



HAL
open science

Standard-based lexical models for automatically structured dictionaries

Mohamed Khemakhem

► **To cite this version:**

Mohamed Khemakhem. Standard-based lexical models for automatically structured dictionaries. Document and Text Processing. Université Paris Cité, 2020. English. NNT : 2020UNIP7146 . tel-03274454v2

HAL Id: tel-03274454

<https://theses.hal.science/tel-03274454v2>

Submitted on 30 Jun 2021

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.



Université de Paris

École doctorale Sciences Mathématiques de Paris-Centre (ED 386)
Laboratoire ALMAnaCH

Standard-based Lexical Models for Automatically Structured Dictionaries

Par Mohamed KHEMAKHEM

Thèse de Doctorat en Informatique - Traitement Automatique des Langues

Dirigée par Laurent ROMARY

Présentée et soutenue publiquement le 01/10/2020

Devant un jury composé de:

Benoît CRABBÉ, Professeur, Université de Paris - UFRL, Président de Jury
Rute COSTA, Professeur, CLUNL - UNL Lisbonne, Rapporteur
Patrice BELLOT, Professeur, LIS - Polytech' Marseille, Rapporteur
Karlheinz MÖRTH, Directeur, ACDH - ÖAW Vienne, Examineur
Toma Tasovac, Directeur, DARIAH Berlin et BCDH Belgrade, Examineur
Patrice LOPEZ, Fondateur et gérant, Science-Miner, Examineur



Except where otherwise noted, this is work licensed under
<https://creativecommons.org/licenses/by-nc-nd/3.0/fr/>

Declaration of Authorship

I, Mohamed KHEMAKHEM, declare that this thesis titled, "Standard-based Lexical Models for Automatically Structured Dictionaries" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"Dictionary: The universe in alphabetical order."

"Un dictionnaire, c'est tout l'univers par ordre alphabétique"

"Ein Wörterbuch ist das gesamte Universum in alphabetischer Reihenfolge"

Anatole France

UNIVERSITÉ PARIS DIDEROT - UNIVERSITÉ DE PARIS

Abstract

ED 386 - École Doctorale Sciences Mathématiques de Paris-Centre
Department of Computer Science

Doctor of Philosophy

Standard-based Lexical Models for Automatically Structured Dictionaries

by Mohamed KHEMAKHEM

Abstract

Dictionaries could be considered as the most comprehensive reservoir of human knowledge, which carry not only the lexical description of words in one or more languages, but also the common awareness of a certain community about every known piece of knowledge in a time frame. Print dictionaries are the principle resources which enable the documentation and transfer of such knowledge. They already exist in abundant numbers, while new ones are continuously compiled, even with the recent strong move to digital resources.

However, a majority of these dictionaries, even when available digitally, is still not fully structured due to the absence of scalable methods and techniques that can cover the variety of corresponding material. Moreover, the relatively few existing structured resources present limited exchange and query alternatives, given the discrepancy of their data models and formats.

In this thesis we address the task of parsing lexical information in print dictionaries through the design of computer models that enable their automatic structuring. Solving this task goes hand in hand with finding a standardised output for these models to guarantee a maximum interoperability among resources and usability for downstream tasks.

First, we present different classifications of the dictionary resources to delimit the category of print dictionaries we aim to process. Second, we introduce the parsing task by providing an overview of the processing challenges and a study of the state of the art. Then, we present a novel approach based on a top-down parsing of the lexical information. We also outline the architecture of the resulting system, called GROBID-Dictionaries, and the methodology we followed to close the gap between the conception of the system and its applicability to real-world scenarios.

After that, we draw the landscape of the leading standards for structured lexical resources. In addition, we provide an analysis of two ongoing initiatives, TEI-Lex-0 and LMF, that aim at the unification of modelling the lexical information in print and electronic dictionaries. Based on that, we present a serialisation format that is inline with the schemes of the two standardisation initiatives and fits the approach implemented in our parsing system.

After presenting the parsing and standardised serialisation facets of our lexical models, we provide an empirical study of their performance and behaviour. The investigation is based on a specific machine learning setup and series of experiments carried out with a selected pool of varied dictionaries. We try in this study to present different ways for feature engineering and exhibit the strength and the limits of the best resulting models. We also dedicate two series of experiments for exploring the scalability of our models with regard to the processed documents and the employed machine learning technique.

Finally, we sum up this thesis by presenting the major conclusions and opening new perspectives for extending our investigations in a number of research directions for parsing entry-based documents.

Résumé

Les dictionnaires peuvent être considérés comme le réservoir le plus compréhensible de connaissances humaines, qui contiennent non seulement la description lexicale des mots dans une ou plusieurs langues, mais aussi la conscience commune d'une certaine communauté sur chaque élément de connaissance connu dans une période de temps donnée. Les dictionnaires imprimés sont les principales ressources qui permettent la documentation et le transfert de ces connaissances. Ils existent déjà en grand nombre, et de nouveaux dictionnaires sont continuellement compilés.

Cependant, la majorité de ces dictionnaires dans leur version numérique n'est toujours pas structurée en raison de l'absence de méthodes et de techniques évolutives pouvant couvrir le nombre du matériel croissant et sa variété. En outre, les ressources structurées existantes, relativement peu nombreuses, présentent des alternatives d'échange et de recherche limitées, en raison d'un sérieux manque de synchronisation entre leurs schémas de structure.

Dans cette thèse, nous abordons la tâche d'analyse des informations lexicales dans les dictionnaires imprimés en construisant des modèles qui permettent leur structuration automatique. La résolution de cette tâche va de pair avec la recherche d'une sortie standardisée de ces modèles afin de garantir une interopérabilité maximale entre les ressources et une facilité d'utilisation pour les tâches en aval.

Nous commençons par présenter différentes classifications des ressources dictionnaires pour délimiter les catégories des dictionnaires imprimés sur lesquelles ce travail se focalise. Ensuite, nous définissons la tâche d'analyse en fournissant un aperçu des défis de traitement et une étude de l'état de l'art. Nous présentons par la suite une nouvelle approche basée sur une analyse en cascade de l'information lexicale. Nous décrivons également l'architecture du système résultant, appelé GROBID-Dictionaries, et la méthodologie que nous avons suivie pour rapprocher la conception du système de son applicabilité aux scénarios du monde réel.

Ensuite, nous prestons des normes clés pour les ressources lexicales structurées. En outre, nous fournissons une analyse de deux initiatives en cours, TEI-Lex-0 et LMF, qui visent à unifier la modélisation de l'information lexicale dans les dictionnaires imprimés et électroniques. Sur cette base, nous présentons un format de sérialisation conforme aux schémas des deux initiatives de normalisation et qui est assorti à l'approche développée dans notre système d'analyse lexicale.

Après avoir présenté les facettes d'analyse et de sérialisation normalisées de nos modèles lexicaux, nous fournissons une étude empirique de leurs performances et de leurs comportements. L'étude est basée sur une configuration spécifique d'apprentissage automatique et sur une série d'expériences menées avec un ensemble sélectionné de dictionnaires variés. Dans cette étude, nous essayons de présenter différentes manières d'ingénierie des caractéristiques et de montrer les points forts et les limites des meilleurs modèles

résultants. Nous consacrons également deux séries d'expériences pour explorer l'extensibilité de nos modèles en ce qui concerne les documents traités et la technique d'apprentissage automatique employée.

Enfin, nous clôturons cette thèse en présentant les principales conclusions et en ouvrant de nouvelles perspectives pour l'extension de nos investigations dans un certain nombre de directions de recherche pour l'analyse des documents structurés en un ensemble d'entrées.

Acknowledgements

This thesis has been supported by the European PARTHENOS project, as a part of the Work Package 4 under grant No. 654119, and by the BasNum project under the ANR grant No. ANR-18-CE38-0003.

This work would not have seen the light without the support of people who believed in my personal and scientific capacities. It goes without saying that I had a great chance to have Prof. Dr. Laurent Romary as a PhD advisor. Besides his exceptional supervision and support, I learnt a lot from working with him in different exciting projects. I can't thank him enough for the trust he put in me. I would like to thank Prof. Dr. Rute Costa and Prof. Dr. Patrice Bellot for the time they dedicated to evaluate my thesis.

I'm very thankful to my family and especially to my sister, Aïda, who taught me how to code my first algorithms and believed in me when many others did not. No words could describe my gratitude to her and to my brother in law, Hssan, for their unconditional support.

I also would like to thank Dr. Patrice Lopez for developing and maintaining the GROBID infrastructure and his precious guidance and technical support. I thank all my colleagues at Inria-ALMANaCH team and Centre Marc Bloch for being part of an outstanding research environment. My thanks go of course to Dr. Richard Eckart de Castilho, who shaped my programming skills during my internship at UKP Lab. I also can't forget the participants of the GROBID-Dictionaries Workshop series, in particular Simon Gabay, who provided me with valuable early feedback and helped me in extending the scope of this work.

Special thanks go to Richard James for the review of my English.

Last but not least, I'm grateful to all my friends and cousins Cristina, Meryem, Amal, Karim, Ahmed, Oussama and Meher who were there during the hard times.

Contents

Declaration of Authorship	iii
Abstract	vii
Acknowledgements	xi
1 Introduction	1
1.1 Overview	1
1.2 Task Definition & Research Questions	2
1.3 Thesis Organisation	3
2 Dictionaric Resources	7
2.1 Introduction	7
2.2 Macro-Logical Structure	7
2.2.1 Onomasiological Dictionaries	7
2.2.2 Semasiological Dictionaries	7
2.3 Format	8
2.3.1 Born-digital Dictionaries	9
2.3.2 Digitised Dictionaries	9
2.3.3 Machine Readable Dictionaries	9
2.3.4 Computerised Dictionaries, Lexical Databases and NLP Lexica	9
2.4 Time	10
2.4.1 Modern Dictionaries	10
2.4.2 Legacy Dictionaries	10
2.5 Content	10
2.5.1 Multilinguality Dimension	12
a. Monolingual	12
b. Bilingual	12
c. Multilingual	12
2.5.2 Lexicographic Dimension	12
a. Lexical	12
b. Encyclopaedic	12
c. Etymological & Diachronic	13
2.6 Chapter Summary	13
3 Parsing Lexical Information in Print Dictionaries	15
3.1 Introduction	15
3.2 Challenges	15
3.2.1 Born-digital vs Digitised Documents	15

3.2.2	Logical Structure vs Physical Structure	18
	Logical Structure	18
	Physical Structure	20
3.3	State of the art	20
3.3.1	Rule-based	20
3.3.2	Probabilistic Models	22
	Text Sequence Labelling	23
	Graphical Models for Sequence Labelling	23
	Evaluation metrics	26
	Previous Work	27
3.3.3	Probabilistic Models for Parsing Bibliographic Data	31
3.4	Chapter Summary	33
4	Lexical Models for Parsing Print Dictionaries	35
4.1	Introduction	35
4.2	Cascading Parsing	35
4.2.1	Approach	36
4.2.2	Bibliographic Information and Lexical Information: Parsing Similarities and Differences	36
4.2.3	GROBID	39
4.3	GROBID-Dictionaries	40
4.3.1	Cascading Lexical Models	40
4.3.2	Feature Engineering	45
4.3.3	Model Activation & Call	47
4.4	Lexicographic Knowledge Acquisition	50
4.4.1	Easing Setup	50
	Initial Configuration: IT Expert Use	50
	Enhanced Usability & Unified Execution Environment: DH Use	51
4.4.2	Lightening Annotation	52
	Creating Training Data	53
	Training Data Annotation	53
	Train, Test and Evaluate	54
4.4.3	Training Domain Experts and Collecting Feedback	55
	Hands-on Sessions with End-Users	55
	Participants with Diverse Backgrounds	56
	Various tested Material	56
	Outcome & Gathered insights	56
4.5	Chapter Summary	57
5	Standards for Structured Lexical Resources	59
5.1	Introduction	59
5.2	TEI	59
5.2.1	Background	59
5.2.2	Modelling Perspectives	60
	Encoding Level	60
	Encoding Workflow & Choices	61
5.2.3	TEI-based Lexical Resources	63

