemble les contributions de 38 intervenantes et intervenants issus de la recherche et des services de documentation, dont 10 doctorantes et doctorants. A travers ces points de vue variés, la place a été faite à différentes approches de l'ouverture des sciences, notamment en fonction des disciplines scientifiques.

Link: [la-science-ouverte/](#)

## U.5 Intégrité scientifique dans les métiers de la recherche

Date: May 22th, 2023

Duration: 2 hours

Subject: L'objectif de cette formation est de diffuser une culture de l'intégrité scientifique au sein des établissements. Plus qu'à transmettre des connaissances, il s'agit surtout de sensibiliser aux différents enjeux associés à l'intégrité scientifique et de favoriser une démarche critique en proposant les éléments de base nécessaires pour comprendre et porter les exigences de l'intégrité scientifique.

Link: [integrite-scientifique-dans-les-metiers-de-la-recherche/](#)



## U.6 Impacts environnementaux du numérique

Date: May 22th, 2023

Duration: 2 hours

Subject: MOOC pour se questionner sur les impacts environnementaux du numérique, apprendre à mesurer, décrypter et agir, pour trouver sa place de citoyen dans un monde numérique.

Link: [impacts-environnementaux-du-numerique/](#)

## U.7 L'intelligence artificielle... avec intelligence

Date: May 22th, 2023

Duration: 2 hours

Subject: MOOC citoyen accessible à toutes et à tous de 7 à 107 ans pour se questionner, expérimenter et comprendre ce qu'est l'Intelligence Artificielle... avec intelligence !

Link: [lintelligence-artificielle-avec-intelligence/](#)

## U.8 Team Management

Date: October 12th and 13th, 2023

Duration: 12 hours

Subject: Working in teams is inevitable in contemporary organisations regardless of its status (profit, not-for-profit or public). This two-day seminar aims to equip participants with the basics of team management, leadership, and membership. Participants will be exposed to concepts and frameworks used by team managers and leaders. Learners will engage in hands-on activities to understand team dynamics and how to navigate their way to be successful team leaders, managers and members.

Link: [formation/2560](#)

## U.9 Project Management

Date: January 10th and 11th, 2024

Duration: 12 hours

Subject: This workshop aims to equip PhD students from various disciplines with a strong foundation in project management concepts, skills, and practical applications. By achieving these objectives, students will be better prepared to manage projects effectively and contribute

to successful project outcomes in their academic and professional endeavours.

Link: [formation/2755](#)

## U.10 Ma thèse en 180 secondes

Date: November 10th, 2023 to February 19th, 2024

Duration: 14 hours

Subject: Le concours Ma thèse en 180 secondes permet aux doctorant(e)s de présenter leur sujet de recherche en termes simples à un auditoire profane et diversifié. Chaque participant(e) doit réaliser, en trois minutes, un exposé clair, concis et néanmoins convaincant de son projet de recherche. Le tout avec l'appui d'une seule diapositive ! La formation a pour objectif de vous accompagner dans l'écriture de votre pitch et sa mise en scène afin que vous puissiez présenter vos travaux de recherche en trois minutes.

Link: [formation/2784](#)



# BIBLIOGRAPHY

## Automatic Text Recognition (ATR)

- Eikvil, Line (1993), *OCR - Optical Character Recognition*.
- Naiemi, Fatemeh, Vahid Ghods, and Hassan Khalesi (June 2022), “Scene text detection and recognition: a survey”, in: *Multimedia Tools and Applications* 81, DOI: [10.1007/s11042-022-12693-7](https://doi.org/10.1007/s11042-022-12693-7).
- Romero, Verónica, Nicolás Serrano, et al. (Sept. 2011), “Handwritten Text Recognition for Historical Documents”, in: *Proceedings of the Workshop on Language Technologies for Digital Humanities and Cultural Heritage*, Association for Computational Linguistics, pp. 90–96, URL: <https://aclanthology.org/W11-4114> (visited on 10/24/2022).
- Romero, Verónica, Alejandro H. Toselli, et al. (Apr. 2016), “Handwriting Transcription and Keyword Spotting in Historical Daily Records Documents”, in: *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pp. 275–280, DOI: [10.1109/DAS.2016.70](https://doi.org/10.1109/DAS.2016.70).
- Sánchez, Joan Andreu et al. (May 2014), “Handwritten text recognition for historical documents in the transcriptorium project”, in: *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, Association for Computing Machinery, pp. 111–117, ISBN: 978-1-4503-2588-2, DOI: [10.1145/2595188.2595193](https://doi.org/10.1145/2595188.2595193), URL: <https://doi.org/10.1145/2595188.2595193> (visited on 10/24/2022).
- Ströbel, Phillip Benjamin, Simon Clematide, Martin Volk, and Tobias Hodel (Mar. 2022), *Transformer-based HTR for Historical Documents*, en, arXiv:2203.11008 [cs], URL: <http://arxiv.org/abs/2203.11008> (visited on 10/10/2023).

## ATR Tools

- Assefi, Mehdi (Dec. 2016), “OCR as a Service: An Experimental Evaluation of Google Docs OCR, Tesseract, ABBYY FineReader, and Transym”, in: *ISCV*.
- Breuel, Thomas (Jan. 2008), “The OCRopus open source OCR system”, in: vol. 6815, p. 68150, DOI: [10.1117/12.783598](https://doi.org/10.1117/12.783598).
- Chagué, Alix, Floriane Chiffolleau, and Hugo Scheithauer (Aug. 2024), “Collaboration and Transparency: A User-Generated Documentation for eScriptorium”, in: *DH2024 Reinvention & Responsibility*, Alliance of Digital Humanities Organizations, Washington D. C., United States, URL: <https://hal.science/hal-04594142>.

- Heliński, Marcin, M. Kmiecik, and Tomasz Parkola (2012), “Report on the comparison of Tesseract and ABBYY FineReader OCR engines”, en, in: *undefined*, URL: [http://lib.psnc.pl/Content/358/PSNC\\_Tesseract-FineReader-report.pdf](http://lib.psnc.pl/Content/358/PSNC_Tesseract-FineReader-report.pdf) (visited on 05/13/2022).
- Jain, Pooja, Kavita Taneja, and Harmunish Taneja (Apr. 2021), “Which OCR toolset is good and why : A comparative study”, en, in: *Kuwait Journal of Science* 48.2, Number: 2, ISSN: 2307-4116, DOI: [10.48129/kjs.v48i2.9589](https://doi.org/10.48129/kjs.v48i2.9589), URL: <https://journalskuwait.org/kjs/index.php/KJS/article/view/9589> (visited on 05/22/2024).
- Kahle, Philip et al. (Nov. 2017), “Transkribus - A Service Platform for Transcription, Recognition and Retrieval of Historical Documents”, in: *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 04, pp. 19–24, DOI: [10.1109/ICDAR.2017.307](https://doi.org/10.1109/ICDAR.2017.307).
- Kiessling, Benjamin (Dec. 2019), *Kraken - a Universal Text Recognizer for the Humanities*, fr, DOI: [10.34894/Z9G2EX](https://doi.org/10.34894/Z9G2EX), URL: <https://dataverse.nl/dataset.xhtml?persistentId=doi:10.34894/Z9G2EX> (visited on 05/03/2022).
- Kiessling, Benjamin et al. (Sept. 2019), “eScriptorium: An Open Source Platform for Historical Document Analysis”, in: *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, vol. 2, pp. 19–19, DOI: [10.1109/ICDARW.2019.10032](https://doi.org/10.1109/ICDARW.2019.10032).
- Nockels, Joe et al. (Sept. 2022), “Understanding the application of handwritten text recognition technology in heritage contexts: a systematic review of Transkribus in published research”, en, in: *Archival Science* 22.3, pp. 367–392, ISSN: 1389-0166, 1573-7500, DOI: [10.1007/s10502-022-09397-0](https://doi.org/10.1007/s10502-022-09397-0), URL: <https://link.springer.com/10.1007/s10502-022-09397-0> (visited on 03/16/2023).
- Patel, Chirag, Atul Patel, and Dharmendra Patel (Oct. 2012), “Optical Character Recognition by Open source OCR Tool Tesseract: A Case Study”, in: *International Journal of Computer Applications* 55, pp. 50–56, DOI: [10.5120/8794-2784](https://doi.org/10.5120/8794-2784).
- Reul, Christian et al. (Nov. 2019), “OCR4all—An Open-Source Tool Providing a (Semi-)Automatic OCR Workflow for Historical Printings”, in: *Applied Sciences* 9.22, p. 4853, ISSN: 2076-3417, DOI: [10.3390/app9224853](https://doi.org/10.3390/app9224853), URL: <http://dx.doi.org/10.3390/app9224853>.
- Smith, R. (Sept. 2007), “An Overview of the Tesseract OCR Engine”, in: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, vol. 2, ISSN: 2379-2140, pp. 629–633, DOI: [10.1109/ICDAR.2007.4376991](https://doi.org/10.1109/ICDAR.2007.4376991).

## Deep Learning

- Aggarwal, Charu C. (2023), *Neural Networks and Deep Learning: A Textbook*, en, Cham: Springer International Publishing, DOI: [10.1007/978-3-031-29642-0](https://doi.org/10.1007/978-3-031-29642-0), URL: <https://link.springer.com/10.1007/978-3-031-29642-0> (visited on 07/09/2024).
- Chollet, Francois (2017), *Deep Learning with Python*, 1st, USA: Manning Publications Co., ISBN: 1617294438.
- Haykin, Simon S. and Simon S. Haykin (2009), *Neural networks and learning machines*, en, 3rd ed, New York: Prentice Hall.
- Srivastava, Nitish et al. (Jan. 2014), “Dropout: a simple way to prevent neural networks from overfitting”, *in: J. Mach. Learn. Res.* 15.1, pp. 1929–1958, ISSN: 1532-4435.

## DH Projects

- Abiven, Karine et al. (May 2022), “Vers une collection numérique des libelles de la Fronde ou comment relier des mazarinades”, *in: Le Verger*, “Circulation des écrits littéraires de la Première Modernité & Humanités numériques”, URL: <https://hal.science/hal-03790226>.
- Boillet, Mélodie et al. (2019), “HORAÉ: an annotated dataset of books of hours”, *in: Proceedings of the 5th International Workshop on Historical Document Imaging and Processing*, HIP '19, Sydney, NSW, Australia: Association for Computing Machinery, pp. 7–12, ISBN: 9781450376686, DOI: [10.1145/3352631.3352633](https://doi.org/10.1145/3352631.3352633), URL: <https://doi.org/10.1145/3352631.3352633>.
- Bourgeois, Nicolas et al. (June 2022), “Le projet AGODA. Annoter et publier les débats parlementaires français de la fin du XIX e siècle : défis et solutions”, *in: Présentation des projets AGODA et Gallicorpora*, Bibliothèque nationale de France, Paris, France, URL: <https://hal.science/hal-03762957>.
- Chagué, Alix (May 2021a), “CREMMA : Une infrastructure mutualisée pour la reconnaissance d’écritures manuscrites et la patrimonialisation numérique”, *in: Sciences du patrimoine - sciences du texte. Confrontation des méthodes*, Ecole nationale des chartes, Paris, France, URL: <https://inria.hal.science/hal-03541887>.
- (Dec. 2021b), *Tapus Corpus*, version 1.0, URL: <https://github.com/HTR-United/tapuscorpus>.
- Chagué, Alix, Victoria Le Fournier, et al. (Oct. 2019), “Deux siècles de sources disparates sur l’industrie textile en France : comment automatiser les traitements d’un cor-

- pus non-uniforme ?”, in: *Colloque DHNord 2019 "Corpus et archives numériques"*, MESHs Lille Nord de France, Lille, France, URL: <https://inria.hal.science/hal-02448921>.
- Chagué, Alix, Lucas Terriel, and Laurent Romary (Nov. 2020), *Des images au texte : LECTAUREP, un projet de reconnaissance automatique d'écriture*, DHNord2020: The Measurement of Images: Computational Approaches in the History and Theory of the Arts, Poster, URL: <https://hal.science/hal-03008579>.
- Chiffolleau, Floriane and Anne Baillot (Apr. 2022), “Le projet DAHN : une pipeline pour l'édition numérique de documents d'archives”, working paper or preprint, URL: <https://hal.science/hal-03628094>.
- Gabay, Simon and Thibault Clérice (Jan. 2024), *CATMuS-Print [Large]*, Version Number: 2024-01-30, DOI: [10.5281/zenodo.10592716](https://doi.org/10.5281/zenodo.10592716), URL: <https://doi.org/10.5281/zenodo.10592716>.
- Gabay, Simon, Thibault Clérice, Pauline Jacsont, et al. (May 2024), “Reconnaissance des écritures dans les imprimés”, in: *Humanistica 2024*, OCR, Association francophone des humanités numériques, Meknès, Morocco, URL: <https://hal.science/hal-04557457>.
- Idmhand, Fatiha, Ioana Galleron, and Sabine Loudcher (Jan. 2023), “Consortium-HN ARIANE. Synthèse du projet scientifique”, working paper or preprint, URL: <https://shs.hal.science/halshs-04060828>.
- Kiessling, Benjamin (Sept. 2022), *Printed Arabic Base Model Trained on the OpenITI Corpus*, DOI: [10.5281/zenodo.7050296](https://doi.org/10.5281/zenodo.7050296), URL: <https://doi.org/10.5281/zenodo.7050296>.
- Mariotti, Viola (Oct. 2020), *Transcription automatique des feuillets du Manuscrit du Roi*, fr-FR, Billet, ISSN: 2740-773X, DOI: [10.58079/r8sy](https://doi.org/10.58079/r8sy), URL: <https://maritem.hypotheses.org/193> (visited on 07/10/2024).
- Massot, Marie-Laure, Arianna Sforzini, and Vincent Ventresque (Mar. 2019), “Transcribing Foucault's handwriting with Transkribus”, in: *Journal of Data Mining and Digital Humanities Atelier Digit\_Hum*, DOI: [10.46298/jdmdh.5043](https://doi.org/10.46298/jdmdh.5043), URL: <https://hal.science/hal-01913435>.
- Pinche, Ariane, Thibault Clérice, Alix Chagué, Jean-Baptiste Camps, Malamatenia Vlachou-Efstathiou, Matthias Gille Levenson, Olivier Brisville-Fertin, Federico Boschetti, Franz Fischer, Michael Gervers, Agnès Boutreux, Avery Manton, and Simon Gabay (July



- 2024), *CATMuS Medieval*, version 1.5.0, DOI: [10 . 5281 / zenodo . 12743230](https://doi.org/10.5281/zenodo.12743230), URL: <https://doi.org/10.5281/zenodo.12743230>.
- Pinche, Ariane, Thibault Clérice, Alix Chagué, Jean-Baptiste Camps, Malamatenia Vlachou-Efstathiou, Matthias Gille Levenson, Olivier Brisville-Fertin, Federico Boschetti, Franz Fischer, Michael Gervers, Agnès Boutreux, Avery Manton, Simon Gabay, et al. (Aug. 2024), “CATMuS-Medieval: Consistent Approaches to Transcribing Manuscripts”, in: *DH2024*, ADHO, Washington DC, United States, URL: <https://inria.hal.science/hal-04346939>.
- Pinche, Ariane, Simon Gabay, et al. (n.d.), *Données HTR incunables du 15e siècle*, URL: <https://github.com/Gallicorpora/HTR-incunable-15e-siecle>.
- Romanov, Maxim and Masoumeh Seydi (2019), *OpenITI: a Machine-Readable Corpus of Islamicate Texts (Version 2019.1.1)*, English.
- Sagot, Benoît et al. (Dec. 2022), “Gallic(orpor)a : Extraction, annotation et diffusion de l’information textuelle et visuelle en diachronie longue”, in: *DataLab de la BnF : Restitution des travaux 2022*, DataLab de la BnF, Paris, France, URL: <https://hal.science/hal-03930542>.
- Scheithauer, Hugo, Sarah Bénière, and Laurent Romary (Aug. 2024), “Automatic restructuring of auction sales catalogs layout and content”, in: *DH2024 - Reinvention and Responsibility*, Alliance of Digital Humanities Organizations, Washington DC, United States, URL: <https://hal.science/hal-04547239>.

## Digital Scholarly Editing

- Bénière, Sarah, Floriane Chiffolleau, and Laurent Romary (May 2024), “TEI Specifications for a Sustainable Management of Digitized Holocaust Testimonies”, in: *First Workshop on Holocaust Testimonies as Language Resources (HTRes) @LREC-COLING 2024*, ELRA Language Resources Association (ELRA) and International Committee on Computational Linguistics (ICCL), Turin, Italy, URL: <https://hal.science/hal-04538552>.
- Bénière, Sarah, Floriane Chiffolleau, and Hugo Scheithauer (June 2024), “Streamlining the Creation of Holocaust-related Digital Editions with Automatic Tools”, in: *EHRI Academic Conference - Researching the Holocaust in the Digital Age*, EHRI-3, Warsaw, Poland, URL: <https://inria.hal.science/hal-04594190>.

- Burnard, Lou (July 2019), “What is TEI Conformance, and Why Should You Care?”, en, in: *Journal of the Text Encoding Initiative Issue 12*, ISSN: 2162-5603, DOI: [10.4000/jtei.1777](https://doi.org/10.4000/jtei.1777), URL: <http://journals.openedition.org/jtei/1777> (visited on 01/09/2023).
- Chiffolleau, Floriane (June 2020a), *Difficulties in creating the transcription model*, Billet, DOI: [10.58079/nmyr](https://doi.org/10.58079/nmyr), URL: <https://digitalintellectuals.hypotheses.org/3812> (visited on 05/28/2024).
- (Mar. 2020b), *Encoding an XML Tree model for my corpus*, Billet, DOI: [10.58079/nmyf](https://doi.org/10.58079/nmyf), URL: <https://digitalintellectuals.hypotheses.org/3360> (visited on 05/28/2024).
- (Aug. 2020c), *Encoding the corpus*, Billet, DOI: [10.58079/nmyu](https://doi.org/10.58079/nmyu), URL: <https://digitalintellectuals.hypotheses.org/3891> (visited on 05/28/2024).
- (July 2020d), *How to produce a model for the segmentation*, Billet, DOI: [10.58079/nmys](https://doi.org/10.58079/nmys), URL: <https://digitalintellectuals.hypotheses.org/3844> (visited on 05/28/2024).
- (Dec. 2020e), *Publication of my digital edition – Working with TEI Publisher*, Billet, DOI: [10.58079/nmyw](https://doi.org/10.58079/nmyw), URL: <https://digitalintellectuals.hypotheses.org/3912> (visited on 05/28/2024).
- (July 2020f), *Transcribing the corpus*, Billet, DOI: [10.58079/nmyt](https://doi.org/10.58079/nmyt), URL: <https://digitalintellectuals.hypotheses.org/3872> (visited on 05/28/2024).
- (Sept. 2021a), *Availability and high quality: distributing the facsimile with NAKALA*, Billet, DOI: [10.58079/nmz9](https://doi.org/10.58079/nmz9), URL: <https://digitalintellectuals.hypotheses.org/4294> (visited on 05/28/2024).
- (Oct. 2021b), “Keeping it open: a TEI-based publication pipeline for historical documents”, working paper or preprint, URL: <https://hal.science/hal-04357295>.
- (June 2021c), *Publication of my digital edition – Developing my TEI Publisher application*, Billet, DOI: [10.58079/nmz3](https://doi.org/10.58079/nmz3), URL: <https://digitalintellectuals.hypotheses.org/4173> (visited on 05/28/2024).
- (Dec. 2021d), *Publication of my digital edition – Online launch of the TEI Publisher application*, Billet, DOI: [10.58079/nmzd](https://doi.org/10.58079/nmzd), URL: <https://digitalintellectuals.hypotheses.org/4399> (visited on 05/28/2024).
- (Mar. 2022), *Recognizing and encoding the corpus’ named entities*, Billet, DOI: [10.58079/nmzf](https://doi.org/10.58079/nmzf), URL: <https://digitalintellectuals.hypotheses.org/4470> (visited on 05/28/2024).

- Chiffolleau, Floriane and Hugo Scheithauer (Mar. 2024), “Leveraging EHRI Online Editions for training automated edition tools”, in: *EHRI Workshop Natural Language Processing Meets Holocaust Archives*, EHRI-3, Prague, Czech Republic, URL: <https://inria.hal.science/hal-04594084>.
- Driscoll, Matthew James and Elena Pierazzo, eds. (2016), *Digital Scholarly Editing : Theories and Practices*, en, Digital Humanities Series, Cambridge: Open Book Publishers, ISBN: 978-2-8218-8400-7, URL: <https://books.openedition.org/obp/3381> (visited on 05/29/2024).
- Pierazzo, Elena (July 2019), “What future for digital scholarly editions? From Haute Couture to Prêt-à-Porter”, en, in: *International Journal of Digital Humanities* 1.2, pp. 209–220, ISSN: 2524-7832, 2524-7840, DOI: [10.1007/s42803-019-00019-3](https://doi.org/10.1007/s42803-019-00019-3), URL: <http://link.springer.com/10.1007/s42803-019-00019-3> (visited on 04/20/2023).
- Turska, Magdalena, James Cummings, and Sebastian Rahtz (Sept. 2016), “Challenging the Myth of Presentation in Digital Editions”, en, in: *Journal of the Text Encoding Initiative Issue 9*, ISSN: 2162-5603, DOI: [10.4000/jtei.1453](https://doi.org/10.4000/jtei.1453), URL: <https://journals.openedition.org/jtei/1453> (visited on 05/18/2022).

## Ground Truth

- Chagué, Alix and Thibault Clérice (July 2023), “‘I’m here to fight for ground truth’: HTR-United, a solution towards a common for HTR training data”, in: *Digital Humanities 2023: Collaboration as Opportunity*, Alliance of Digital Humanities Organizations and University of Graz, Graz, Austria, URL: <https://inria.hal.science/hal-04094233>.
- Chagué, Alix, Thibault Clérice, and Laurent Romary (Nov. 2021), “HTR-United : Mutualisons la vérité de terrain !”, in: *DHNord2021 - Publier, partager, réutiliser les données de la recherche : les data papers et leurs enjeux*, MESHs, URL: <https://hal.archives-ouvertes.fr/hal-03398740> (visited on 05/03/2022).
- Gabay, Simon, Thibault Clérice, and Christian Reul (May 2020), “OCR17: Ground Truth and Models for 17th c. French Prints (and hopefully more)”, URL: <https://hal.archives-ouvertes.fr/hal-02577236> (visited on 05/03/2022).

- Gatos, Basilis et al. (Apr. 2014), “Ground-Truth Production in the Transcriptorium Project”, *in: 2014 11th IAPR International Workshop on Document Analysis Systems*, pp. 237–241, DOI: [10.1109/DAS.2014.23](https://doi.org/10.1109/DAS.2014.23).
- Springmann, Uwe et al. (Sept. 2018), *Ground Truth for training OCR engines on historical documents in German Fraktur and Early Modern Latin*, DOI: [10.48550/arXiv.1809.05501](https://doi.org/10.48550/arXiv.1809.05501), URL: <http://arxiv.org/abs/1809.05501> (visited on 10/24/2022).
- Ströbel, Phillip Benjamin, Simon Clematide, and Martin Volk (May 2020), “How Much Data Do You Need? About the Creation of a Ground Truth for Black Letter and the Effectiveness of Neural OCR”, English, *in: Proceedings of the Twelfth Language Resources and Evaluation Conference*, European Language Resources Association, pp. 3551–3559, ISBN: 979-10-95546-34-4, URL: <https://aclanthology.org/2020.lrec-1.436> (visited on 10/24/2022).