5.2.4	Discussion	63
5.3	LMF (2008)	63
5.3.1	Background	64
5.3.2	Meta-model	64
5.3.3	Serialisation	65
5.3.4	LMF-based Lexical Resources	65
5.3.5	Discussion	67
5.4	OntoLex-Lemon	68
5.4.1	Background	68
5.4.2	Modelling	68
5.4.3	Serialisation	71
5.4.4	OntoLex-Lemon-based Resources	71
5.4.5	Discussion	73
5.5	Chapter Summary	73
6	Novel Standardised Schemes for Encoding Dictionaries	75
6.1	Introduction	75
6.2	TEI-Lex-0	75
6.2.1	Context	76
6.2.2	Modelling challenges	77
	Recursive entries	77
	Revising the entry model	78
	Written and Spoken Forms	79
	Miscellaneous	82
6.2.3	Serialisation Model	82
	Entry Class Diagram	82
	Discussion	84
6.3	LMF Reloaded	84
6.3.1	Context	84
6.3.2	Modelling Challenges	85
	Restructuring	85
	Simplification	86
	Enrichment	86
6.3.3	TEI Serialisation Model: ISO 24613-4	89
6.3.4	Discussion	93
6.4	GROBID-Dictionaries Output Scheme	93
6.4.1	Modelling Challenges	94
	Cascading Modelling	94
	Satisfying different encodings for the same structures	97
6.4.2	Serialisation Models	98
	Internal Serialisation	98
	Final Serialisation & Discussion	101
6.4.3	Chapter Summary	103

7	GROBID-Dictionaries in Action	105
7.1	Introduction	105
7.2	Machine Learning Experiment Setup	106
7.2.1	Experimental goals	106
7.2.2	Interfering Factors	106
	Sample Selection and Annotation	107
	OCR Impact	109
7.2.3	Dictionary Samples & Annotation	111
	Dictionnaire de la Langue Française (DLF)	112
	Easier English Basic Dictionary (EEBD)	112
	Mixtec-Spanish Dictionary (MxSp)	115
	Fang-French (FangFr) & French-Fang (FrFang) Dictionary	116
7.3	Experiment Series 1: Training with One Dictionary	117
7.3.1	Feature Engineering Experiments	119
	Experiments	119
	Discussion	127
7.3.2	Learning Curve Experiments	127
	Experiments	127
	Discussion	132
7.4	Experiment Series 2: Training with Multiple Dictionaries	132
7.4.1	Feature engineering Experiments	133
	Experiments	133
	Discussion	141
7.4.2	Learning Curve	141
	Experiments	141
	Discussion	141
7.5	Experiment Series 3: Testing with Unseen Dictionaries	144
	Experiments	145
	Discussion	145
7.6	Scaling up	149
7.6.1	Experiment Series 4: Beyond dictionaries	149
	Legacy Manuscript Auction Catalogs	149
	Address Directories	155
7.6.2	Experiment Series 5: Deep Learning	156
	Introduction	156
	Experiments	157
7.7	Chapter Summary	160
8	Summary & Perspectives	161
8.1	Summary	161
8.2	Perspectives	163
A	Descriptive Vectors and Feature Templates	167
A.1	Descriptive Vectors	167
A.1.1	Descriptive Vectors for Dictionary Segmentation Model (a.k.a. GROBID's First Model)	167

A.1.2	Descriptive Vectors for Dictionary Body Segmentation Model Onward	168
A.2	Feature Templates	169
A.2.1	Unigram Templates of Dictionary Segmentation Model (a.k.a. GROBID's First Model)	169
A.2.2	Unigram Templates from Dictionary Body Segmentation Model Onward	172
A.2.3	Bigram Templates of Dictionary Segmentation Model (a.k.a. GROBID's First Model)	174
A.2.4	Bigram Templates from Dictionary Body Segmentation Model Onward	177
A.2.5	Engineered Templates from Dictionary Body Segmentation Model Onward	179
B	Models Call for the "Parse Full Dictionary" Level in GROBID-Dictionaries' Web Application	183

List of Figures

2.1	Onomasiological Approach	8
2.2	Semasiological Approach	8
2.3	Excerpt from Basnage Dictionary (Furetière, 1701) of the entry ABORDER	11
2.4	Classification of Dictionaric Resources	14
3.1	Metadata text introduced by a PDF engine in the text of dictio- nary pages (Publishing, 2009)	16
3.2	Digitised entry in Basnage dictionary (Furetière, 1701)	17
3.3	Left: Entry ACT in (Publishing, 2009). Right: Entries ABUSE (Pub- lishing, 2009)	19
3.4	Excerpts from different dictionaries with different logical struc- ture	21
3.5	Text Sequence Labelling	23
3.6	Example of bilingual dictionaries processed by (Ma et al., 2003) and (Karagol-Ayan, Doermann, and Dorr, 2003)	28
3.7	Illustration of the Hyper-Level zones in a scientific paper recog- nised by Header Segmentation Model in GROBID	32
4.1	First and second segmentation levels of a dictionary page (Larousse, 1972)	37
4.2	Example of the segmentation performed by the Lexical Entry model (Larousse, 1972)	37
4.3	Left: Excerpt from Bibliographic References Section in (Khe- makhem, Herold, and Romary, 2018). Right: Excerpt of Lexical Entries in (Publishing, 2009)	38
4.4	Excerpt from GROBID's architecture ¹	39
4.5	Sequence labelling a dictionary article (Larousse, 1972) using the "Lexical Entry" segmentation model	41
4.6	GROBID-Dictionaries's Cascading Architecture	42
4.7	Implemented MATTER Workflow	49
4.8	Cascading Model Selection in GROBID-Dictionaries	49
4.9	A GROBID-Dictionaries image in a Docker container	52
4.10	Training data annotation in oxygen author mode for the first model: page headers vs. page body	54
5.1	Entry CABBAGE in (Mueller, 1878) and examples of its two Levels of TEI encoding	61
5.2	Different Depths of TEI encoding for Logical Structure	62

5.3	Dependencies between the LMF (2008) core and extension packages (Soria, Monachini, and Vossen, 2009)	65
5.4	LMF (2008) Object Diagram for Modelling the MWE "DEAD CENTER"	66
5.5	OntoLex-Lemon Core Model (McCrae et al., 2017)	69
5.6	First Instance of OntoLex-Lemon Model for the MWE "DEAD CENTRE"	70
5.7	Another Example of Instantiating OntoLex-Lemon Model for the MWE "DEAD CENTRE"	70
5.8	Turtle Serialisation of the MWE "DEAD CENTRE" and its Components	72
6.1	Left: Dictionary Article in an Arabic Dictionary (Almonjid, 2014). Right: Corresponding Minimal TEI-Lex-0 Encoding	78
6.2	Minimal Encoding of Homographs as Entries in TEI-Lex-0 (Dictionary Articles from (Publishing, 2009))	79
6.3	Minimal Encoding of Homographs as Senses in TEI-Lex-0 (Dictionary Articles from (Publishing, 2009))	80
6.4	Left: POS Encoding Using <gram> Element. Right: POS Encoding Using <pos> Element	80
6.5	Left: Extract from Mixtec-Spanish Dictionary (Alvarado, 1593) Containing Inflected Forms. Right: Corresponding Encoding in TEI-Lex-0	81
6.6	TEI-Lex-0 Serialisation for <entry> Model	83
6.7	Example of Modelling the MWE "DEAD CENTRE" and its Components using the New LMF Core and MRD Models	87
6.8	Instance of the New Etymology Extension for the Entry "DICTIONARY" in (Ernest, 1966)	88
6.9	LMF Serialisation for <entry> Model	89
6.10	Example of LMF Modelling of Homographs with two Form Objects	91
6.11	Example of the new LMF Serialisation for the MWE "DEAD CENTER" and its Components	92
6.12	Etymological Constructs in (Mueller, 1878)	95
6.13	Minimal TEI Encoding as an Output of the Etym/Quote Model	95
6.14	Minimal TEI Encoding as an Output of the Etym Model	96
6.15	Internal Serialisation of GROBID-Dictionaries's Models	99
6.16	Final Serialisation of the <entry> Model in GROBID-Dictionaries	102
7.1	Excerpt from a Mixtec-Spanish Dictionary (Alvarado, 1593)	110
7.2	Excerpt from the Dictionnaire de la Langue Française Dictionary (Littré, 1873)	113
7.3	Excerpt from the Easier English Basic Dictionary (Publishing, 2009)	114
7.4	Excerpt from the Fang-French & French-Fang Dictionary (Galley, 1964) (FangFr)	117
7.5	Excerpt from the Fang-French & French-Fang Dictionary (Galley, 1964) (FrFang)	118

7.6	Mono-sample Evaluation of the Dictionary Segmentation Model Using two Classes of Templates	120
7.7	Mono-sample Evaluation of the Dictionary Body Segmentation Model Using three Classes of Templates	122
7.8	Mono-sample Evaluation of the Lexical Entry Model Using three Classes of Templates	123
7.9	Mono-sample Evaluation of the Form Model Using three Classes of Templates	124
7.10	Mono-sample Evaluation of the GramGrp Model Using three Classes of Templates	125
7.11	Mono-sample Evaluation of the Sense Model Using three Classes of Templates	126
7.12	Mono-sample Evaluation of the Sub-Sense Model Using three Classes of Templates	128
7.13	Learning Curve of the Different Models Given the Number of Training Pages from DLF	129
7.14	Learning Curve of the Different Models Given the Number of Training Pages from EEBD	130
7.15	Learning Curve of the Different Models Given the Number of Training Pages from MxSp	130
7.16	Learning Curve of the Different Models Given the Number of Training Pages from FangFr	131
7.17	Learning Curve of the Different Models Given the Number of Training Pages from FrFang	131
7.18	Multi-sample Evaluation of the Dictionary Segmentation Model Using two Classes of Templates	133
7.19	Multi-sample Evaluation of the Dictionary Body Segmentation Model Using three Classes of Templates	135
7.20	Multi-sample Evaluation of the Lexical Entry Model Using three Classes of Templates	136
7.21	Multi-sample Evaluation of the Form Model Using three Classes of Templates	137
7.22	Multi-sample Evaluation of the GramGrp Model Using three Classes of Templates	138
7.23	Multi-sample Evaluation of the Sense Model Using three Classes of Templates	139
7.24	Multi-sample Evaluation of the Sub-Sense Model Using three Classes of Templates	140
7.25	Learning Curve of the Different Models Given the Number of Training Pages from 2 BL	142
7.26	Learning Curve of the Different Models Given the Number of Training Pages from 3 BL	142
7.27	Learning Curve of the Different Models Given the Number of Training Pages from DB	143
7.28	Learning Curve of the Different Models Given the Number of Training Pages from ML	143