## Historical Background

- Audoin-Rouzeau, Stéphane and Annette Becker (2013), “Entrer dans la guerre”, français, *in: La Grande Guerre: 1914-1918*, Gallimard, chap. 1.
- Barcelo, Laurent and Élisabeth Préfacier du Réau (1995), *Paul d’Estournelles de Constant (Prix Nobel de la Paix 1909): l’expression d’une idée européenne*, français, Paris, France: Éd. l’Harmattan.
- Estournelles de Constant, Paul-Henri-Benjamin d’ and Nicholas Murray Butler (2018), *En guerre pour la paix: correspondance Paul d’Estournelles de Constant-Nicholas Murray Butler*, français, ed. by Stéphane Tison, trans. by Nadine Lange-Akhund, Alma éditeur.
- Hilberg, Raul et al. (2006a), “Conséquences”, français, *in: La destruction des Juifs d’Europe*, Paris, France: Gallimard, chap. XI.
- (2006b), “Les antécédents”, français, *in: La destruction des Juifs d’Europe*, Paris, France: Gallimard, chap. II.
- (2006c), “Les structures de la destruction”, français, *in: La destruction des Juifs d’Europe*, Paris, France: Gallimard, chap. III.
- Wieviorka, Olivier (2023), “Les causes de la Seconde Guerre mondiale”, français, *in: Histoire totale de la Seconde Guerre Mondiale*, Perrin, chap. 1.

## Languages

- Bagou, Odile and Ulrich Frauenfelder (n.d.), “SEGMENTATION, psycholinguistique”, in: *Encyclopedia Universalis* (), Consulted on July 2, 2024. Available online: <https://www.universalis.fr/encyclopedie/segmentation-psycholinguistique/>.
- Bakró-Nagy, Marianne (2012), “The Uralic Languages”, eng, in: *Revue belge de Philologie et d’Histoire* 90.3, pp. 1001–1027, DOI: 10.3406/rbph.2012.8272, URL: [https://www.persee.fr/doc/rbph\\_0035-0818\\_2012\\_num\\_90\\_3\\_8272](https://www.persee.fr/doc/rbph_0035-0818_2012_num_90_3_8272) (visited on 06/06/2024).
- Daniels, P. and W. Bright (2010), *The World’s Writing Systems*, Oxford University Press, Incorporated, ISBN: 9780195386929.
- Kapović, Mate, ed. (2017), *The Indo-European languages*, en, Second edition, Routledge language family series, London New York: Routledge, Taylor & Francis Group.
- Wells, J.C. (Feb. 2000), “Orthographic diacritics and multilingual computing”, in: *Language Problems & Language Planning* 24, pp. 249–272, DOI: 10.1075/lplp.24.3.04wel.

## Model Evaluation

- Haverals, Wouter (2023), *CERberus: guardian against character errors*, version 1.0, URL: <https://github.com/WHaverals/CERberus>.
- Neudecker, Clemens et al. (2021), “A survey of OCR evaluation tools and metrics”, in: *Proceedings of the 6th International Workshop on Historical Document Imaging and Processing*, HIP ’21, Lausanne, Switzerland: Association for Computing Machinery, pp. 13–18, ISBN: 9781450386906, DOI: 10.1145/3476887.3476888, URL: <https://doi.org/10.1145/3476887.3476888>.
- Terriel, Lucas (Dec. 2021), “Atelier : Production d’un modèle affiné de reconnaissance d’écriture manuscrite avec eScriptorium et évaluation de ses performances. Évaluer son modèle HTR/OCR avec KaMI (Kraken as Model Inspector)”, in: *Les Futurs Fantastiques - 3e Conférence Internationale sur l’Intelligence Artificielle appliquée aux Bibliothèques, Archives et Musées*, AI4LAM and Bibliothèque nationale de France, URL: <https://hal.science/hal-03495762>.

## Natural Language Processing

- Denham, K.E. and A.C. Lobeck (2013), *Linguistics for Everyone: An Introduction*, Wadsworth/Cengage Learning.
- Eisenstein, J. (2019), *Introduction to Natural Language Processing*, Adaptive Computation and Machine Learning series, MIT Press, ISBN: 978-0-262-04284-0.
- Jurafsky, Daniel and James Martin (Feb. 2008), *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, vol. 2, p. 577.
- MacKay, David J. C. (2002), *Information Theory, Inference & Learning Algorithms*, USA: Cambridge University Press, p. 640.
- Morris, Andrew, Viktoria Maier, and Phil Green (Oct. 2004), “From WER and RIL to MER and WIL: improved evaluation measures for connected speech recognition”, *in*: p. 5, DOI: [10.21437/Interspeech.2004-668](https://doi.org/10.21437/Interspeech.2004-668).
- Vasiliev, Yuli (Apr. 2020), *Natural Language Processing with Python and spaCy: A Practical Introduction*, en, No Starch Press, ISBN: 978-1-71850-053-2.

## Open Science

- Galleron, Ioana and Fatiha Idmhand (May 2021), “FAIRiser des données: état des lieux, barrières et choix. Une réflexion à partir des données des corpus d’auteurs”, *in*: *Colloque Humanistica 2021 - 10-12 mai 2021 Rennes (France)*, Association francophone des humanités numériques, Rennes, France, URL: <https://shs.hal.science/halshs-03224294>.
- Giglia, E. (Jan. 2019), “OPERAS: Bringing the long tail of social sciences and humanities into open science”, *in*: *JLIS.it* 10, pp. 140–156, DOI: [10.4403/jlis.it-12523](https://doi.org/10.4403/jlis.it-12523).
- Shafranovich, Yakov (Oct. 2005), *Common Format and MIME Type for Comma-Separated Values (CSV) Files*, RFC 4180, DOI: [10.17487/RFC4180](https://doi.org/10.17487/RFC4180), URL: <https://www.rfc-editor.org/info/rfc4180>.
- Wilkinson, Mark D. et al. (Mar. 2016), “The FAIR Guiding Principles for scientific data management and stewardship”, *in*: *Scientific Data* 3.1, p. 160018, ISSN: 2052-4463, DOI: [10.1038/sdata.2016.18](https://doi.org/10.1038/sdata.2016.18), URL: <https://doi.org/10.1038/sdata.2016.18>.

## Post-ATR Correction

- Chiron, Guillaume et al. (Nov. 2017), “ICDAR2017 Competition on Post-OCR Text Correction”, in: *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, Kyoto, France: IEEE, pp. 1423–1428, DOI: [10.1109/icdar.2017.232](https://doi.org/10.1109/icdar.2017.232), URL: <https://hal.science/hal-03025499>.
- Lin, Junxia and Johannes Ledolter (2021), “A Simple and Practical Approach to Improve Misspellings in OCR Text”, en, *in*.
- Mollá-Aliod, Diego and Steve Cassidy (Dec. 2017), “Overview of the 2017 ALTA Shared Task: Correcting OCR Errors”, in: *Proceedings of the Australasian Language Technology Association Workshop 2017*, ed. by Jojo Sze-Meng Wong and Gholamreza Haffari, Brisbane, Australia, pp. 115–118, URL: <https://aclanthology.org/U17-1014>.
- Nguyen, Thi Tuyet Hai et al. (July 2021), “Survey of Post-OCR Processing Approaches”, in: *ACM Comput. Surv.* 54.6, ISSN: 0360-0300, DOI: [10.1145/3453476](https://doi.org/10.1145/3453476), URL: <https://doi.org/10.1145/3453476>.
- Nguyen, Thi-Tuyet-Hai et al. (June 2019), “Deep Statistical Analysis of OCR Errors for Effective Post-OCR Processing”, en, in: *2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, IEEE, pp. 29–38, ISBN: 978-1-72811-547-4, DOI: [10.1109/JCDL.2019.00015](https://doi.org/10.1109/JCDL.2019.00015), URL: <https://ieeexplore.ieee.org/document/8791206/> (visited on 04/18/2023).
- Rigaud, Christophe et al. (Sept. 2019), “ICDAR 2019 Competition on Post-OCR Text Correction”, in: *15th International Conference on Document Analysis and Recognition*, Sydney, Australia, pp. 1588–1593, URL: <https://hal.science/hal-02304334>.

## Segmentation

- Clérice, Thibault (Dec. 2023), “You Actually Look Twice At it (YALTAi): using an object detection approach instead of region segmentation within the Kraken engine”, en, in: *Journal of Data Mining & Digital Humanities Historical Documents and Automatic Text Recognition*, p. 9806, ISSN: 2416-5999, DOI: [10.46298/jdmdh.9806](https://doi.org/10.46298/jdmdh.9806), URL: <https://jdmdh.episciences.org/9806> (visited on 06/11/2024).
- Clérice, Thibault et al. (Aug. 2024), “Layout Analysis Dataset with SegmOnto”, in: *DH2024 - Annual conference of the Alliance of Digital Humanities Organizations*, ADHO, Washington DC, United States, URL: <https://inria.hal.science/hal-04513725>.

- Coquenot, Denis, Clément Chatelain, and Thierry Paquet (Sept. 2021), “Handwritten text recognition: from isolated text lines to whole documents”, in: *ORASIS 2021*, Centre National de la Recherche Scientifique [CNRS], Saint Ferréol, France, URL: <https://hal.science/hal-03339648>.
- Gabay, Simon and Ariane Pinche (2021), “SegmOnto: common vocabulary and practices for analysing the layout of manuscripts (and more)”, in: URL: <https://hal.science/hal-03336528/>.
- Romanello, Matteo and Sven Najem-Meyer (Nov. 2022), *Layout Ground Truth for Historical Commentaries*, en, DOI: [10.5281/ZENODO.7271729](https://doi.org/10.5281/ZENODO.7271729), URL: <https://zenodo.org/record/7271729> (visited on 06/12/2024).
- Staab, Steffen and Rudi Studer, eds. (2009), *Handbook on Ontologies*, en, Berlin, Heidelberg: Springer Berlin Heidelberg, DOI: [10.1007/978-3-540-92673-3](https://doi.org/10.1007/978-3-540-92673-3), (visited on 08/30/2024).
- Tensmeyer, Chris and Curtis Wigington (Sept. 2019), “Training Full-Page Handwritten Text Recognition Models without Annotated Line Breaks”, en, in: *2019 International Conference on Document Analysis and Recognition (ICDAR)*, Sydney, Australia: IEEE, pp. 1–8, ISBN: 978-1-72813-014-9, DOI: [10.1109/ICDAR.2019.00011](https://doi.org/10.1109/ICDAR.2019.00011), URL: <https://ieeexplore.ieee.org/document/8978164/> (visited on 06/11/2024).





# List of Figures

1.1	Steps of the OCR process . . . . .	24
1.2	Printed edition of <i>Notre-Dame de Paris</i> by Victor Hugo . . . . .	26
1.3	Excerpt from a medieval manuscript from the 14th century . . . . .	29
1.4	Excerpt from an incunabulum from the 15th century . . . . .	29
1.5	Excerpt from a letter from Auguste Delâtre to Edouard Foley . . . . .	29
1.6	Excerpt from a Latin medieval manuscript from the 16th century . . . . .	29
1.7	Excerpt from a Parisian notary’s registries of deeds . . . . .	29
1.8	Schema of a neural network architecture . . . . .	31
1.9	Schema of a model training . . . . .	32
2.1	An EHRI document annotated at region-level with the SegmOnto vocabulary . .	43
2.2	Entries of the PhD datasets in the HTR-United’s catalog . . . . .	46
3.1	Letter no552 – June 24th, 1919 . . . . .	55
4.1	Word cloud of the set Ground Truth . . . . .	66
4.2	Word cloud of the set War (big version) . . . . .	66
4.3	Common between the set Other and set War . . . . .	72
4.4	Part-of-speech division of common elements between the set Other and set War	75
5.1	Examples of Levenshtein distance of 1 . . . . .	79
5.2	Examples of insertions (I), deletions (D), substitutions (S) and their Levenshtein distance . . . . .	79
5.3	Equations for the WER and Wacc . . . . .	79
5.4	Example of a WER and CER calculation (GT = Ground Truth; P = Prediction)	79
5.5	Versus text for the page 1358-4 . . . . .	90
5.6	Versus text from the model Other applied to the page 948-1 . . . . .	93
6.1	Bar chart representation of the distribution of the Initials n-grams in the set War	106

---

7.1	Comparison of the model War (MW), Other (MO), and Ground Truth (MGT) to the ground truth of the 21st line of the page 1 of the letter 1367 . . . . .	117
7.2	Examples of blue cells with insertion(s) (top) or deletion(s) (bottom) in the prediction . . . . .	118
7.3	Examples of grey cells for tetragrams (top), trigrams (middle) and bigrams (bottom) . . . . .	119
7.4	Example of a line from the table of bigrams where all the predictions had an error	122
7.5	Example of a line from the table of bigrams where one prediction has not the same length (model War), one has several erroneous n-grams (model Other) and one has the right prediction (model Ground Truth) . . . . .	122
7.6	Example of a line from the table of trigrams where one prediction has an erroneous n-gram (model War), one has the right prediction (model Other) and one has not the same length (model Ground Truth) . . . . .	122
7.7	Example of a line from the table of tetragrams where one prediction has not the same length (model War), one has the right prediction (model Other) and one has an erroneous n-gram (model Ground Truth) . . . . .	122
7.8	Results by numbers from the Token Error Analysis tables . . . . .	123
7.9	Results by percentages from the Token Error Analysis tables . . . . .	123
8.1	Images from the EHRI dataset . . . . .	150
9.1	Excerpt from the general distribution of the alphabet . . . . .	153
9.2	The character <i>e</i> in each table . . . . .	155
9.3	Difference of percentage for the characters <i>ž</i> and <i>Ů</i> . . . . .	157
9.4	Common n-grams between two languages . . . . .	162
10.1	Clean and clear images . . . . .	170
10.2	Noisy images . . . . .	170
10.3	Gibberish substitutions in Danish . . . . .	174
10.4	Prediction errors of the model on diacritics . . . . .	175
10.5	Examples of blue cells with deletion(s) (English) or insertion(s) (Danish) in the prediction . . . . .	177
10.6	Examples of grey cells for bigrams (Italian), trigrams (Slovak) and tetragrams (French) . . . . .	178
10.7	Example of a line, in Danish, from the table of bigrams, where there were two erroneous n-grams but the occurrences results differ . . . . .	178
10.8	Results from the Token Error Analysis tables - General Information . . . . .	179
10.9	Results from the Token Error Analysis tables - By Language . . . . .	180

## List of Figures

---

10.10 Two ways to handwrite a word in a same document . . . . .	200
E.1 Bigrams Page 1 (Token Error Analysis) . . . . .	236
E.2 Bigrams Page 2 (Token Error Analysis) . . . . .	237
E.3 Bigrams Page 3 (Token Error Analysis) . . . . .	238
E.4 Bigrams Page 4 (Token Error Analysis) . . . . .	239
E.5 Bigrams Page 5 (Token Error Analysis) . . . . .	240
E.6 Bigrams Page 6 (Token Error Analysis) . . . . .	241
E.7 Bigrams Page 7 (Token Error Analysis) . . . . .	242
E.8 Bigrams Page 8 (Token Error Analysis) . . . . .	243
E.9 Bigrams Page 9 (Token Error Analysis) . . . . .	244
E.10 Bigrams Page 10 (Token Error Analysis) . . . . .	245
E.11 Bigrams Page 11 (Token Error Analysis) . . . . .	246
E.12 Trigrams Page 1 (Token Error Analysis) . . . . .	247
E.13 Trigrams Page 2 (Token Error Analysis) . . . . .	248
E.14 Trigrams Page 3 (Token Error Analysis) . . . . .	249
E.15 Trigrams Page 4 (Token Error Analysis) . . . . .	250
E.16 Trigrams Page 5 (Token Error Analysis) . . . . .	251
E.17 Trigrams Page 6 (Token Error Analysis) . . . . .	252
E.18 Trigrams Page 7 (Token Error Analysis) . . . . .	253
E.19 Trigrams Page 8 (Token Error Analysis) . . . . .	254
E.20 Trigrams Page 9 (Token Error Analysis) . . . . .	255
E.21 Trigrams Page 10 (Token Error Analysis) . . . . .	256
E.22 Trigrams Page 11 (Token Error Analysis) . . . . .	257
E.23 Tetragrams Page 1 (Token Error Analysis) . . . . .	258
E.24 Tetragrams Page 2 (Token Error Analysis) . . . . .	259
E.25 Tetragrams Page 3 (Token Error Analysis) . . . . .	260
E.26 Tetragrams Page 4 (Token Error Analysis) . . . . .	261
E.27 Tetragrams Page 5 (Token Error Analysis) . . . . .	262
E.28 Tetragrams Page 6 (Token Error Analysis) . . . . .	263
E.29 Tetragrams Page 7 (Token Error Analysis) . . . . .	264
E.30 Tetragrams Page 8 (Token Error Analysis) . . . . .	265
E.31 Tetragrams Page 9 (Token Error Analysis) . . . . .	266
E.32 Tetragrams Page 10 (Token Error Analysis) . . . . .	267
E.33 Tetragrams Page 11 (Token Error Analysis) . . . . .	268
F.1 Distribution of the alphabet by character Page 1 (Multilingual model) . . . . .	270

---

F.2	Distribution of the alphabet by character Page 2 (Multilingual model) . . . . .	271
F.3	Distribution of the alphabet by character Page 3 (Multilingual model) . . . . .	272
F.4	Distribution of the alphabet by character Page 4 (Multilingual model) . . . . .	272
F.5	Distribution of the alphabet by language Page 1 (Multilingual model) . . . . .	273
F.6	Distribution of the alphabet by language Page 2 (Multilingual model) . . . . .	273
F.7	Distribution of the alphabet by language Page 3 (Multilingual model) . . . . .	274
F.8	Distribution of the alphabet by language Page 4 (Multilingual model) . . . . .	274
F.9	General distribution of the alphabet Page 1 (Multilingual model) . . . . .	275
F.10	General distribution of the alphabet Page 2 (Multilingual model) . . . . .	276
F.11	General distribution of the alphabet Page 3 (Multilingual model) . . . . .	277
I.1	Bigrams Page 1 (Multilingual model) . . . . .	290
I.2	Bigrams Page 2 (Multilingual model) . . . . .	291
I.3	Bigrams Page 3 (Multilingual model) . . . . .	292
I.4	Bigrams Page 4 (Multilingual model) . . . . .	293
I.5	Bigrams Page 5 (Multilingual model) . . . . .	294
I.6	Bigrams Page 6 (Multilingual model) . . . . .	295
I.7	Bigrams Page 7 (Multilingual model) . . . . .	296
I.8	Trigrams Page 1 (Multilingual model) . . . . .	297
I.9	Trigrams Page 2 (Multilingual model) . . . . .	298
I.10	Trigrams Page 3 (Multilingual model) . . . . .	299
I.11	Trigrams Page 4 (Multilingual model) . . . . .	300
I.12	Trigrams Page 5 (Multilingual model) . . . . .	301
I.13	Trigrams Page 6 (Multilingual model) . . . . .	302
I.14	Trigrams Page 7 (Multilingual model) . . . . .	303
I.15	Tetragrams Page 1 (Multilingual model) . . . . .	304
I.16	Tetragrams Page 2 (Multilingual model) . . . . .	305
I.17	Tetragrams Page 3 (Multilingual model) . . . . .	306
I.18	Tetragrams Page 4 (Multilingual model) . . . . .	307
I.19	Tetragrams Page 5 (Multilingual model) . . . . .	308
I.20	Tetragrams Page 6 (Multilingual model) . . . . .	309
I.21	Tetragrams Page 7 (Multilingual model) . . . . .	310
J.1	Groundtruth . . . . .	312
J.2	Complete list of words . . . . .	313
J.3	List of words from the set Other . . . . .	314
J.4	List of words from the set War . . . . .	314

## List of Figures

---

K.1	Common between the sets Other and War . . . . .	315
K.2	Unique to the set Other . . . . .	316
K.3	Unique to the set War . . . . .	316
L.1	Part-of-speech division from the set Other . . . . .	318
L.2	Part-of-speech division from the set War . . . . .	318
L.3	Part-of-speech division of unique elements from the set Other (no commonality with the set War) . . . . .	319
L.4	Part-of-speech division of unique elements from the set War (no commonality with the set Other) . . . . .	319
L.5	Part-of-speech division of common elements between the sets Other and War . .	320
M.1	All caps - set Other . . . . .	321
M.2	All caps - set War . . . . .	322
M.3	All caps - set Ground Truth . . . . .	322
M.4	Initials - set Other . . . . .	323
M.5	Initials - set War . . . . .	323
M.6	Initials - set Ground Truth . . . . .	324
M.7	Lowercases - set Other . . . . .	324
M.8	Lowercases - set War . . . . .	325
M.9	Lowercases - set Ground Truth . . . . .	325



# List of Tables

3.1	Composition of the set War . . . . .	59
3.2	Composition of the set Other . . . . .	59
3.3	Composition of the test set . . . . .	60
4.2	14 part-of-speech classes of the sets . . . . .	73
5.1	Metrics for the models applied to the sets (S = Set; M = Model; O = Other; W = War; R = Retrained) . . . . .	85
5.2	Metrics for the model Other applied to the set Other . . . . .	86
5.3	Metrics for the model Other applied to the set War . . . . .	86
5.4	Metrics for the model War applied to the set Other . . . . .	87
5.5	Metrics for the model War applied to the set War . . . . .	87
5.6	Metrics for the model War Retrained applied to the set Other . . . . .	88
5.7	Metrics for the model War Retrained applied to the set War . . . . .	88
6.1	Examples of n-gram divisions according to the tokens given . . . . .	100
6.2	Occurrences of the raw lists created from the sets . . . . .	103
6.3	Occurrences of the raw dictionaries created from the sets . . . . .	103
6.4	Regular expressions to clean the dictionaries of units of trigrams . . . . .	103
6.5	Table of the distribution of the Initials n-grams in the set War . . . . .	106
6.7	Distribution of the n-grams in All Caps . . . . .	109
6.9	Distribution of the n-grams in Initials . . . . .	109
6.11	Distribution of the n-grams in Lowercases . . . . .	111
6.12	Distribution of the least and most popular n-grams . . . . .	111
8.1	Distribution of the EHRI training data . . . . .	147
9.1	Occurrences of the raw lists created from the sets . . . . .	159
9.2	Occurrences of the raw dictionaries created from the sets . . . . .	159
10.1	Distribution of the EHRI test set . . . . .	168