7.29	Learning Curve of the Different Models Given the Number of Training Pages from All	144
7.30	Evaluation of the Dictionary Segmentation Model Using the Best Templates and Testing with Unseen Sample	145
7.31	Evaluation of the Dictionary Body Segmentation Model Using the Best Templates and Testing with Unseen Sample	146
7.32	Evaluation of the Lexical Entry Model Using the Best Templates and Testing with Unseen Sample	146
7.33	Evaluation of the Form Model Using the Best Templates and Testing with Unseen Sample	147
7.34	Evaluation of the GramGrp Model Using the Best Templates and Testing with Unseen Sample	147
7.35	Evaluation of the Sense Model Using the Best Templates and Testing with Unseen Sample	148
7.36	Evaluation of the Sub-Sense Model Using the Best Templates and Testing with Unseen Sample	148
7.37	Resemblance between MSCs and Encyclopedic Dictionaries	150
7.38	Encoding of an MSC	151
7.39	Evaluation of the Dictionary Segmentation Model Trained and Tested with MSCs and Dictionaries	152
7.40	Evaluation of the Dictionary Body Segmentation Model Trained and Tested with MSCs and Dictionaries	153
7.41	Evaluation of the Lexical Entry Model Trained and Tested with MSCs and Dictionaries	153
7.42	Evaluation of the Form Model Trained and Tested with MSCs and Dictionaries	154
7.43	Evaluation of the Sense Model Trained and Tested with MSCs and Dictionaries	154
7.44	Evaluation of the Lexical Entry Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labelers	158
7.45	Evaluation of the Form Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labelers	158
7.46	Evaluation of the GramGrp Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labelers	159
7.47	Evaluation of the Sense Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labelers	159
7.48	Evaluation of the Sub-Sense Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labelers	160
8.1	Excerpt from the Bibliography Collection (Wiegand, 2014) used for Experimenting the Combination of GROBID and GROBID-Dictionaries Models	165
B.1	Selecting Morphological and Grammatical Models	183
B.2	Selecting Semantic Models	184
B.3	Selecting Etymological Models	184
B.4	Selecting Lexical Entry Model for Parsing Related Entries	184
B.5	Selecting Cross-Reference Models	185

B.6	Selecting Lexical Entry Model for Parsing Sub-Entries	185
B.7	State after of Full Parsing Selection	186

List of Tables

4.1	Some of the Dictionaries experimented with in some sessions of the workshop series	57
6.1	Main labels of the etymology extension	94
7.1	Page Sampling Statistics (Bowers, Khemakhem, and Romary, 2019)	108
7.2	Field Level Evaluation of the Dictionary Segmentation Model	109
7.3	Field Level Evaluation of the Dictionary Body Segmentation Model	109
7.4	Field Level Evaluation of the Lexical Entry Model	111
7.5	Summary of the First Series of Experiments	127
7.6	Summary of the Second Series of Experiments	141
7.7	Proposed TEI Encoding of Entries in Address Directories (Didot-Bottin, 1901)	156

*In loving memory of my father, Rchid
KHEMAKHEM*

Chapter 1

Introduction

1.1 Overview

Computational lexicography is a field which has emerged from the combination of lexicography - the field dealing with the design and compilation of lexica for human use - with computational methods. This "adventurous" combination, as called by Hanks, 2013, that started in the late 60's freed western lexicographers from the content dictated alphabetical order compilation to the logical order proceeding. The impact of further computerising lexicography has been significant and led to the establishment of a large field which, according to Gibbon, 2000, covers tasks such as *text mining* for *corpus-based* lexicon construction, the construction of lexica for Natural Language Processing (*NLP*) applications, automatic acquisition of *syntactic* or *semantic* information from texts, re-use of Machine-Readable Dictionaries (*MRDs*) for new lexica, and computer production of lexica for human use.

The creation of lexica for *NLP* use has been focused in the last two decades on corpus based approaches at the expense of endeavours dealing with the reuse of *MRDs* which represent in general the digitised version of print dictionaries. This imbalanced situation has actually been the consequence of increased accessibility of large corpora coupled with a rapidly growing number of advanced dedicated processing tools. To favorise lexical acquisition from corpora over exploiting *MRDs*, Lemnitzer and Kunze, 2005 further argued that such resources are too old, internally and mutually inconsistent, narrow in scope, missing important information like frequency and distribution information and finally biased towards infrequent phenomena such as obsolete senses and usages.

In the previous decade, such a claim used to be relatively valid for well-resourced languages (e.g., English, French, German, Dutch..) and was strongly questionable for under-resourced languages where print dictionaries always represented the backbone for creating *NLP* lexica. In fact, print dictionaries encapsulate semi-structured lexical information vital for either rapidly harvesting material for building lexica in less resourced languages or for enriching well established resources with information about the diachrony of words. But after the retro-digitisation movement following the breakthroughs in Optical Character Recognition (*OCR*) techniques, legacy dictionaries have returned in the recent years to the spotlight. Currently an abundant number of digitised dictionaries are constantly uploaded to publicly

accessible repositories^{1 2}. In addition, the copyright clearance of such documents, as result of ageing or freely born intellectual properties, has opened up the possibility of their re-use. In parallel, established concepts in corpus linguistics, such as comparable corpora (Kenning, 2010), have raised the question of applying the same approaches for lexical acquisition from the large newly available corpora of dictionaries.

These facts, along with relatively primitive approaches to automatically structuring such resources, have unwittingly created a huge gap where most of the dedicated methods are ad-hoc and, consequently, unable to cover the important stream of dictionary material for NLP downstream applications.

1.2 Task Definition & Research Questions

In this thesis, we aim to reduce the aforementioned gap by studying the nature of digital print dictionaries and finding a suitable approach for structuring such resources on a large scale.

The task we are trying to solve comprises two major steps: the first deals with *automatically parsing* lexical information encapsulated in the text of a digital dictionary. The identification of such constructs greatly depends on a second milestone which studies the design of a *scheme* that supports different organisations of lexical information in the target classes of dictionaries. Therefore, we are considering the exploration of dedicated standards for modelling lexica as they provide a suitable framework for scaling up the creation of interoperable lexica (Calzolari, 2008). Switching back and forth between the two sub-tasks is necessary as the complexity of the lexical information needs to be supported by the parsing technique and the parser(s)'s grammar.

For the first sub-task, most existing approaches have been focused on *rule-based* methods (Mykowiecka, Rychlik, and Waszczuk, 2012; Fahmy and Fayed, 2014; Maxwell and Bills, 2017; Ranaivo-Malançon et al., 2017; Steingrímsson, 2018) with few attempts to make use of *machine learning* techniques (Karagol-Ayan, Doermann, and Dorr, 2003; Crist, 2011; Bago and Ljubešić, 2015). The dictionaries tested in these approaches have a relatively flat structure and the main task addressed has been tagging the tokens of dictionary articles. None of the state of the art methods has proposed an *end-to-end* architecture for structuring print dictionaries that have different and deep structures such as nested *entries* or *senses*. The only exception to this is the work of Karagol-Ayan, Doermann, and Dorr, 2003 who have built a framework for digitising, parsing and generating lexica focused on salient structures in *bilingual dictionaries*.

In the above mentioned related works, few addressed (Maxwell and Bills, 2017) or mentioned (Mykowiecka, Rychlik, and Waszczuk, 2012) the need to generate standardised lexica using Text Encoding Initiative (*TEI*) (Budin, Majewski, and Mörth, 2012) or Lexical Markup Framework (*LMF*) (Francopoulo

¹<https://archive.org/search.php?query=dictionaries>

²<https://galica.bnf.fr/conseils/content/dictionnaire>

et al., 2006). Other lexicographic projects (Eckle-Kohler and Gurevych, 2012; Czaykowska-Higgins, Holmes, and Kell, 2014) have been more focused on the standardisation of existing *NLP* resources and shared the use of *TEI* and *LMF*, which are considered to be key frameworks within the lexicographic community. The need to find a unified scheme and guidelines that combine the best of the two standards has been clearly formalised (Czaykowska-Higgins, Holmes, and Kell, 2014; Romary, 2015) with some practical suggestions, as the division in their use within the lexicographic community represents an obstacle to the large scale interoperability dream. It is important to point out that the task of categorising lexical structures and their modelling using possible alternatives offered by a standard already represents a highly debatable subject among lexicographers.

These challenges raise the following research questions, which we tackle in this thesis:

- **RQ1 - Sample-agnostic Models for Print Dictionary Parsing:** The first question concerns the study of different existing print dictionaries by analysing their physical and logical facets and the consequent processing complexity. Then we want to explore building generic *lexical models* to parse different categories of dictionaries. We call a *model* a parser that enables the analysis and the structuring of text blocks in a dictionary according to a defined scheme.
- **RQ2 - Unified Scheme for Structured Dictionaries:** The second investigation will be carried out on the leading standards for modelling lexica, namely *TEI* and *LMF*. The goal is to find a compromising *scheme* that meets both the lexicographic requirements and parsing constraints identified in the first question.
- **RQ3 - Scaling-up Lexical Models:** In the last question we will empirically address the leverage of the constructed lexical models by studying the improvement of their performance on the same dictionary and other dictionaries. The possibility of finding more generic models covering documents with similar content is by no means excluded from our investigation.

Approaching all these research questions at once represents in itself a core research question in Digital Humanities (DH): to what extent could a collaboration be possible between the *computational* and the *humanist*?

1.3 Thesis Organisation

This research work is structured in eight chapters. In the current section, we present an overview of the organisation of the thesis:

Chapter 2 - Dictionaric Resources

Having established the context, in this chapter we will present a classification of existing digital print dictionaries. The goal of such a step is to gradually

delimit, based on different aspects of such material, the category of dictionary resources that fall within the scope of our research.

Chapter 3 - Parsing Lexical Structures in Print Dictionaries

In the third chapter we present the challenges related to the parsing of lexical information in the already defined target dictionary resources. Then we will draw the state of the art by reporting related studies and explain how they approached the aforementioned difficulties. We will also discuss the most advanced techniques to position our work within the computational landscape.

Chapter 4 - Lexical Models for Automatically Structuring Print Dictionaries

The fourth chapter is dedicated to the presentation of our approach and the architecture of the lexical models. We show how we were inspired by the analogy between the parsing of bibliographic data and our task and we will demonstrate how we succeeded in adapting an existing machine learning infrastructure to the requirements of our context. Details about the process of building these models for the new infrastructure called GROBID-Dictionaries, the challenges, and the consequent conceptual and technical solutions will be provided.

Chapter 5 - Standards for Structured Lexical Resources

In this chapter we give an overview of the most widely used standards within the lexicographic community. We set out to explain the shortcomings of these frameworks with respect to the requirements of automatic processing tasks, mainly the parsing application presented in Chapter 4.

Chapter 6 - Novel Standardised Schemes for Encoding Dictionaries

The pitfalls identified in Chapter 5 will be tackled in this chapter, where we present standardisation initiatives in line with our unification goals. We will present our involvement in shaping TEI-Lex-0 and the new LMF frameworks and how such interaction has been translated into the definition of the scheme of our lexical models.

Chapter 7 - GROBID-Dictionaries in Action

In this chapter more experiments, carried out with GROBID-Dictionaries and different dictionaries in different setups, will be described and evaluated. The goal here is to provide an extensive overview of the performance of our machine learning models when they are exposed to different dictionary material. The limits of the architecture and the possibilities to scale up parsing

dictionaries based on the collected data and more advanced techniques will be discussed. Experiments performed on non-dictionaric resources will also be presented to show the genericity of our approach and resulting models.

Chapter 8 - Summary and Perspectives

The last chapter will be dedicated to the conclusions drawn from the results of the experiments presented and community feedback. We will also present our vision for building on our work with respect to the requirements of the task undertaken and the advances in machine learning techniques.