---

10.3	Comparative results for the EHRI NFC model with and without punctuation (*)	173
10.4	General distribution of unknown occurrences in the tables . . . . .	186
10.5	Distribution by language of unknown occurrences in the tables . . . . .	187
A.1	Table for the words present in the sets . . . . .	204
A.3	Table for the words uniquely present in each set and in common . . . . .	204
A.5	Table of the POS for the words present in the sets (tokens) . . . . .	205
A.7	Table of the POS for the words present in the sets (lemmas) . . . . .	205
A.9	Table of the POS for the words uniquely present in each set and in common (tokens) . . . . .	205
A.11	Table of the POS for the words uniquely present in each set and in common (lemmas) . . . . .	205
B.1	Metrics for the models applied to the letter 607 page 3 . . . . .	206
B.3	Metrics for the models applied to the letter 607 page 17 . . . . .	207
B.5	Metrics for the models applied to the letter 722 page 1 . . . . .	207
B.7	Metrics for the models applied to the letter 1170 page 3 . . . . .	208
B.9	Metrics for the models applied to the letter 1358 page 4 . . . . .	208
B.11	Metrics for the models applied to the letter 678 page 1 . . . . .	209
B.13	Metrics for the models applied to the letter 844 page 1 . . . . .	209
B.15	Metrics for the models applied to the letter 948 page 1 . . . . .	210
B.17	Metrics for the models applied to the letter 1000 page 3 . . . . .	210
B.19	Metrics for the models applied to the letter 1367 page 1 . . . . .	211
C.1	All tokens . . . . .	212
C.3	All tokens (by type) (numbers) . . . . .	213
C.5	All tokens (by type) (percentages) . . . . .	213
C.7	Tokens from set Other (All Caps) . . . . .	213
C.9	Tokens from set Other (Initials) . . . . .	213
C.11	Tokens from set Other (Lowercases) . . . . .	214
C.13	Tokens from set Other (11 and more) . . . . .	214
C.15	Tokens from set Other (Only 1) . . . . .	214
C.17	Tokens from set War (All Caps) . . . . .	214
C.19	Tokens from set War (Initials) . . . . .	214
C.21	Tokens from set War (Lowercases) . . . . .	215
C.23	Tokens from set War (11 and more) . . . . .	215
C.25	Tokens from set War (Only 1) . . . . .	215
C.27	Tokens from set Ground Truth (All Caps) . . . . .	215

## List of Tables

---

C.29	Tokens from set Ground Truth (Initials) . . . . .	215
C.31	Tokens from set Ground Truth (Lowercases) . . . . .	216
C.33	Tokens from set Ground Truth (11 and more) . . . . .	216
C.35	Tokens from set Ground Truth (Only 1) . . . . .	216
C.37	Most popular tokens (11 and more) (numbers) . . . . .	217
C.44	Most popular tokens (11 and more) (percentages) . . . . .	217
C.51	Least popular tokens (Only 1 occurrence) (numbers) . . . . .	218
C.58	Least popular tokens (Only 1 occurrence) (percentages) . . . . .	218
D.1	Prediction errors of the letter 607 page 3 (set Other) . . . . .	220
D.2	Prediction errors of the letter 607 page 17 (set Other) . . . . .	222
D.3	Prediction errors of the letter 722 page 1 (set Other) . . . . .	224
D.4	Prediction errors of the letter 1170 page 3 (set Other) . . . . .	225
D.5	Prediction errors of the letter 1358 page 4 (set Other) . . . . .	227
D.6	Prediction errors of the letter 678 page 1 (set War) . . . . .	228
D.7	Prediction errors of the letter 844 page 1 (set War) . . . . .	229
D.8	Prediction errors of the letter 948 page 1 (set War) . . . . .	231
D.9	Prediction errors of the letter 1000 page 3 (set War) . . . . .	233
D.10	Prediction errors of the letter 1367 page 1 (set War) . . . . .	234
G.1	Comparative results for the EHRI NFC model (normal) . . . . .	279
G.3	Comparative results for the EHRI NFC model (no punctuation) . . . . .	280
H.1	Prediction errors from the model EHRI NFC . . . . .	288
N.1	Composition of the set War . . . . .	328
N.2	Composition of the set Other . . . . .	329
N.3	Composition of the test set . . . . .	329
N.4	Constitution of the Ground Truth . . . . .	330
O.1	Distribution of the EHRI training data . . . . .	332
O.2	Distribution of the EHRI test set . . . . .	332



# INDEX

## A

- Automatic Speech Recognition
  - (ASR) 27, 80, 83, 89
  - MER 80, 83
  - SER 80
- Automatic Text Recognition (ATR)
  - 15, 16, 18, 20, 21, 23, 26,
  - 30, 31, 33, 34, 37–39, 42,
  - 44, 45, 48, 54, 77, 78,
  - 80–82, 99, 146, 176, 201

## C

- Comparative analysis 20, 68, 74,  
82–84, 89, 97, 115, 165,  
171, 176, 182, 192

## D

- DAHN Project 11, 17, 53, 198
- deep learning 23, 30, 33, 35, 39, 41
- diacritics 83, 89, 148, 151, 155–157,  
166, 167, 171, 175, 176,  
187, 190–193, 196, 197
- digital scholarly edition 11–17, 36,  
53, 198

## E

- EHRI dataset 158–160, 165, 197,  
198
- EHRI Online Editions 41, 143, 144,  
146, 147, 149, 167, 168
- Begrenzte Flucht 144, 169

Diplomatic Reports 144, 169

Early Holocaust Testimonies  
144, 169

Nisko 144, 145, 169

EHRI Project 11, 17, 143, 144

error 16, 19, 26, 31, 38, 45, 47, 54,  
77, 78, 80, 82, 89, 92, 93,  
97, 115–121, 126–129, 132,  
133, 135–137, 165, 171,  
174–178, 180–184, 188–193

eScriptorium 35, 37, 38, 58, 82, 84,  
115, 176

## F

FAIR principles 12–14

France 13, 15, 51, 52, 58, 141, 143

## G

Germany 141, 142, 144

graphics

bar chart 105

frequency list 63–65, 67, 68

heatmap 152, 154, 155, 161,  
164

pie chart 71, 74, 76

word cloud 64, 65, 67–70, 74, 76

ground truth 15, 19, 20, 38, 40, 42,  
44, 45, 48, 54, 57, 58, 61,  
62, 67, 77, 81, 94, 116, 118,  
128, 129, 146, 148, 149,  
195, 196, 198

**H**

**Handwritten Text Recognition**  
 (HTR) 18, 19, 23, 27, 28,  
 35, 36, 39, 40, 42, 44, 82,  
 98, 199

**Holocaust** 11, 141–143, 145

**HTR-United** 38, 44, 45

**I**

**IIIF** 14, 15, 38

**J**

**Jews/Jewish population** 142–145

**K**

**KaMI app** 82–85, 89

**Kraken** 37, 39, 41, 81, 149, 197

**L**

**Language**

**Czech** 147, 148, 155, 156, 158,  
 160, 163, 164, 166, 167,  
 175, 191

**Danish** 147, 148, 156, 158, 160,  
 166–169, 174, 175, 182,  
 184, 185, 187–191

**Dutch** 70, 147, 167

**English** 11, 26, 54, 70, 144, 145,  
 147, 148, 156, 158, 160,  
 163, 166–169, 174, 175,  
 182, 184, 185, 187, 188,  
 196, 198

**French** 11, 42, 54, 68, 70, 89,  
 147, 167–169, 175, 182,  
 184, 186, 190, 191, 193, 198

**German** 11, 70, 144, 145, 147,  
 148, 154–156, 158, 160,

163, 166–168, 198

**Hungarian** 147, 148, 155, 156,  
 158, 160, 164, 166, 167

**Italian** 70, 147, 167–169, 175,  
 182, 185, 187, 191, 192,  
 197, 198

**Japanese** 146, 147

**Polish** 145, 147, 148, 155, 156,  
 158, 160, 163, 166, 167

**Slovak** 147, 148, 156, 158, 160,  
 163, 166–169, 174–176,  
 182, 184, 187, 189, 190, 196

**Yiddish** 11, 146, 147

**language** 11, 16, 20, 26, 28, 34, 36,  
 42, 44, 48, 63, 68, 70, 89,  
 97, 98, 102, 137, 141,  
 145–149, 151, 152,  
 154–158, 160, 161,  
 163–169, 171, 174–177,  
 179, 182–188, 192, 193,  
 195–198

**Lexicon analysis** 63, 67–69, 76

**M**

**machine learning** 28, 58

**metrics** 44, 48, 77, 78, 80, 81,  
 83–85, 89, 92, 115, 165,  
 171, 174–176

**CER** 78, 80, 81, 89, 91–93, 174

**error rate** 78, 81

**Hamming distance** 80, 83, 89,  
 118

**Levenshtein distance** 47, 78, 81,  
 83, 91

**WER** 78, 80, 81, 89, 92, 93,  
 174, 175

**Word accuracy (Wacc)** 78, 91,

- 92, 175
- Model**
- EHRI (MEHRI) 181, 198
- Ground Truth (MGT) 112, 113, 116, 117, 120, 121, 125, 127–130, 132, 133, 135, 136, 181, 192
- Other (MO) 58, 84, 85, 91, 92, 112, 116, 117, 120, 121, 125, 127–131, 133, 134, 136
- War (MW) 58, 84, 85, 91–93, 112, 116, 117, 120, 121, 125, 127–132, 134–136
- War Retrained (MWR) 58, 84, 85, 91–93
- model** 16, 17, 19, 20, 26, 28, 31, 35–38, 40, 42, 44, 45, 47, 48, 54, 57, 58, 61, 62, 67, 68, 70, 76–78, 80, 81, 83–85, 89, 91–93, 97–99, 105, 107, 110, 112, 113, 115–118, 120, 123–125, 127–133, 135–137, 146–149, 157, 165–169, 171, 175–177, 179, 181, 182, 184–187, 189–193, 195–201
- multilingual**
- dataset 145–148, 151, 157, 158, 164, 171, 187, 195, 196, 201
- documents 141, 198
- model 146, 149, 151, 165, 196–198, 201
- scale 176
- support 34
- training data 201
- multilingualism** 20, 139, 141, 151, 165, 197, 201
- N**
- n-gram** 20, 95, 97–100, 102, 104, 105, 107, 108, 110, 112, 115–121, 123–131, 133, 135–137, 139, 141, 149, 151, 158, 160, 161, 163–166, 176–180, 182, 183, 186, 188, 189, 195–201
- 2-gram/bigram** 98, 99, 101, 102, 104, 105, 107, 108, 110, 112, 119, 120, 123, 126–129, 132, 134–136, 158, 160, 163, 164, 177, 180–183, 185, 187–192, 195, 196, 199
- 3-gram/trigram** 99, 101, 102, 104, 105, 107, 108, 110, 112, 119, 120, 123, 126–129, 132, 133, 135, 136, 158, 160, 163, 164, 177, 180–183, 185, 187–190, 192, 195, 199
- 4-gram/tetragram** 99, 101, 102, 104, 105, 107, 108, 110, 112, 119, 121, 123, 126–130, 132, 133, 135–137, 158, 160, 163, 164, 177, 180, 181, 183, 185, 187–191, 195
- neural network** 16, 17, 19, 23, 26, 28, 30, 31, 33–35, 37, 40, 42, 45, 48, 61, 99, 195, 201
- Nicholas Murray Butler** 52–54, 67

**O**

open science 12, 13, 34, 35, 45  
 open source 13, 14, 33–35, 81, 144  
 Optical Character Recognition  
 (OCR) 18, 19, 23, 24, 26,  
 28, 33–35, 39, 42, 47, 81,  
 98, 199

**P**

Paul d'Estournelles de Constant 17,  
 19, 51–56, 61, 89, 97, 141,  
 158, 182  
 Post-OCR correction 14, 17, 19, 45,  
 47, 48, 54, 175  
 prediction 16, 19, 30, 38, 45, 47, 62,  
 68, 77, 80–84, 89, 91–93,  
 115–121, 124, 125, 127,  
 128, 130, 165, 171, 177,  
 178, 185–193  
 prediction error 45, 48, 91, 92, 97,  
 115, 117, 165, 166, 176  
 Python 68, 70, 101, 104, 117, 148,  
 158, 160

**R**

regular expression 16, 102

**S****Set**

Ground Truth (SGT) 63, 64, 67,  
 99, 101, 104, 107, 108, 110,  
 112, 196, 201  
 Other (SO) 58, 61, 63–65, 67,  
 69–71, 74, 76, 84, 85, 89,  
 91, 92, 99, 101, 104, 105,  
 107, 108, 110, 112, 196, 201

War (SW) 58, 63–65, 67,  
 69–71, 74, 76, 84, 85, 89,  
 91, 92, 99, 101, 104, 105,  
 107, 108, 110, 112, 196, 201  
 single-language 149, 156, 157,  
 195–197, 201

**T****text analysis**

character-level 25, 98, 151  
 infralexical level 95, 98, 104, 176  
 lemma 69–71, 74  
 lemmatization 69, 70  
 lexicon 20, 48, 49, 57, 58,  
 61–63, 67, 70, 76, 77, 84,  
 89, 91, 93, 97, 98, 100, 115,  
 137, 195, 196  
 part-of-speech (POS) tagging  
 71, 74  
 token 47, 69–71, 74, 100, 102,  
 105, 116–120, 123,  
 125–128, 133, 158, 161,  
 176–178, 180, 182, 183,  
 185, 190, 197, 199  
 tokenization 69, 70, 100  
 word-level 94, 97, 98, 137, 196

Text Encoding Initiative (TEI) 14,  
 18, 36

Token analysis 128, 151, 158, 160,  
 166, 197, 199, 201

Token error analysis 165, 176, 177,  
 181, 187, 196, 201

**training**

accuracy 19, 25, 26, 33, 35, 42,  
 45, 58, 61, 77, 81, 82, 92,  
 112, 115, 149, 168, 174,  
 175, 184, 191, 195–198, 201

model training 17, 23, 32, 45,  
112, 146, 148, 149, 167,  
184, 195, 198  
overfitting 30, 84, 91–93, 168  
training data 17, 19, 30, 32, 35,  
36, 39, 40, 48, 61, 84, 93,  
97–99, 107, 116, 117,  
119–121, 124, 126,  
131–133, 146, 149, 151,  
158, 164–168, 171,  
174–178, 181–183, 185,  
187–191, 195–199, 201  
transcription (of a document) 16,  
32, 42, 44, 45, 77, 78, 80,  
116–118, 120, 186–193,  
198, 201

## U

United States 52, 56, 141, 143, 145

## W

workflow 12–17, 35  
digitization 14, 17, 24  
encoding 14, 16, 36, 37, 62  
publication 14–16, 144, 198  
segmentation 14, 16, 24, 25, 36,  
38–41, 54, 116, 169  
transcription 14, 16, 23, 36,  
38–40, 44, 54, 62, 116, 149,  
171  
World War I (WWI) 11, 12, 51, 141  
World War II (WWII) 142, 145  
writing systems 145, 146  
abjad 145, 146  
abugida 145  
alphabetical 145, 146  
logographic 145, 146  
syllabary 145, 146





# Table of Contents

<b>Acknowledgement</b>	<b>3</b>
<b>List of Acronyms</b>	<b>9</b>
<b>Introduction</b>	<b>11</b>
A Workflow for Creating Digital Scholarly Editions . . . . .	12
What is a Digital Scholarly Edition? . . . . .	12
Open Science and FAIR Principles: The Heart of the Workflow . . . . .	12
Components of a Workflow . . . . .	14
The Importance of Implementing a Workflow . . . . .	15
Practical Applications of a Digital Scholarly Editions Workflow . . . . .	15
Automatic Text Recognition (ATR): A Core Component of This PhD Research . . . . .	16
Challenges in Transcription . . . . .	16
Datasets Used in Research Experiments . . . . .	17
ATR: An Overview of Key Concepts . . . . .	18
Typewritten Documents: An Ideal Resource For This PhD . . . . .	19
Problem Statement and Outline of the Thesis . . . . .	19
<b>I Automatic Text Recognition: An Evolving Technique in Need of Understanding</b>	<b>21</b>
<b>1 From OCR to ATR: Evolution of Transcription Techniques</b>	<b>23</b>
1.1 OCR: Working with Bounding Boxes and Characters . . . . .	23
1.1.1 The Development of OCR and Its Purposes . . . . .	23
1.1.2 How Does OCR Operate? . . . . .	24
1.2 ATR: The Emergence of Recognition via Neural Networks . . . . .	26
1.2.1 The Need for a New Recognition Method . . . . .	26
1.2.2 A Solution Found in the Use of Neural Networks . . . . .	28

<b>2</b>	<b>A New Horizon Opening with the Emergence of Deep Learning Approaches</b>	<b>33</b>
2.1	With Great Changes Come Great Software . . . . .	33
2.1.1	From Proprietary to Open Source, from Generic Algorithms to Neural Networks, from Printed to Handwritten . . . . .	33
2.1.2	Transkribus vs eScriptorium: The Battle of the HTR Software . . . . .	35
2.2	New Techniques, Emerging Concerns, Innovative Solutions . . . . .	39
2.2.1	Novel Approaches to Segmentation . . . . .	39
2.2.2	Creating Efficient and Sufficient Ground Truth: An Enigma . . . . .	42
2.2.3	Post-ATR Correction and Prediction Errors: What Does It Entail? . . . . .	45
<b>II</b>	<b>The (Null) Influence of the Lexicon</b>	<b>49</b>
<b>3</b>	<b>The Correspondence of Paul d’Estournelles de Constant: A Thematic Corpus to Lead Our Study</b>	<b>51</b>
3.1	History and Presentation of the Corpus . . . . .	51
3.1.1	Historical Background . . . . .	51
3.1.2	The Main Characters . . . . .	52
3.1.3	The Correspondence . . . . .	52
3.1.4	The Project . . . . .	53
3.1.5	Structure of the Document . . . . .	54
3.2	A Rich and Diverse Corpus . . . . .	55
3.2.1	A Large Source . . . . .	55
3.2.2	An Eclectic Collection . . . . .	56
3.3	Choosing the Experiment’s Test Set . . . . .	57
3.3.1	A Few Elements to Consider . . . . .	57
3.3.2	Which Topics to Select? . . . . .	57
3.3.3	Production of Models . . . . .	58
3.3.4	Sampling the Test Set . . . . .	58
<b>4</b>	<b>Knowing the Content Before Judging its Effect</b>	<b>61</b>
4.1	Learning More About the Topics in the Dataset . . . . .	61
4.1.1	What Sources to Explore, and How, to Obtain More Knowledge? . . . . .	61
4.1.2	A Dual Style of Results . . . . .	63
4.1.3	A Greater Understanding of the Composition of the Sets . . . . .	65
4.2	Exploring the Distribution of the Test Sets . . . . .	67
4.2.1	A Continuing Experiment, but with a Different Focus . . . . .	67

4.2.2	The Test Sets' Content in Its Many Shapes . . . . .	69
4.2.3	What Does the Content Tell Us? . . . . .	74
<b>5</b>	<b>Estimating the Effect of Lexicon-Based Generated Models</b>	<b>77</b>
5.1	How to Evaluate the Accuracy of a Model . . . . .	77
5.1.1	A Variety of Metrics . . . . .	77
5.1.2	Open Source Tools for Evaluation . . . . .	81
5.1.3	My Choice: Kami App . . . . .	82
5.2	Comparing Two Different Lexicon-Based Models . . . . .	84
5.2.1	How Was the Comparative Analysis Conducted? . . . . .	84
5.2.2	WER, CER, Insertions, Deletions, Substitutions, "Versus Text": a Plethora of Statistics and Errors Comparisons at hand . . . . .	89
5.2.3	A Lack of Influence of the Lexicon in Prediction Errors . . . . .	91
<b>III</b>	<b>Exploring the Impact of the Infralexical Level: The N- Gram</b>	<b>95</b>
<b>6</b>	<b>Studying the N-Gram and Its Distribution Within the Dataset</b>	<b>97</b>
6.1	A New Level of Study . . . . .	97
6.1.1	A Choice Induced by the Previous Experiment's Results . . . . .	97
6.1.2	Same Dataset, New Interests . . . . .	98
6.2	The N-Gram from Every Angle . . . . .	99
6.2.1	Which N-Grams Will Be Studied? . . . . .	99
6.2.2	How to Obtain My Series of N-Grams? . . . . .	100
6.2.3	Various Visualizations to Better Understand My N-Gram Series . . . . .	102
6.3	What Did We Learn About the Distribution of the N-Gram? . . . . .	105
6.3.1	A Significant, Sizable Gap Between the Sets . . . . .	105
6.3.2	The Most and Least Popular N-Grams: A Promising Lead Towards Un- derstanding the Efficiency of the Model . . . . .	110
<b>7</b>	<b>Evaluating the Impact of the N-Gram Using Prediction Errors</b>	<b>115</b>
7.1	Obtaining an Error List . . . . .	115
7.1.1	The Data from the Comparative Analysis As the Source . . . . .	115
7.1.2	The Structure of the Table of Erroneous N-Grams . . . . .	117
7.1.2.1	Table Distribution . . . . .	117
7.1.2.2	Table Colours . . . . .	118
7.2	A Diversity of Information About N-Grams . . . . .	119

7.2.1	What Information About the N-Grams Does the Table Provide? . . . . .	119
7.2.1.1	Composition of the Table . . . . .	119
7.2.1.2	New Table Elements . . . . .	120
7.2.1.3	Illustrated Examples from the Table . . . . .	120
7.2.2	What Additional Information Can We Gather from the Table? . . . . .	123
7.2.2.1	Table Data Numerical Distribution . . . . .	123
7.2.2.2	Table Data Percentage Distribution . . . . .	125
7.3	Many Results, No Concrete Outcome . . . . .	125
7.3.1	A First General Idea of the Recognition's Skills of the Models . . . . .	125
7.3.1.1	"COLUMNS" Section Analysis . . . . .	125
7.3.1.2	"N-GRAMS" Section Analysis . . . . .	126
7.3.1.3	Percentages Analysis . . . . .	128
7.3.1.4	Tables Data Distribution Analysis . . . . .	129
7.3.2	Some More Elements of Response With the Details of the Tables . . . . .	130
7.3.2.1	Tetragrams Results . . . . .	130
7.3.2.2	Trigrams Results . . . . .	132
7.3.2.3	Bigrams Results . . . . .	134
7.3.2.4	General Analysis . . . . .	135
7.3.2.5	Conclusion . . . . .	136

## **IV Multilingualism: An Answer to the N-Grams 139**

### **8 A New Dataset: Multilingual Documents from the Holocaust 141**

8.1	History and Presentation of the Corpus . . . . .	141
8.1.1	Historical Background . . . . .	141
8.1.2	The Holocaust . . . . .	142
8.1.3	Working on the Holocaust . . . . .	143
8.1.4	The EHRI Online Editions . . . . .	144
8.2	A Multilingual Dataset . . . . .	145
8.2.1	Choosing the Right Script . . . . .	145
8.2.2	Same Script, Different Languages . . . . .	146
8.2.3	Several Languages, Various Diacritics . . . . .	148
8.2.4	Training a Multilingual Text Recognition Model . . . . .	149