Chapter 2

Dictionaric Resources

2.1 Introduction

Dictionaries or *lexica* (the plural of *lexicon*) have different representations and content, and therefore can be categorised according to different aspects. This thesis is focused on a parsing task applied to a specific category of dictionaric resources. To define this task, we firstly aim in this chapter to present the most common classifications of dictionaries and establish the nature of the material we are dealing with. The following categories are based on several dimensions that touch on the macro-logical structure, format, age and content of a digital dictionary.

2.2 Macro-Logical Structure

By macro-logical structure in this context, we denote the representation of relationships among words, meanings and concepts in a dictionary. Semasiology and Onomasiology are directly opposite approaches for studying and presenting such relationships.

2.2.1 Onomasiological Dictionaries

An *onomasiological* - from the Greek *ónoma* (name) and *logos* (study) - approach focuses on categorising words expressing a certain concept based on a possible *synonymy* of the words. It tries to answer the question "How is a concept expressed?". Figure 2.1 illustrates the word synonymy mechanism addressed in an onomasiological system.

Such an approach is commonly applied for building terminologies, but is also followed by lexicographers for compiling *thesauri* (Roget, 1911; Lamy and Towell, 1998) and *synonymy dictionaries* (Urdang, 1986).

Onomasiological resources are beyond the scope of the dictionaries studied in this thesis.

2.2.2 Semasiological Dictionaries

Semasiology - from the Greek *semasia* (meaning) - deals with *polysemy*, the coexistence of possible meanings of a particular word. The question to be

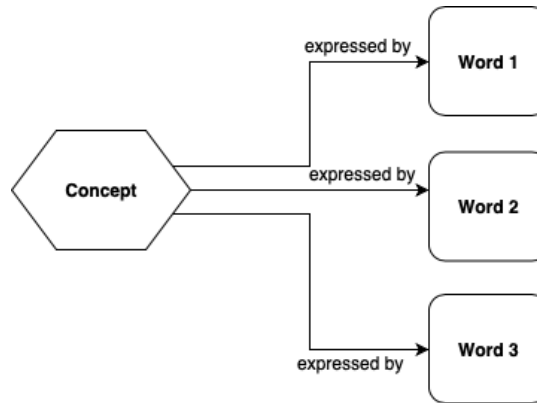


FIGURE 2.1: Onomasiological Approach

answered in such an approach is "What does a word mean?". Figure 2.2 illustrates the polysemy mechanism studied in a semasiological system.

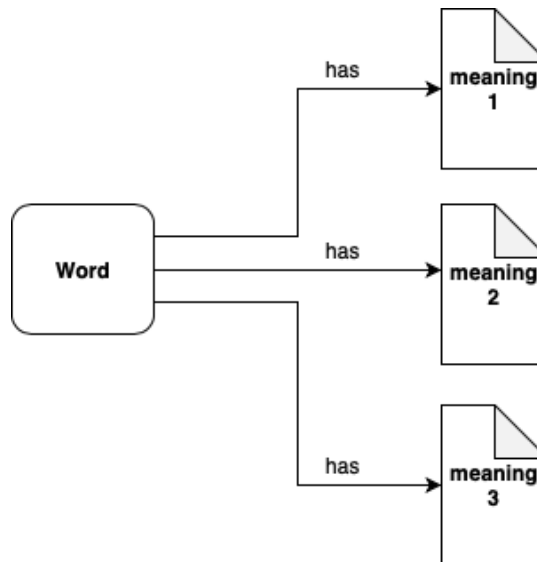


FIGURE 2.2: Semasiological Approach

Classic dictionaries (Hornby et al., 1974; Hindley et al., 2000), used by language learners, are the most common form of dictionaries compiled following the semasiological approach.

In the context of this thesis, it is semasiological dictionaries that we are interested in.

2.3 Format

The format of a dictionary is determinant for the techniques to be used for the NLP task that we are addressing. We are interested in dictionaries in digital formats having specific features. In this section, we present a classification of possible formats of dictionaries and we specify the ones that fall within the present context.

2.3.1 Born-digital Dictionaries

This category denotes dictionaries that have been originally compiled using computers and generated as a file that could be displayed using a web browser, a text editor or any other dedicated software (e.g., Microsoft Word, PDF viewer, etc). Their text content is usually represented as a sequence of text searchable by a machine and its quality greatly depends on the tool used to generate the file.

2.3.2 Digitised Dictionaries

This class gathers dictionaries in digital format resulting from the application of an Optical Character Recognition (OCR) system on scans or images of a print dictionary. Such resources are called also "retro-digitised" or "OCR'd" dictionaries.

The automatically recognised text, also known as OCRs, can be exported by the OCR tool as a separate text document or on the original scan as text layer, visible upon selection. The quality of the text and its layout information in such dictionaries varies widely, depending on the characteristics of the original scans, the OCR tool and the conditions in which the digitisation process has been carried out.

Both digital born and digitised dictionaries represent a typical format for resources to be structured using methods developed in the context of this work.

2.3.3 Machine Readable Dictionaries

Certain born digital dictionaries are generated from MRDs (Atkins, 1991; Muller and Beddow, 2002; Dendien and Pierrel, 2003) which are resources that consist of text files controlled by pieces of code guiding the typesetting process. Such a format is commonly used by publishers to produce print and digital versions of dictionaries.

For the structuring task we are addressing, we do not consider processing this kind of resources.

2.3.4 Computerised Dictionaries, Lexical Databases and NLP Lexica

Computerised dictionaries are structured resources reflecting the logical structure of lexical entries such as part of speech, definitions, examples, etc. The explicit structure can be easily transformed into resources, suitable for computational tasks, called "Lexical Databases" or "NLP Lexica".

In the context of the present thesis, such a class of resources represents the target format.

2.4 Time

Time is reflected in dictionaries through certain lexicographic designs and practices which could lead to different dictionaries for very similar content. We distinguish two broad categories of resources: *modern* and *legacy*. It is hard to draw a clear line between the two of them as the classification is relatively subjective. But we will try to describe both categories based on our experience with different materials we have been dealing with.

2.4.1 Modern Dictionaries

These are the dictionaries that we use nowadays where, from the layout perspective, there are clear boundaries among the lexical entries and their main lexical information. Typographic information is omnipresent and in most cases consistently used as they have been generated and controlled automatically. The lexical information in such resources is usually expressed in modern language with fewer prolix descriptions.

2.4.2 Legacy Dictionaries

The age of existing legacy dictionaries ranges from a few decades (Larousse, 1948; Hornby et al., 1974) to few centuries (Furetière, 1701; Littré, 1873). Prose lexical descriptions, poetic examples and old language constructs and formulations are very common in the content of such resources. Besides the complex semasiological system in representing relatively similar lexical structures such as entries, sub-entries, related entries and senses, inconsistency in the use of the very system can be noticed in different parts of the same dictionary. This fact is totally natural as such resources took years, in some cases decades, to be compiled and often by more than one lexicographer. Typographic information is usually poor, which is not helpful for a human reader to infer the lexicographic system followed by the creator of the dictionary. Figure 2.3 shows a case of a less informative typography in a legacy dictionary along with a complex logical structure of entries, senses and related entries that could be interpreted differently.

Most digital legacy dictionaries were originally retro-digitised from print dictionaries that were produced before the use of computers was introduced. In addition to the logical complexity briefly explained above, issues related to the OCR process, which will be explained in Section 3.2, make the recognition of the text in such documents very challenging for any advanced OCR system and consequently result in a noisy text in the digitised version.

Modern and legacy dictionaries are both in the center of focus of this thesis.

2.5 Content

We see the content of a dictionary as having two dimensions: *multilinguality* and of *lexicographic* information.

ABORDER. v. act. & n. Arriver en quelque lieu, spécialement par mer; prendre terre. *J'aborde, j'abordai, je suis abordé.* La flotte des Indes est *abordée* en Espagne. Les Marchands *abordent* de tous côtez à la foire de Beaucaire le 21. de Juillet. On ne convient pas qu'Enée soit *abordé* en Italie. **SENT. DE CL.** Il n'est pas seur d'*aborder* de cette côte, parce que la mer se retirant, les vaisseaux y demeurent à sec. **ABL.** Il ne put *aborder* à cause que la rive étoit escarpée. **ID.** Ils *aborderent* en des pais inconnus. **VAU.** Il signifie, Arriver en foule. Les presens *abordent* chez moi de toutes parts. **ABL.** Il signifie encore, Entrer, parvenir. Nous ne pumes *aborder* de la place, parceque toutes les avenues étoient gardées. Il fut impossible d'*aborder* jusqu'à l'autel à cause de la foule du peuple.

ABORDER, signifie aussi, Venir à bord d'un vaisseau. On a contraint ce vaisseau ennemi de mettre pavillon bas, & d'*aborder.* On dit de deux vaisseaux qui s'approchant en droiture, s'enferrent par leurs éperons, qu'ils *s'abordent* de franc étable. On dit, *Aborder* au port, sur les rivieres: mais en termes de marine, quand on veut dire gagner le rivage, on ne dit pas *aborder,* mais *mouiller, toucher, rendre le bord.*

ABORDER, signifie encore, Attaquer l'ennemi hardiment, tant par mer, que par terre. Les vaisseaux dans les batailles tâchent toujours d'empêcher qu'on ne les *aborde.* Ce bataillon *aborda* les ennemis avec une contenance ferme.

ABORDER, signifie aussi, Approcher quelcun pour lui parler. Ce Ministre est si honnête, qu'on l'*aborde* facilement. Il l'*aborda* avec ce compliment. Les Grands doivent soulager le respect, & la timidité de ceux qui n'osent les *aborder.* **M. ESP.**

On dit aussi, qu'on n'oseroit, ou qu'on ne peut *aborder* d'un lieu, soit à cause de sa situation, ou de quelque autre obstacle qui le rend inaccessible, soit des voleurs, ou des bêtes farouches. Quand ce dogue est lâché, on n'oseroit *aborder* de la basse-cour.

ABORDER la remise. Terme de Fauconnerie, qui se dit lorsque la perdrix poussée par l'oiseau a gagné quelque buisson: alors on *aborde la remise* sous le vent, afin que les chiens sentent mieux la perdrix cachée dans le buisson.

ABORDÉ, ÉE. part. & adj.

FIGURE 2.3: Excerpt from Basnage Dictionary (Furetière, 1701) of the entry ABORDER

2.5.1 Multilinguality Dimension

The number of languages to be used in a semasiological system has a direct impact on the constructs in the content of a dictionary.

a. Monolingual

Monolingual dictionaries represent the basic form of dictionaries where words in a certain language are explained and described in the same language. Referencing words in other languages remains however possible, especially for expressing relationships between words and their origins, a study called *etymology* or *diachrony*.

b. Bilingual

Translation equivalents, along with explanations of varying length, constitute the core of *bilingual* dictionaries. Such resources are often used by learners of new languages to find translations of words in a newly learnt language or the other way around. In such dictionaries, it is very common to find two parts: one part for language A to language B and the second part for language B to language A. Compared to monolingual dictionaries, bilingual ones have shorter entries as the goal in this case is not to give an exhaustive lexical description.

c. Multilingual

Multilingual dictionaries are the least common form of dictionaries as multilingualism is relatively less frequent among people. The lexical description in such a category is even more compact than in bilingual dictionaries as an alignment between translation equivalents in three or more languages should be preserved for readability and pedagogical purposes.

2.5.2 Lexicographic Dimension

The last dimension in our classification of dictionaric resources represents the category of the lexicographic content.

a. Lexical

Lexical dictionaries are known as the default dictionaries for most dictionary users. This kind of document has the goal of representing the meaning and other lexical aspects related to *lexemes*, or words of a language. *Named entities*, such as person or city names, are not in the scope of such dictionaries, but are included in the following category.

b. Encyclopaedic

As mentioned earlier, named entities fall in the scope of encyclopaedic content in dictionaries. Such a content can be found in separate documents called

encyclopaedic dictionaries or *encyclopaedias* (Hindley, 1971; Larousse, 1982) as it can usually constitute a separate part in a dictionary. However, in many cases, the encyclopaedic content is mixed with the lexical content in one dictionary and the alphabetical order is the only sorting system that applies.

c. Etymological & Diachronic

As signalled earlier, *etymology* and *diachrony* are disciplines for studying the origin of words, which implies investigating relationships between words not only in the same language but also in different ones. Such a description could be condensed within the description of an entry in a lexical dictionary but may be separated and exhaustively described in dedicated documents called *etymological* dictionaries (Ernout et al., 1951; Ernest, 1966).

2.6 Chapter Summary

In this chapter we presented a classification of dictionaries that goes from macro structures and format to the class of content that a dictionary can encompass. In Figure 2.4 we summarise the consequent classification of digital dictionaries we are dealing with in the following chapters.

The rectangles in green represent the class of dictionary material we might have as input to structure, whereas the one in orange is the target category we want to reach.

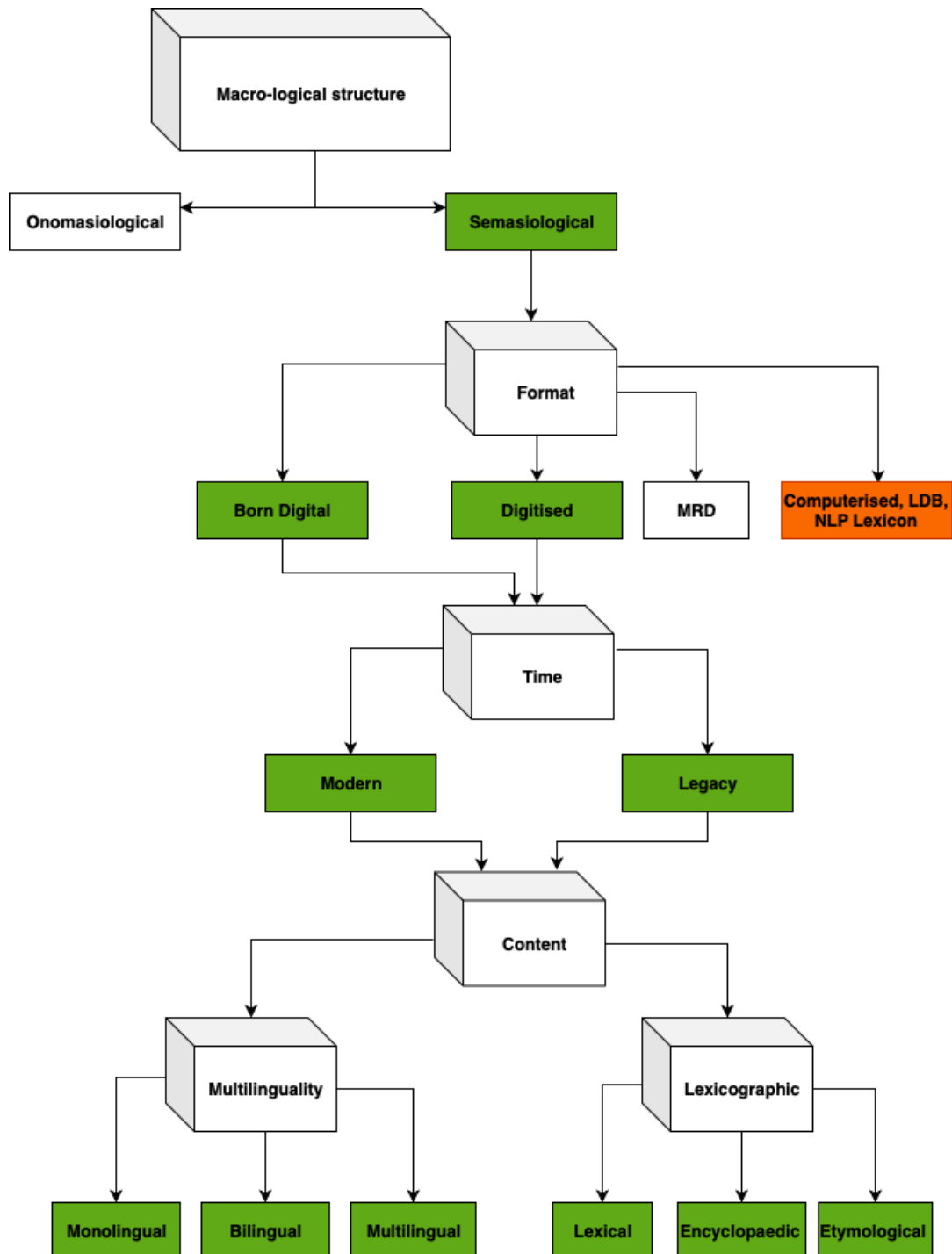


FIGURE 2.4: Classification of Dictionaric Resources

Chapter 3

Parsing Lexical Information in Print Dictionaries

3.1 Introduction

According to the Oxford English Dictionary ¹, *parse* as a verb has two meanings. The first is defined as "resolve (a sentence) into its component parts and describe their syntactic roles" while the second is a sub-sense of the former, related to the computing domain, and has the meaning of "analyse (a string or text) into logical syntactic components". Both senses apply in the context of the parsing task we are aiming at. The second has a broader computational meaning and in the case of dictionaries could be parsing the text of a page to recognise structures such as headers, footers, lexical entries or their components.

Dictionaries falling in the target categories defined in the previous chapter represent several challenges for their parsing. Previous studies have tried to overcome these obstacles partially or entirely to reach different levels of text analysis. In the current chapter we showcase the challenges for parsing print dictionaries and present the state of the art.

3.2 Challenges

Parsing print dictionaries comes up against several obstacles that need to be studied in order to have a better understanding of the computational challenges involved. We try to decode these aspects by focusing on two axes.

3.2.1 Born-digital vs Digitised Documents

Print dictionaries in digital format have been produced using computers by means of either *Dictionary Writing Systems* (DWSs) for the case of born-digital dictionaries or digitisation softwares obviously for digitised ones.

Abel, 2012 gives a general overview of the different DWS uses and new trends in compiling dictionaries, following the era of pen and paper. *In-house* systems, implemented in academic projects such as DEB (Horák and Rambosek, 2007) and Jibiki (Mangeot-Nagata, 2006), and *off-the-shelf* tools, used

¹<https://en.oxforddictionaries.com/definition/parse>

extensively by publishers as well as in academia, for instance TshwaneLex (Joffe and De Schryver, 2004), DPS (McNamara, 2003), or ABBYY Lingvo (Kuzmina and Rylova, 2010), along with other commercial tools and ad-hoc editors have helped lexicographer to compile, retype or convert dictionaries.

The diversity of the technologies used remained, however, uncontrolled which resulted in *pre-print* versions that can represent serious challenges for automatic processing. For instance, but not limited to, a large number of dictionaries are exported and available as PDF files. PDF stands for Portable Document Format (PDF) which is the *de facto* and *de jure* exchange format that still causes computational headaches and remains an active research topic especially for extracting text and its typography from the original documents (Tiedemann, 2014; Thaiprayoon and Haruechaiyasak, 2016).

<p>able /'eɪb(ə)/ <i>adjective</i> 1. □ to be able to do something to be capable of something or have the chance to do something ○ <i>They weren't able to find the house.</i> ○ <i>Will you be able to come to the meeting?</i> 2. good at doing something, or good at doing many things ○ <i>She's a</i></p>	<p><i>deputy took over.</i> ○ <i>In the absence of any official support, we had to raise our own funds.</i></p> <p>absent /'æbsənt/ <i>adjective</i> not there ○ <i>Ten of the staff are absent with flu.</i></p> <p>absolute /'æbsəlu:t/ <i>adjective</i> complete or total</p>
--	--

<p>absolutely</p> <p>absolutely <i>adverb</i> 1. /'æbsəlu:tli/ completely ○ <i>I am absolutely sure I left the keys in my coat pocket.</i> 2. /,æbsə'lutli/ yes, of course ○ <i>Did you build it yourself? – Absolutely!</i></p>	<p>2</p>	<p>accompany</p> <p><i>little gift.</i> 2. to say 'yes' or to agree to something ○ <i>She accepted the offer of a job in Australia.</i> ○ <i>I invited her to come with us and she accepted.</i> (NOTE: Do not confuse with except.)</p>
--	-----------------	---

(a) The original dictionary

```
absolute absolute /bsəlut/ adjective com-plete or to
tal Basic.fm Page 1 Friday, January 16, 2004 3:10 P
absolutely 2
accompany absolutely absolutely adverb 1. /bsəlutli/
com-pletely I am absolutely sure I left the
keys in my coat pocket. 2. /bsə lutli/
yes, of course Did you build it your-
self? – Absolutely!
```

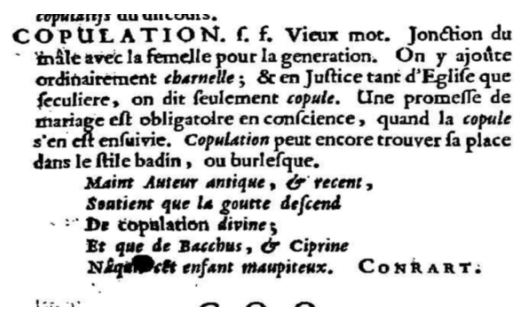
(b) Metadata added to the raw text of the dictionary

FIGURE 3.1: Metadata text introduced by a PDF engine in the text of dictionary pages (Publishing, 2009)

Figure 3.1 shows an example of anomalies that can be introduced by a PDF engine. In this case, metadata information has been found in almost all the pages of the dictionary but it is only seen in the raw text when extracted from the PDF file.

Processing digitised dictionaries is, in its turn, facing more complexity, not only because of the different compilation workflows and OCR systems, but also given the state of the raw documents. Unclear scans resulting from

damaged pages or low resolutions, obsolete fonts, old scripts and old orthography among many other issues, challenge the most advanced digitisation techniques.



(a) The scan in PDF format

```

COPULATION. $. f. Vieux mot. Jonction d
âle avec la femelle pour la generation. On y ajoutê
ordinairement charnelle ; & en Justice tant d'Eglise que
eculiere, on dit seulement copule. Une prome$$e de
ariage est obligatolre en con$$cience, quand la copal.
s'en est ensuivie. Copulation peut encore trouver sa plac
dans le $tile badin, ou burle$que
aim Auieir antique, & recent,
Sontient que la gourte de$$cerns
be copaladon divine :
Et qe de Bacchns, & Ciprine
lrs ees enfant maupiteux. Conrart

```

(b) OCR output by Transkribus (Kahle et al., 2017)

FIGURE 3.2: Digitised entry in Basnage dictionary (Furetière, 1701)

An OCR software sometimes puts spaces in the middle of words, and it can incorrectly recognise an individual letter in a word, such as misrepresenting the letter "T" for the letter "i." or the sequence "rn" for the letter "m". Correctly and consistently identifying typography used to markup microstructure in dictionaries also remains questionable for state of the art OCR systems, such as OCRpus (Breuel, 2008), Transkribus (Kahle et al., 2017), Calamari (Wick, Reul, and Puppe, 2018), or ABBYY Finereader². In Figure 3.2 we can clearly notice many text recognition errors in the OCR output for a legacy dictionary (Furetière, 1701). We see here an illustration of the impact of the scan quality and the old orthography on the digitisation outcome.