### **9 Analysing How the Multilingual Model Cooperates 151**

9.1	Learning About the Alphabet of the Multilingual Dataset . . . . .	151
-----	---	-----

9.1.1	Obtaining Insight into the Alphabet . . . . .	151
9.1.2	An Uneven Distribution of the Alphabet . . . . .	155
9.2	Exploring the Interactions of the N-Grams of the Multilingual Dataset . . . . .	158
9.2.1	A Study of the N-Grams, Notably the Least and Most Popular . . . . .	158
9.2.2	Exploring the Similarities and Differences Between Languages . . . . .	163
9.2.2.1	Results for the Most Popular N-Grams . . . . .	163
9.2.2.2	Results for the Least Popular N-Grams . . . . .	163
9.2.2.3	General Observations . . . . .	164
<b>10</b>	<b>Testing the Efficacy of the Model by Observing Its Errors</b>	<b>165</b>
10.1	A Personalized Test Set for the Multilingual Model . . . . .	165
10.1.1	What Should Be Included for an Efficient Test? . . . . .	165
10.1.2	Presentation of the Test Set . . . . .	168
10.2	The Metrics: Evidence of the Model's Overall Efficiency . . . . .	171
10.2.1	Applying the Model to Produce Metrics . . . . .	171
10.2.2	A Result Highlighting the Strengths and Weaknesses of the Model . . . . .	174
10.3	Comparing Errors to Ground Truth: A Positive Response to Our Hypothesis . . . . .	176
10.3.1	Gathering Data About the Prediction Errors of the EHRI Model . . . . .	176
10.3.2	What Can We Learn from the General Observations of the Tables? . . . . .	180
10.3.3	A Variety of Languages Leading to the Same Conclusion . . . . .	187
10.3.3.1	English Results . . . . .	188
10.3.3.2	Danish Results . . . . .	188
10.3.3.3	Slovak Results . . . . .	189
10.3.3.4	French Results . . . . .	190
10.3.3.5	Italian Results . . . . .	191
	<b>Conclusion</b>	<b>195</b>
	An Answer To This Thesis' Research Question: The N-Grams . . . . .	195
	Uppercase and Diacritics: The N-Grams' Limits . . . . .	196
	Concrete Applications of Acquired Knowledge . . . . .	197
	N-Grams and Digital Scholarly Editions . . . . .	198
	Impact of the N-Grams: What About Handwritten Texts? . . . . .	199

<b>Annex I</b>	
<b>Tables</b>	<b>203</b>
<b>A Tables of Results (Content Analysis)</b>	<b>204</b>
<b>B Metrics (Comparative Analysis)</b>	<b>206</b>
B.1 Metrics by Letters for the Set Other . . . . .	206
B.2 Metrics by Letters for the Set War . . . . .	209
<b>C Tables of Results (Token Analysis)</b>	<b>212</b>
C.1 Total . . . . .	212
C.2 Set Other . . . . .	213
C.3 Set War . . . . .	214
C.4 Set Ground Truth . . . . .	215
C.5 Comparison . . . . .	216
<b>D Comparison Transcription (Token Error Analysis)</b>	<b>219</b>
<b>E Tables of N-Grams (Token Error Analysis)</b>	<b>235</b>
<b>F Alphabet Analysis (Multilingual Model)</b>	<b>269</b>
F.1 Distribution of the Alphabet By Characters . . . . .	269
F.2 Distribution of the Alphabet By Language . . . . .	269
F.3 General Distribution of the Alphabet . . . . .	269
<b>G Metrics (Multilingual Model)</b>	<b>278</b>
<b>H Comparison Transcription (Multilingual Model)</b>	<b>281</b>
<b>I Tables of N-Grams (Multilingual Model)</b>	<b>289</b>
<b>Annex II</b>	
<b>Figures</b>	<b>311</b>
<b>J Word Clouds (Lexicon Analysis)</b>	<b>312</b>
<b>K Word Clouds (Content Analysis)</b>	<b>315</b>
<b>L Diagrams (Content Analysis)</b>	<b>317</b>

<b>M Bar Charts (Token Analysis)</b>	<b>321</b>
<b>Annex III</b>	
<b>Datasets</b>	<b>327</b>
<b>N Paul d’Estournelles de Constant’ Correspondence</b>	<b>328</b>
N.1 Composition of the Set War (SW) . . . . .	328
N.2 Composition of the Set Other (SO) . . . . .	328
N.3 Composition of the Test Set . . . . .	329
N.4 Composition of the Ground Truth . . . . .	329
<b>O European Holocaust Research Infrastructure Editions</b>	<b>331</b>
O.1 List of the EHRI Online Editions Used to Create the Dataset . . . . .	331
O.2 Distribution of the Dataset . . . . .	331
O.3 Distribution of the Test Set . . . . .	332
<b>Annex IV</b>	
<b>Resources</b>	<b>333</b>
<b>P Python Scripts</b>	<b>334</b>
P.1 Lexicon Analysis . . . . .	334
P.1.1 <code>clean_groundtruth.py</code> . . . . .	334
P.1.2 <code>groundtruth.py</code> . . . . .	334
P.2 Content Analysis . . . . .	334
P.2.1 <code>counting.py</code> . . . . .	334
P.2.2 <code>creating_list_of_tokens</code> . . . . .	334
P.2.3 <code>producing_part_of_speech_tagging.py</code> . . . . .	334
P.2.4 <code>producing_tokens_and_lemmas.py</code> . . . . .	335
P.2.5 <code>removing_incorrect_tokens.py</code> . . . . .	335
P.3 Token Analysis . . . . .	335
P.3.1 <code>creating_list_of_tokens_from_ngrams.py</code> . . . . .	335
P.3.2 <code>creating_list_of_tokens_from_texts.py</code> . . . . .	335
P.3.3 <code>producing_dictionary_of_ngrams_occurrences.py</code> . . . . .	335
P.3.4 <code>producing_list_of_ngrams.py</code> . . . . .	335
P.4 Token Error Analysis . . . . .	335
P.4.1 <code>producing_list_of_ngrams_from_several_models.py</code> . . . . .	335



---

P.5	Multilingual Model EHRI . . . . .	336
P.5.1	<code>count_characters.py</code> . . . . .	336
P.5.2	<code>producing_list_of_ngrams_from_several_models.py</code> . . . . .	336
P.5.3	<code>creating_list_of_tokens_from_ngrams.py</code> . . . . .	336
P.5.4	<code>producing_dictionary_of_ngrams_occurrences.py</code> . . . . .	336
P.5.5	<code>producing_list_of_ngrams.py</code> . . . . .	336
<b>Q</b>	<b>Glossary</b>	<b>337</b>
<b>R</b>	<b>Web Resources</b>	<b>340</b>
<b>Annex V</b>		
<b>PhD activities</b>		<b>343</b>
<b>S</b>	<b>Chronology</b>	<b>344</b>
<b>T</b>	<b>Conferences</b>	<b>346</b>
T.1	Next Gen TEI 2021 . . . . .	346
T.2	EVEILLE 2022 . . . . .	346
T.3	Humanistica 2022 . . . . .	346
T.4	Documents anciens et reconnaissance automatique des écritures manuscrites . .	346
T.5	Digital Humanities 2022 Responding to Asian Diversity . . . . .	347
T.6	TEI 2022 . . . . .	347
T.7	DH2023 . . . . .	347
T.8	From Source to Full Text: Workshop on Using Automatic Text Recognition (ATR)	347
T.9	CHR 2023 . . . . .	348
T.10	Natural Language Processing Meets Holocaust Archives . . . . .	348
T.11	Workshop LREC-COLING 2024: Holocaust Testimonies as Language Resources	348
T.12	EHRI Academic Conference 2024 . . . . .	348
T.13	DH2024 Reinvention & Responsibility . . . . .	349
<b>U</b>	<b>Training</b>	<b>350</b>
U.1	Ethique de la recherche . . . . .	350
U.2	How to write and publish your paper . . . . .	350
U.3	Découvrir la diffusion de la culture scientifique et technique - Module 1 . . . . .	351
U.4	La science ouverte . . . . .	351
U.5	Intégrité scientifique dans les métiers de la recherche . . . . .	351

## Table of Contents

---

U.6	Impacts environnementaux du numérique . . . . .	352
U.7	L'intelligence artificielle... avec intelligence . . . . .	352
U.8	Team Management . . . . .	352
U.9	Project Management . . . . .	352
U.10	Ma thèse en 180 secondes . . . . .	353
 <b>Bibliography</b>		 <b>355</b>
	Automatic Text Recognition (ATR) . . . . .	356
	ATR Tools . . . . .	356
	Deep Learning . . . . .	358
	DH Projects . . . . .	358
	Digital Scholarly Editing . . . . .	360
	Ground Truth . . . . .	362
	Historical Background . . . . .	363
	Languages . . . . .	364
	Model Evaluation . . . . .	364
	Natural Language Processing . . . . .	365
	Open Science . . . . .	365
	Post-ATR Correction . . . . .	366
	Segmentation . . . . .	366
 <b>List of Figures</b>		 <b>373</b>
 <b>List of Tables</b>		 <b>377</b>
 <b>Index</b>		 <b>379</b>



# RÉSUMÉ FRANÇAIS DE LA THÈSE

---

Cette thèse travaille à identifier ce qu'un modèle de reconnaissance de texte apprend pendant son entraînement, à travers l'examen du contenu de ses vérités de terrain et de ses erreurs de prédiction, avec comme objectif principal l'amélioration des connaissances sur le fonctionnement d'un réseau de neurones, par le biais d'expériences focalisées sur des documents tapuscrits (tapés à la machine).

L'introduction de cette thèse se concentre sur les deux éléments décisifs dans la mise en place du travail de thèse : les éditions scientifiques numériques et la reconnaissance automatique de texte. Tout d'abord, l'introduction présente l'idée d'un *workflow* (suite de tâches) pour la création d'éditions scientifiques numériques.

Après avoir répondu à la question "Qu'est-ce qu'une édition scientifique numérique ?" et expliqué les principes de la science ouverte ainsi que les principes FAIR (Facile à trouver, Accessible, Interopérable, Réutilisable), qui sont au cœur de ce *workflow*, cette section détaille les différentes composantes d'un *workflow* (numérisation, segmentation, transcription, encodage, publication) et souligne l'importance de sa mise en œuvre. Elle fait ensuite une présentation des applications pratiques d'un *workflow* d'éditions scientifiques numériques, à travers divers outils et standards régulièrement employés.

Par la suite, l'introduction se concentre sur la reconnaissance automatique de texte (ATR), qui constitue un élément fondamental de cette recherche doctorale. Les défis liés à la transcription et aux jeux de données utilisés dans les expériences de recherche sont abordés. Un aperçu des concepts clés de l'ATR, à savoir la reconnaissance optique de caractères (OCR) et la reconnaissance de texte manuscrit (HTR), est également fourni, avec un développement du principe de documents tapuscrits, considérés comme une ressource idéale pour cette recherche.

Enfin, l'introduction se termine par l'énoncé du problème de recherche et la présentation du plan de la thèse, structuré en quatre parties: l'évolution des techniques d'ATR, l'influence (nulle) du lexique dans la reconnaissance, l'exploration de l'impact des n-grammes, et la réponse effective fournie par le multilinguisme.

La première partie de cette thèse présente un état de l'art de la reconnaissance automatique de texte, une technique en constante évolution.

Elle débute par un examen des méthodes de transcription, en traçant l'évolution de la re-

connaissance optique de caractères (OCR) à la reconnaissance automatique de texte (ATR). L'OCR, basé sur l'utilisation de *bounding boxes* (rectangle entourant un caractère, au sein d'une image, et définissant son étendue spatiale) et de caractères, est d'abord abordé à travers ses différents développements et ses objectifs, avant d'en analyser le fonctionnement interne. Ensuite, l'émergence de l'ATR est explorée, en mettant en évidence le rôle clé des réseaux de neurones dans cette transition, motivée par la nécessité de nouvelles méthodes plus performantes.

Cette première partie de la thèse se poursuit en explorant les perspectives ouvertes par l'apprentissage profond, avec l'émergence de logiciels propriétaires et *open source*, basés sur des algorithmes génériques puis sur des réseaux de neurones, pour des applications allant de la reconnaissance de textes imprimés à celle de manuscrits. Elle se penche notamment sur la comparaison des performances des principaux logiciels de reconnaissance de texte manuscrit, soit Transkribus et eScriptorium.