Several studies in the literature have investigated the impact of noisy data on basic downstream NLP tasks such as sentence boundary detection, tokenisation, and POS tagging (Lopresti, 2009) as well as more advanced computational applications like topic modelling (Mutuvi et al., 2018).

This digitisation issue represents a serious obstacle for any parser aiming at decoding the logical system behind the typography conventions implemented in a dictionary, as we explain in the following section.

²<https://www.abbyy.com/en-eu/finereader/>

3.2.2 Logical Structure vs Physical Structure

Dictionaries as a whole and their lexical entries are highly structured objects embedding recurrent components. The organisation of a print dictionary can be analysed from two related perspectives: *logical* and *physical* structures

Logical Structure

The logical structure of a dictionary represents the functional constructs and their mutual connections, enabling the representation of the lexical information designed by the creator of the lexicon.

Semasiological dictionaries for alphabet-based languages share a general logical organisation. Such a dictionary is usually segmented into chapters where each chapter contains a list of *dictionary articles*, also called *lexical entries*, starting with the same letter and ordered alphabetically. In bilingual dictionaries, it is common to have chapters organised by pairs of languages.

Lexical entries are considered the elementary constructs of semasiological dictionaries and therefore it is important to define their key components to understand their logical representation and the links between them. Each entry contains a range of predetermined possible constructs that may differ, depending on the entry type and the purpose of the dictionary. It is beyond the scope of this thesis to give an exhaustive list of the possible lexical elements in lexical entries in all kinds of dictionaries. But to give some indication of the complexity of logical structures in lexical entries it is useful to sum up these constructs in the following items:

- *headword*: called also *lemma*. This is considered to be the key element in a dictionary article which is the subject of the lexical or encyclopaedic description and represents a written or spoken form. It usually occupies the first position in the lexical entry and marked with bold characters.
- *variant form*: a variant form of a headword. For instance, "color" is the American *variant form* of a headword "colour" in a British dictionary
- *inflected form*: a form resulting from the application of a regular or irregular inflection paradigm (i.e. pattern) to an original word. For example, "colored" is the *inflected form* of "color" representing its past participle.
- *part of speech (POS)*: grammatical category of a word that may be a verb, noun, adjective, preposition, etc
- *morphological information*: any kind of information related to the morphology of a word form such as orthography, variant and inflected forms, etc. It describes the headword as well as its related forms.
- *grammatical information*: carries the grammatical description of a form such as the POS, gender, number, etc.

- *sense*: represents one possible meaning of the headword and is expressed through definitions, examples, usage domain, etc. A sense could carry a multi level nesting of sub-senses and may replace the function a lexical entry in some dictionaric designs (see Figure 3.3)
- *etymology*: provides information about the origins of the headword and its senses. Etymology may figure as a brief description in several possible spots within the dictionary article, or it can occupy the whole focus of the lexical entry's description in the case of etymological dictionaries.
- *translation equivalent*: maps a headword to its equivalent in another language and can be accompanied by definitions or examples. Such a description is the core representation in bilingual dictionaries.
- *related entry*: also called *compound* or *Multi Word Expression* (MWE). It represents the use of the headword or one of its senses in more complex constructs such as idioms, collocations, etc.
- *cross references*: are links to other lexical entries having a semantic relation such as synonymy or antonymy and are usually triggered by the use of "see X" or "cf. X" or their equivalents in other languages.

act /ækt/ *noun* **1.** something which is done ○ *He thanked her for the many acts of kindness she had shown him over the years.* **2.** a part of a play or show ○ *Act 2 of the play takes place in the garden.* **3.** a short performance ○ *The show includes acts by several young singers.* **4.** a law passed by Parliament ○ *an act to ban the sale of weapons.* **verb** **1.** to do something ○ *You will have to act quickly if you want to stop the fire.* ○ *She acted in a very responsible way.* □ **to act as someone or something** to do the work of someone or something ○ *The thick curtain acts as a screen to cut out noise from the street.* **2.** to behave in a particular way ○ *She's been acting very strangely.* ◇ **to get your act together** to organise yourself properly ○ *If they don't get their act together, they'll miss their train.*

abuse¹ /ə'bjʊ:z/ *noun* **1.** rude words ○ *The people being arrested shouted abuse at the police.* **2.** very bad treatment ○ *the sexual abuse of children* ○ *She suffered physical abuse in prison.* (NOTE: [all senses] no plural)

abuse² /ə'bjʊ:z/ *verb* **1.** to treat someone very badly, usually physically or sexually ○ *She had been abused as a child.* **2.** to make the wrong use of something ○ *He abused his position as finance director.* **3.** to say rude things about someone ○ *The crowd noisily abused the group of politicians as they entered the building.*

FIGURE 3.3: Left: Entry ACT in (Publishing, 2009). Right: Entries ABUSE (Publishing, 2009)

In Figure 3.3 the red, green, purple and pink fields represent respectively the headword, pronunciation, POS and related entry. The article **act** is a homograph which has been represented in this dictionary as a lexical entry having one sense as noun and another one as a verb (the blue fields). Each sense has sub-senses and the second has an embedded related entry. This representation may also be interpreted as a noun entry having a sub-entry as a verb. In both cases, the sense representation differs from the logical pattern followed to model another homograph in the same dictionary, **abuse**, which

could also be a noun and a verb. The logical structure of morphological and grammatical information remains the same, however.

The highlighted inconsistency in modelling entries is one of a number of phenomena that can be observed in modern dictionaries and is more frequent in legacy materials. The logical structure, or the syntax, of lexical entries translated in the presence of certain information or their order may totally change in a different dictionary. The difference depends on the multilinguality, lexicographic and time dimensions introduced in Chapter 2.

Figure 3.4 illustrates a high degree of diversity in the structures of entries from different dictionaries. Such internal and external variations represent a true challenge for having a unified parser that covers all these varieties.

Physical Structure

The logical structure of entries is translated in print dictionaries through a series of navigational components and markers such as font, font size and symbols (e.g. squares, bullets, diamonds). A fixed and predictable typographical system is essential to allow dictionary users to quickly and easily find the information they are looking for. The same applies to a parser which will decode the syntax of lexical entries based on their markup system.

Such a structure remains relatively preserved in the case of the same born-digital dictionary. But given the highly probable deformation of both the text and its typography in digitised documents, explained in Section 3.2.1, the physical structure would be greatly impacted. Consequently, features reflecting the typography of the lexical components would be weak, and in many cases, biasing for a parser aiming at predicting the labels to be assigned to the tokens of lexical constructs.

For both, digitised and born-digital dictionaries, the physical structure depends on the purpose of the lexicographic document as well as the choices of the lexicographer who will have a certain background, practices and typography preferences. Therefore, the combination of all these factors yields a wide range of physical representations of the lexical information (see Figure 3.4) which challenges their uniform parsing.

3.3 State of the art

The problem of parsing the structure of print documents, including dictionaries, has been addressed in the literature by different approaches. In this section, we provide an overview of the commonly used techniques. We put the focus on the most advanced methods which have made use of machine learning to solve the parsing task as well as similar tasks.

3.3.1 Rule-based

Rule-based techniques are very common in computational fields. They have been popular since the early use of computers as, for instance, they had been used for building early expert systems (Buchanan and Duda, 1983). Such a

act /ækt/ *noun* **1.** something which is done ○ *He thanked her for the many acts of kindness she had shown him over the years.* **2.** a part of a play or show ○ *Act 2 of the play takes place in the garden.* **3.** a short performance ○ *The show includes acts by several young singers.* **4.** a law passed by Parliament ○ *an act to ban the sale of weapons* ■ *verb* **1.** to do something ○ *You will have to act quickly if you want to stop the fire.* ○ *She acted in a very responsible way.* □ **to act as someone** or **something** to do the work of someone or something ○ *The thick curtain acts as a screen to cut out noise from the street.* **2.** to behave in a particular way ○ *She's been acting very strangely.* ◇ **to get your act together** to organise yourself properly ○ *If they don't get their act together, they'll miss their train.*

(A) Lexical,
Monolin-
gual (Pub-
lishing,
2009)

ASUMÉ (h) n.4, pl. *mesumé* (vb *sumé* h). Manque, privation. *Asumé bizi, byôm*, manque de vivres, de richesses. Syn. : *nsumga*.

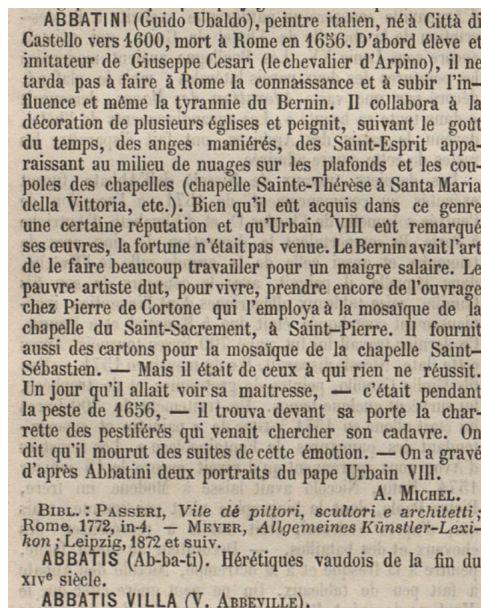
ASUMGA (b) n.4, pl. *mesumga* (vb *sum* b). Commencement, cause originelle. Syn. : *atargé*. Proverbe : *Asumga mvê, asughla abi*, au début tout est beau, à la fin ça ne va plus.

ASUNZOGHE (hh) n.1, pl. *basunzoghé* (visage d'éléphant). D'abord, le premier, avant-garde, devant. *Asunzoghé, vakh me bizi*, tout d'abord, donne-moi à manger. *Asunzoghé a zam bia yia ye bo*, la lère chose à faire. *É bôr asunzoghé*, ceux de l'avant-garde. *Bedu akokh*, ceux de l'arrière-garde, ceux qui ferment la marche. Syn. : *ôsusua, ôsu, bia*, (trompe d'éléphant).

(C) Bilingual (Galley,
1964)

center, centre, n. — F. *centre*, fr. L. *centrum*, fr. Gk. κέντρον, 'point, prickle, spike, ox goad, point round which a circle is described', from the stem of κέντειν, 'to prick, goad', whence also κέντωρ, 'a goader, driver', κέστος (for *κεντός), 'embroidered', κέστρον, 'pickaxe', κόντος, 'pole', fr. I.-E. base **kent-*, 'to prick', whence also Bret. *kenr*, OIr. *cinteir*, 'a spur', OHG. *hantag*, 'sharp, pointed', Lett. *sīts*, 'hunter's spear', *situ, sist*, 'to strike', W. *cethr*, 'nail'. Cp. **centrifugal, centripetal, concentrate, eccentric, Dicentra, paracentesis**. Cp. also **cestrum, cestus, 'girdle, kent, 'a pole, quant, 'a pole'**.
Derivatives: *center, centre*, intr. and tr. v., *center-ing, centr-ing, centre-ing, n.*

(B) Etymological (Ernest,
1966)



(D) Ency-
clopaedic (Berthelot,
1886)

FIGURE 3.4: Excerpts from different dictionaries with different logical structure

method relies on deterministic rules defined by observing patterns in a data sample. The rules are basically conditional instructions:

if condition → **do action**

The set of defined rules constitutes a *grammar* that gives a formal description of certain patterns within the observed sample. A parser relies on such

a grammar to perform the analysis of text structures.

Many ad-hoc rule-based approaches are still being used today to parse dictionaries using tailored grammars. In fact, one way (Khemakhem et al., 2009; Fayed et al., 2014) is to rely mainly on textual markers in the lexical description to infer the function of the parsed structures. Such an approach is commonly used when typography information do not exist and the text of the digitised or retyped dictionary is the only available material to analyse, as is the case for several MRDs.

Other studies (Mykowiecka, Rychlik, and Waszczuk, 2012; Maxwell and Bills, 2017; Steingrímsson, 2018) make use of the typography information, collected from the OCR output, combined with the textual markers to build the parsing grammar. Old but efficient platforms for a flexible writing grammar for new parsers are still in use, such as for example the LexParse tool (Hauser and Storrer, 1993; Lemnitzer and Kunze, 2005).

On the one hand, such an approach has several attractive sides, especially for small projects. First, the rules could be quickly implemented, or just written in the case of existing platforms like LexParse, and results could be obtained within a few days or weeks. Second, the system can be considered as a simulator of the decision process of an expert, but on a larger scale. This fact makes the interpretation of the results straightforward by humans and consequently the rules easy to fine-tune. It also represents a major upside for independent lexicographers who have limited little IT skills, as it frees them from the intervention of IT staff in their own workflow. Finally, since no advanced IT knowledge is needed to develop such a technique, such a choice can drastically reduce the costs for limited budget projects.

On the other hand, the above-mentioned studies have shown that such an approach can be useful to parse dictionaries with a flat structure, bilingual dictionaries for instance, or the shallow constructs in relatively more complex dictionaries. It remains, however, limited for parsing deep and extensive lexical descriptions where the lexicographic information becomes more complex along with possible inconsistencies in the representation of the logical and physical structures. For instance, the dictionary article *FRANCE* in *La grande encyclopédie* (Berthelot, 1886) has 91 pages, *AIR* has 52 pages and *FRANCO-ALLEMANDE (GUERRE)* 31 pages. Any rules defined by humans are too subjective to cover all the patterns hidden in the body of such large lexical entries. The scalability question is not limited to large or legacy samples. The adaptation of the extracted rules to new samples may be costly and in many cases impossible as the dimensions of the new dictionary can vary widely.

3.3.2 Probabilistic Models

The need for scalable methods to parse print dictionaries has become clearly apparent. The use of machines can go beyond simply looking for patterns defined by a human expert. *Probabilistic models* have shown in the literature a great potential to leverage the capacities of *machine learning* techniques when human observation is limited or impossible.

In this section we introduce a family of probabilistic models gathering certain techniques suitable for our target parsing task.

Text Sequence Labelling

The parsing task we are addressing may be seen as the task of assigning a label from a set of possible tags to each token in a sequence of text. Let us assume we want to label this excerpt of an entry "act /akt/ noun 1. something which is done". A probabilistic model has the aim of predicting the probability of assigning a label to each token based on a learnt *distribution* from previously seen data. A label starting with "I-" marks the beginning of a new field in the sequence, whereas other labels (i.e. "<def>") mean we are in the middle or end of a field. In this example: <lemma> marks a lemma, <pron> pronunciation, <pos> part of speech of the headword, <num> numbering tokens and <def> definition of a sense.

<I-lemma> <I-pron> <I-pos> <I-num> <num> <I-def> <def> <def> <def>
 act /akt/ noun 1 something which is done

FIGURE 3.5: Text Sequence Labelling

Such a *text sequence labelling* task is common in the NLP field, where *POS Tagging* and *Named Entity Recognition* (NER) are among the most popular tasks that have been addressed in the literature by applying a family of probabilistic models called *graphical models*.

Graphical Models for Sequence Labelling

The key assumption in graphical modeling is that a distribution over many variables can often be represented as a product of local functions that each depends on a smaller subset of variables. This factorisation shows a close connection to certain conditional independence relationships among these variables. The *graphical* appellation comes from the fact that both types of information can be represented by a graph.

These models describe the probability distribution P based on X and Y , where X and Y are random variables respectively ranging over observation sequences and their corresponding label sequences. The probability distribution P is calculated differently from one graphical model to another and the difference lies in how the independence among variables is modelled.

In this section, we present the conceptual difference between two types of graphical models which have been the most widely used in the literature for sequence labelling: *Hidden Markov Models* (HMMs) and *Conditional Random Fields* (CRFs).

Hidden Markov Models (HMMs) An HMM is a generative model for describing a probability $p(y_{1:N}, x_{1:N})$ over observation $x_{1:N}$ and label $y_{1:N}$ sequences. It is defined as the product:

$$p(y_{1:N}, x_{1:N}) = \prod_{n=1}^N p(y_n | y_{n-1}) p(x_n | y_n)$$

(3.1)

where $p(y_n | y_{n-1})$ denotes the transition between two successive states and $p(x_n | y_n)$ represents the distribution of an observation given its state.

To estimate such a *joint* probability distribution, a HMM needs to enumerate all possible observation sequences for tasks where an observation typically represents an atomic entity, a word in a document or a token in a sentence. Theoretically, such a model can not take into account features of each observation or those of neighbouring entities.

To calculate, for instance, the probability $p(y_3, x_3)$ of the sequence depicted in Figure 3.5, a typical HMM considers the token "noun" as an observation and the label "I-<pos>" as its state. The label and its transition are taken into account but no typographic features (e.g. bold, italic, etc) of "noun", "1.", "/akt/" or any token in the sequence can be considered.

Conditional Random Fields (CRFs) Conditional Random Fields are also a probabilistic framework for labeling a sequence of observations. Whereas an HMM is *generative* assuming independence among input sequence $x_{1:N}$, a CRF tries to relax this independence relationship by describing a *conditional* probability distribution $p(y_n, x_n)$ defined as:

$$p(y_{1:N} | x_{1:N}) = \frac{1}{Z} \exp \left(\sum_{n=1}^N \sum_{i=1}^F \lambda_i f_i(y_{n-1}, y_n, x_{1:N}, n) \right)$$

(3.2)

where x_n is an observation and y_n is a state which respectively belongs to the $x_{1:N}$ and $y_{1:N}$ sequences. f_i denotes an arbitrary set of feature functions and $\lambda_{i:F}$ are their associated *parameters* to be learned. The scalar Z is a normalisation factor to make $p(y_{1:N} | x_{1:N})$ a valid probability over label sequences. Z is defined as

$$Z = \sum_{y_{1:N}} \exp \left(\sum_{n=1}^N \sum_{i=1}^F \lambda_i f_i(y_{n-1}, y_n, x_{1:N}, n) \right)$$

(3.3)

which has an exponential number of terms, difficult to compute in general. Note that Z implicitly depends on $x_{1:N}$ and the parameters λ .

A CRF model removes the constraint that observations only depend on the hidden states in the same event. In fact, CRF provides the ability to model how observations affect each other. Such an ability is enabled through the $f_{i:N}$ functions, also called *feature functions*, which look at pairs of adjacent states y_{n-1} and y_n , the set of input $x_{1:N}$, and where we are in the sequence.

For example, we can define a simple feature function for the text sequence in Figure 3.5 which produces binary values: it is 1 if the current token is "noun", and 0 if the current state y_n is "I-<pos>":

$$f_1(y_{n-1}, y_n, x_{1:N}, n) = \left\{ \begin{array}{ll} 1 & \text{if } y_n = I- \langle pos \rangle \text{ and } x_n = noun \\ 0 & \text{otherwise} \end{array} \right\}$$

(3.4)

A feature function is used depending on its corresponding weight λ_1 . If $\lambda_1 > 0$, whenever f_i is active (i.e. we see the token "noun" in the sentence and its corresponding tag is "I-<pos>"), it increases the probability of the tag sequence $y_{1:N}$. This is another way of saying "the CRF model should prefer the tag I-<pos> for the token "noun". If on the other hand $\lambda_1 < 0$, the CRF model will try to avoid the tag "I-<pos>" for "noun". And when $\lambda_1=0$, this feature has no effect.

To define λ_1 one may set λ_1 by domain knowledge (most probably being positive), or learn λ_1 from corpus (let the data tell us), or both.

As another function, let us consider:

$$f_2(y_{n-1}, y_n, x_{1:N}, n) = \left\{ \begin{array}{ll} 1 & \text{if } y_{n-1} = I- \langle pron \rangle \text{ and } x_{n+1} = 1 \\ 0 & \text{otherwise} \end{array} \right\}$$

(3.5)

This feature function is active if the previous tag is "I-<pron>" and the following token is "1". One would therefore expect a positive λ_2 to go with the feature. Consequently, f_1 and f_2 can both be active for a sequence as in Figure 3.5. This is an example of overlapping features describing the observations and states of a sequence.

The input sequence $x_{1:N}$ is not limited to tokens. $x_{1:N}$ can be the font of token or an attribute marking if a token is bold or not, italic or not etc. In practice, such a representation would be possible by generating for each token (i.e. observation) a vector containing the identity of the token along with its typographic description. Defining a feature function would then consist of selecting attributes from the vector representing the current observation. In such a setup, a feature can, for instance, be:

$$f_3(y_{n-1}, y_n, x_{1:N}, n) = \left\{ \begin{array}{ll} 1 & \text{if } y_n = I- \langle pos \rangle \text{ and } x_n = ITALIC \\ 0 & \text{otherwise} \end{array} \right\}$$

(3.6)

Such a feature is active when the current tag is I-<pos> and the current font is italic.

At this point, it becomes clear how much more flexible CRFs are than HMMs for defining candidate features extracted from the observation and state sequences. Such a flexibility gives the model the ability to use the different hints about the physical structure of dictionary articles to structure their text sequences.

Evaluation metrics

The performance of a model (i.e. tagging algorithm) is usually evaluated using a variety of different metrics, measured against gold standard data which have been tagged by human annotators and are known to be correct. To calculate them, it is necessary to count the different possible decisions:

- True Positives (TP), i.e., positive instances correctly predicted
- True Negatives (TN), i.e., negative instances correctly predicted (the target label has not been used to tag wrong tokens)
- False Positives (FP), i.e., instances which are tagged with a target label where it should not be
- False Negatives (FN), i.e., instances which are not tagged with the target label but should be

Precision reports how many of the model's decisions to tag a token are correct, i.e. the higher the precision of a tagging algorithm is, the more confident we can be about the labelling of tokens with the current tag. Precision is formally defined as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall reports how many of the positive examples in the gold standard are found by our algorithm, i.e. the higher the recall of our algorithm the more confident we can be that all tokens that should have the target label are correctly tagged. Recall is formally defined as:

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score (or F-measure) represents a weighted average of precision and recall. It is usually considered as the most significant metric, as precision and recall are not useful in isolation. These two measures can be considered antagonistic: good precision might be achieved by tagging few but correctly (no incorrect prediction has been made), while perfect recall can be achieved by tagging everything with the target label. The F-measure is defined as:

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Accuracy reports how many of the decisions made by the algorithm are correct in total, i.e. considering both positive and negative examples. While this is also an indicator of tagging quality, it should be carefully judged depending on the dataset. It is considered as a good measure only when we have balanced datasets where the values of false positives and false negatives are almost same. Otherwise, a good accuracy can easily be achieved by always assigning the majority class. Thus, the F1-score is usually considered to be the more meaningful measure. Accuracy is formally defined as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Token/Field level Token, or word, level reports the labelling quality for each different token, where field, also called phrase, level evaluates the quality over a whole sequence of tokens. Field level quality is usually harder to achieve as one wrongly labelled token in a sequence is enough to consider the labelling has failed for the whole sequence.

Previous Work

There are not too many studies in the literature that have made use of machine learning for the purpose of parsing dictionaries. The few relevant endeavours have carried out experiments using mainly one of the two categories of the previously introduced graphical models: HMMs and CRFs.