Pour finir, elle aborde des problématiques récentes telles que les nouvelles approches explorées en segmentation, la difficulté que représente la création de vérités de terrain efficace et suffisante, et la question de ce qu'implique la correction post-ATR.

La deuxième partie de cette thèse s'articule autour de la première hypothèse formulée pour répondre à la question de recherche, portant sur l'influence du lexique, qui s'est révélée être nulle selon les résultats d'expériences.

Dans un premier temps, le corpus thématique utilisé pour orienter cette étude est présenté : la correspondance de Paul d'Estournelles de Constant. Après une présentation du corpus, qui incluent le contexte historique (la Première Guerre mondiale), les figures principales (Paul d'Estournelles de Constant et Nicholas Murray Butler), ainsi que la nature de la correspondance et le projet à l'origine de cette recherche (Projet DAHN), la diversité et la richesse de ce corpus, à la fois éclectique (guerre, politique, société, famille, etc.) et volumineux (1500 lettres), sont soulignées.

Enfin, la réflexion menée pour constituer l'ensemble de test est détaillée, en prenant en compte les critères de sélection des thèmes, la création de modèles, et l'échantillonnage de l'ensemble de test.

Dans un second temps, cette partie cherche à analyser le contenu des données avant d'en évaluer leur effet. Cela commence par un approfondissement des connaissances sur les thèmes du jeu de données, en explorant les sources disponibles et les méthodes utilisées pour acquérir davantage d'informations. Grâce à une double approche de résultats (listes de fréquence et nuages de mots), une meilleure compréhension de la composition des ensembles de données a été obtenue.

L'exploration de la distribution des ensembles de test se poursuit ensuite, mais avec un objectif différent : observer le contenu des jeux de données sous ses diverses formes (tokens (unité

lexicale), lemmes (unité lexicale), *part-of-speech* (classe grammaticale)) et interpréter ce qu'il révèle.

Enfin, la thèse s'intéresse à l'estimation de l'effet des modèles générés à partir du lexique. Après avoir expliqué les méthodes d'évaluation de la précision d'un modèle à travers diverses métriques (distance de Levenshtein, taux d'erreur de mots (WER), taux d'erreur de caractères (CER)) et outils, un focus est opéré sur l'outil *open source* KaMI, choisi pour cette estimation. Une comparaison des deux modèles lexicaux, développés avec les ensembles de tests, est ensuite réalisée. Cette analyse comparative se base sur une série de statistiques, incluant le WER, le CER, ainsi que les insertions, les suppressions, les substitutions et l'observation des résultats de « *versus text* », une option de KaMI. L'étude conclut sur l'absence d'influence significative du lexique dans les erreurs de prédiction.

La troisième partie de cette thèse se concentre sur l'exploration d'une nouvelle hypothèse relative à l'impact du niveau infralexical, spécifiquement des n-grammes (séquences de  $n$  caractères).

Dans un premier temps, l'étude des n-grammes et de leur distribution au sein du jeu de données est réalisée. Ce choix d'analyse est motivé par les résultats de l'expérience précédente, utilisant essentiellement le même jeu de données, mais poursuivant de nouveaux objectifs.

L'exploration des n-grammes est effectuée sous tous les angles. Après avoir défini les n-grammes à étudier (2-grams ou bigrammes, 3-grams ou trigrammes, 4-grams ou tétragrammes) et la méthode d'obtention de la série de n-grammes, diverses visualisations sont créées afin de mieux comprendre cette série (tableaux de résultats, graphiques en barres).

La section se conclut sur les enseignements tirés de la distribution des n-grammes, mettant en évidence un écart significatif de taille entre les séries et soulignant une piste prometteuse pour comprendre l'efficacité des modèles en analysant les n-grammes les moins et les plus populaires.

Dans un second temps, l'impact des n-grammes est évalué en lien avec les erreurs de prédiction. Après l'obtention d'une liste d'erreurs tirée des données de l'analyse comparative, la structure du tableau des n-grammes erronés est présentée, en abordant à la fois leur distribution et la codification par couleurs.

La diversité des informations sur les n-grammes est ensuite discutée, tant à travers le tableau et ses éléments variés, que par les informations extrapolées, telles que la distribution numérique et en pourcentage des données.

Cette étude se conclut par de nombreux résultats sans issue concrète, tout en fournissant une première vision générale des capacités de reconnaissance des modèles, grâce à l'analyse des tableaux produits et de détails issus des séries de n-grammes.

La quatrième et dernière partie de cette thèse vise à confirmer l'influence des n-grammes, une solution non certifiée par les expériences précédentes, en intégrant le critère du multilinguisme,

qui se révèle produire une réponse significative à cette influence.

Dans un premier temps, le nouveau jeu de données utilisé est présenté, à savoir des documents multilingues liés à l'Holocauste. Une présentation du corpus est fournie, en le situant dans son contexte historique (la Seconde Guerre mondiale), et plus spécialement celui de l'Holocauste. Par la suite, le contexte de recherche sur l'Holocauste est exposé, en mettant en avant notamment le projet EHRI et ses éditions en ligne.

Enfin, l'accent est mis sur les particularités d'un jeu de données multilingue, qui nécessite le choix approprié de script. Cela peut inclure un même script (ici, Latin script), utilisé pour différentes langues (allemand, anglais, danois, hongrois, polonais, slovaque, tchèque), mais également plusieurs langues qui englobent différentes diacritiques, soit les signes ajoutés à des caractères pour en transformer l'interprétation (accent grave, aigu, tréma, etc.). Tous ces éléments doivent être pris en compte pour élaborer un modèle multilingue efficace pour la reconnaissance de texte.

Dans un second temps, l'analyse de la coopération des données du modèle multilingue est entreprise. Tout d'abord, l'accent est mis sur la connaissance de l'alphabet du jeu de données multilingues. Cette étude permet d'observer une distribution inégale des caractères au sein de l'alphabet.

Ensuite, l'interaction des n-grammes au sein du jeu de données multilingues est explorée, incluant une étude approfondie des n-grammes, tant ceux qui sont les moins populaires que ceux qui sont les plus utilisés, ainsi qu'une analyse des similitudes et des différences entre les langues présentes dans le corpus.

Dans un dernier temps, l'efficacité du modèle multilingue est testée en examinant ses erreurs. Pour ce faire, un ensemble de test personnalisé est créé, en réfléchissant aux éléments à inclure afin d'assurer un test efficace. Les réflexions qui ont guidé cette sélection sont présentées avant de décrire l'ensemble de tests, qui contient des données rédigées dans des langues incluses dans les vérités de terrain du modèle, mais également des langues inconnues de l'entraînement.

L'attention se porte ensuite sur les métriques, qui fournissent une indication de l'efficacité globale du modèle. Après l'application du modèle pour produire ces métriques, les résultats mettent en évidence les forces et les faiblesses du modèle.

Enfin, une comparaison des erreurs avec les vérités de terrain permet d'obtenir une réponse positive à l'hypothèse initiale. Cette analyse est facilitée par la collecte de données sur les erreurs de prédiction du modèle EHRI. Les observations générales des tableaux révèlent des enseignements significatifs. La variété de langues étudiées (anglais, danois, slovaque, français et italien) aboutit à une conclusion commune : l'influence effective des n-grammes.

La conclusion souligne tout d'abord que ce travail a permis de répondre à la question de recherche de cette thèse, axée sur les n-grammes. Des expériences sur des jeux de données

unilingue et multilingue ont permis d'observer que les bigrammes, et dans certains cas, les trigrammes également, sont utiles pour la reconnaissance de texte. Il est cependant nécessaire d'avoir un jeu de données équilibré et avec une quantité modérée de données, soit ni trop, ni trop peu.

Dans cette conclusion sont également rappelés les limites inhérentes à l'utilisation des n-grammes qui ont pu être observés dans les résultats d'expérience, notamment en ce qui concerne les majuscules et les signes diacritiques. Dans la création des vérités de terrain pour la production de modèle, il sera important de vérifier qu'une quantité suffisante de majuscules et de signes diacritiques est présente afin d'être sûre qu'il puisse être reconnu, ce qui peut représenter une tâche difficile puisqu'ils peuvent être assez rares dans un corpus.

Ensuite, les applications concrètes des connaissances acquises sont présentées, notamment l'idée qu'il est préférable de privilégier la qualité au lieu de la quantité, et ainsi de ne pas ajouter aveuglément et aléatoirement de nouvelles vérités de terrain pour entraîner un modèle de reconnaissance de texte, ainsi que le fait que l'entraînement de modèles multilingues est possible. En suivant, la conclusion de la thèse détaille le lien entre les n-grammes et les éditions scientifiques numériques, et la manière dont les connaissances nouvellement acquises pourraient être appliquées aux éditions scientifiques numériques précédemment développés, ainsi qu'à de futures éditions.

Enfin, une ouverture est proposée concernant l'impact des n-grammes sur un nouveau type de source, en posant la question suivante : "Qu'en est-il des textes manuscrits ?". Les différences dans l'écriture manuscrite, au regard de mêmes caractères ou mots, pourraient créer des irrégularités même dans l'établissement des n-grammes, puisque deux mêmes suites de caractères n'apparaîtraient pas obligatoirement de la même façon, ce qui pourrait entraver la reconnaissance de motifs récurrents au sein des vérités de terrain.







**Titre :** Comprendre le processus de reconnaissance automatique de texte : entraînement de modèles, vérité de terrain et erreurs de prédiction

**Mot clés :** Reconnaissance automatique de texte (ATR) ; Vérités de terrain ; Entraînement de modèle ; Erreurs de prédiction ; Documents tapuscrits ; Réseaux de neurones

**Résumé :** Cette thèse travaille à identifier ce qu'un modèle de reconnaissance de texte apprend pendant son entraînement, à travers l'examen du contenu de ses vérités de terrain et de ses erreurs de prédiction. L'intention principale ici est d'améliorer les connaissances sur le fonctionnement d'un réseau de neurones, avec des expériences focalisées sur des documents tapuscrits. Les méthodes utilisées se sont concentrées surtout sur l'exploration approfondie des données d'entraînement, l'observation des erreurs de prédiction des modèles et la corrélation entre les deux. Une première hypothèse, basée sur l'influence du lexique, fut non concluante. Ce-

pendant, cela a dirigé les observations vers un nouveau niveau d'étude, s'appuyant sur un niveau infralexical : les n-grammes. La distribution de ceux des données d'entraînement a été analysée et subséquemment, comparée à celle des n-grammes récupérés dans les erreurs de prédiction. Des résultats prometteurs ont conduit à une exploration approfondie, tout en passant d'un modèle de langue unique à un modèle multilingue. Des résultats concluants m'ont permis de déduire que les n-grammes pourraient effectivement être une réponse valide aux performances de reconnaissance.

**Title:** Understanding the Automatic Text Recognition's Process: Model Training, Ground Truth and Prediction Errors

**Keywords:** Automatic Text Recognition (ATR); Ground Truth; Model Training; Prediction Errors; Typewritten Documents; Neural Networks

**Abstract:** This thesis works on identifying what a text recognition model can learn during its training, through the examination of its ground truth's content, and its prediction's errors. The main intent here is to improve the knowledge of how a neural network operates, with experiments focused on typewritten documents. The methods used mostly concentrated on the thorough exploration of the training data, the observation of the model's prediction's errors, and the correlation between both. A first hypothesis, based on the influence

of the lexicon, was inconclusive. However, it steered the observation towards a new level of study, relying on an infralexical level: the n-grams. Their training data's distribution was analysed and subsequently compared to that of the n-grams retrieved from the prediction errors. Promising results lead to further exploration, while upgrading from single-language to multilingual model. Conclusive results enabled me to infer that the n-grams might indeed be a valid answer to recognition's performances.