HMM-based

Approach: Ma et al., 2003 presented a system for the acquisition and parsing of lexical information in print bilingual dictionaries. Processing of the content of an OCR'd dictionary page is carried out in two stages before the generation of a structured lexicon. The first step called "Dictionary Parsing" has the task of extracting each entry and marking its main fields (i.e. functional properties of a group of tokens). It is broken into 3 phases:

- *Dictionary acquisition:* by adjusting the setup for scanning and storing dictionary page scans
- *Entry Segmentation:* the step addresses the automatic identification of the starting and ending lines of each entry based on textual and typographic features (e.g. special symbols, word font and size, indentation, spacing..) that are manually extracted and then used to train a Bayesian framework. The process is iterative and relies on a human in the loop to provide corrections of the results to generate better samples for re-training the framework. The workflow yielded accurate segmentation results (between 96% and 99% accuracy)
- *Functional Labelling:* Each identified entry in the previous phase is processed at this level to differentiate fields containing Latin scripts from those that do not. Such fields, called "functional", can be pronunciation or translations, etc. A Nearest Neighbor Matching (Ma and Doermann, 2003) and Support Vector Machine (Borges, 1998) classifiers have been experimented to identify these functional fields. The Support Vector Machine classification has been reported to perform better (above 90% average accuracy).

The second step comes after applying an OCR system and is dedicated to tagging entry tokens, which represents the previously explained sequence labelling task. Ma et al., 2003 experimented two approaches:

- *Rule-based*: the approach is very similar to those presented in Section 3.3.1 and based on observing the lexical description typography to tune a tagger's grammar incrementally. Experimented with dictionaries with highly regular structures, this approach has been reported to work "efficiently".
- *HMMs-based*: for dictionaries with relatively more complex structures and noisy OCRs, HMM models have been used to leverage the joint distribution of features in the dictionary data. Karagol-Ayan, Doermann, and Dorr, 2003 give an exhaustive description of the same approach which has been used to parse similar bilingual dictionaries (see Figure 3.6). Manual annotation has been carried out for 400 randomly selected tokens to be tagged with their functional categories. For the training of the HMM, seven features have been used: (i) Content: Category of the keyword (keyword, special symbol, number, and NULL otherwise) (ii) Font: Font style of the token (normal, bold, italic, etc.). (iii) Starting symbol: marks if the token begins with a special punctuation, NULL otherwise. (iv) Ending symbol: marks if the last character of the token is a special punctuation, NULL otherwise. (v) Second ending symbol: marks if the second to last character of the token is a special punctuation, NULL otherwise. (vi) Is-first token: True if the current token is the first token of an entry, false otherwise. (vii) Is-Latin: True if the characters in the token are Latin-based characters, false otherwise.

Given HMMs's limitations to model such features for each observation (see Section 3.3.2), the implemented HMM framework (DeMenthon and Vuilleumier, 2003) allows through a workaround each token to be converted to a vector of the seven features then tagged with the corresponding category. The predicted states are later mapped to the original tokens.

<p>از <i>U</i> (n. ac. 1), Drove away.—(b), Kindled.—(c), Inivit eam.—VIII, Hastened.—(b), Was excited.—(c), see I (c). 2, Fire.—(b), Thunder-bolt. آرب <i>I, U</i> (n. ac. 1), Adjusted, set (necklace).</p>	<p>latab <i>n</i> name given to young <i>samuk</i>: <i>Gerres spp.</i> látab <i>v</i> [B6; b] for liquids to have oil, usually edible, floating on top. <i>Ang sabaw nag-látab sa mantiká</i>, The soup has streaks of oil floating on top of it. látab <i>v</i> [A13] for liquor to be present in inexhaustible quantities. <i>Maglátab ang tubá sa ámu maduminggu</i>, The toddy simply</p>	<p>【大冤案】 gross injustice 【大元帅】 generalissimo 【大员】 [旧] high-ranking official; 委派 ~ appoint high-ranking officials 【大圆航向】 [航空] great-circle course 【大院】 courtyard; compound; 居民 ~ residential compound 【大约】 ①(约略) approximately; about ②(很可能) probably</p>
Arabic-English	Cebuano-English	Chinese-English
<p>brassin [bra'sē] <i>m</i> brew; mash-tub. brasure [bra'zy:r] <i>f</i> brazed scam; hard solder(ing). bravache [bra'vaʃ] <i>1. su./m</i> bully; swaggerer; <i>2. adj.</i> blustering, swaggering; bravade [ˈva'd] <i>f</i> bravado, bluster; brave [bra:v] brave; good, honest; <i>F</i> smart; <i>un</i> ~ <i>homme</i> a worthy man; <i>un homme</i> ~ a brave man; <i>F faux</i> ~ see <i>bravache 1</i>; braver [bra've] (1a) <i>v/t.</i> defy; brave;</p>	<p>आगम <i>ā-gam</i> [S.], <i>m.</i> 1. coming, approach; entry; appearance. 2. the future, the hereafter. 3. a sacred text, esp. a Veda; a text containing spells and incantations; a <i>tantra</i>. 4. document, deed. 5. income. — ~ <i>वाचना</i>, to determine the future, to foretell; to plan for the future. ~ <i>वात</i>, <i>f.</i> prophecy. — आगम-पत्र, <i>m.</i> title-deed. आगम-बन्ध, <i>m.</i> inv. one who foretells the future; an astrologist. आगम-शुल्क, <i>m.</i> customs or import duties.</p>	<p>a.cross (ikrós') <i>z.</i>, <i>edat</i> ortasından, içinden veya üstünden karşı tarafa geçerek; <i>edat</i> çaprazvari, öbür tarafa, karşı yakada, come across rast gelmek, tesadüf etmek; <i>k. dili</i> görünmek. come across with k. dili istemeyerek vermek. a.cros.tic (ikrós'tik) <i>i.</i> akrostiş. a.cryl.ic (ikril'ik) <i>i.</i> sıcakken yumuşak olan plastik.</p>
French-English	Hindi-English	English-Turkish

FIGURE 3.6: Example of bilingual dictionaries processed by (Ma et al., 2003) and (Karagol-Ayan, Doermann, and Dorr, 2003)

On the token level, both studies (Ma et al., 2003; Karagol-Ayan, Doermann, and Dorr, 2003) have reported comparable F1-scores (around 0.75 for French-English) and (around 0.87 for English-Turkish) using rule-based and HMM-based parsers. On the field level, the former performed clearly better (around 0.75 for French-English and around 0.88 for English-Turkish) than the stochastic approach (around 0.70 for French-English and around 0.77 for English-Turkish). Karagol-Ayan, Doermann, and Dorr, 2003 reported lower stochastic performance with a Hindi-English sample on the field level (0.51 F1-score). They tried to boost the results by applying a rule-based post-processing to the stochastic output. The hybrid method gave relatively better results but remained limited for certain samples.

Discussion: Ma et al., 2003 succeeded in the design of a pipeline comprising several modules implementing different techniques to parse *bilingual* dictionaries using ad-hoc and adaptive methods. The attempt to implement a scalable technique with a human in the loop to adjust the entry extraction results of a Bayesian system was successful. However, the random selection of tokens to build a HMM model, is questionable for the sake of learning labelling lexical *sequences*, even after tweaking the HMM modelling setup.

The HMM-based approach presented for the identification of salient logical structures in bilingual lexical entries can not be reliable for a scalable processing, given the limitation by design of such models.

CRF-based

Approach 1: The exhaustive study carried out by Crist, 2011 can be considered the most relevant work to this part of the thesis. He first explained the conceptual complementarity between *logistic regression* and HMMs for the purpose of modelling observations and their hidden states in a sequence labelling setup. This analogical study, also shared by several introductory studies to graphical modelling (Zhu, 2010; Sutton and McCallum, 2012), has been used to justify his choice for CRFs, which is considered a combination of the best of both techniques, to parse two digitised dictionaries.

The first sample was a bilingual Lau-English dictionary and has relatively few and simple structure entries. All of the 1365 entries of the dictionary, containing over 15 000 tokens, were tagged with 10 different tags marking (i) headwords, (ii) integers distinguishing homographs, (iii) morphological category, (iv) indentation, (v) single letter headers, (vi) use of the headword in a context (vii) English definitions, (viii) abbreviations indicating POS or other morpho-syntactic categories, (ix) cross references and (x) English prose discussions. For each token, a set of 24 features is generated to describe the typography (e.g. bold, italic) and the category of the current token, part of it or preceding token (e.g. final character is semi colon or not, its preceding token ends with a comma, etc).

The second sample of the experiment was an old English dictionary with over 1300 pages and 60000 articles. 13 randomly selected pages containing 306 entries and over 20000 tokens were manually tagged with 17 labels. The

new tags give more precision regarding the categories of the logical lexical structure of each token, such as Latin translations, Greek translations, morpho-syntactic information, etymology, etc. Each token was extensively described using 35 features, with more focus on the language, the morphology and belonging of each token to existing vocabularies (e.g. abbreviations used in the dictionary).

The training of the two CRF models was performed through the use of the MALLEET framework (McCallum, 2002). Accuracy was the measure chosen to evaluate the results and behaviour of each model. In the first experiment, a plateau of 95% was reached by using 3100 tokens, where the mean number of tokens per entry is around 10. The second experiment reported 94% accuracy reached by using over 7400 tokens, where an entry contains on average 65 tokens.

The remainder of the work studies in depth the structure of 100 digitised dictionaries and focuses on the crucial steps for digitising and parsing dictionary highlighting the role played by the OCR quality in such a process.

Discussion 1: Through his exhaustive survey, Crist, 2011 managed to expose the diversity and the complexity of the logical and physical structures in digitised dictionaries, and consequently, the need for a flexible parsing framework as CRFs. Empirically, he succeeded in leveraging the modelling capacity of CRFs to machine learn the parsing of two different dictionaries. Moreover, his experiments showcased the difference in complexity between dictionaries and the consequent need for more descriptive features and larger training datasets.

However, the implemented technique remains experimental and focused on parsing already extracted lexical entries, with no practical solutions for identifying entries' boundaries which is a non-negligible issue. The general accuracy measures provided do not give a precise evaluation of the models for parsing the different structures exhaustively analysed. In fact, the large number of labels to infer at once seems to be overwhelming for one model. We also have reservations about the representativity of the randomly selected entries in the training dataset to cover all the structures to be predicted in the evaluation dataset. Regarding the features used, their selection was tailored to the sample, which limits the ability of the resulting models to scale up. Less language dependent along with long-range features (i.e. features describing several tokens preceding and following the current token) could have resulted in a CRFs model that supports both samples with only more training.

Approach 2: Another work presented by Bago and Ljubešić, 2015 has studied the use of CRFs for the purpose of speeding up language and structural annotation in a multilingual legacy dictionary.

The first use case has the goal of learning the inference of a token's language based on its context in the dictionary article. Over 100 entries, comprising around 2% of all the tokens of the dictionary, were randomly selected.

These tokens were annotated with 3 labels indicating the language: (i) Croatian, (ii) Italian, (iii) and Latin, with a domination of Croatian tokens (i.e. over half). A variable set of around 20 features, mostly focused on the text of the token and its neighbouring, was generated for each token. A set of features includes: Boolean variable whether the token is lower-case or not, the surrounding tokens and their lower-case form, and the frequency of the token's sequential trigrams. An incremental experiment was carried out to find the final subset of best features giving the highest accuracy and F1-score: (i) 0.98 for Croatian, (ii) 0.97 for Italian, (iii) and 0.99 for Latin,.

The second use case deals with parsing the lexical structures carried by the tokens by applying the same approach with a slight adjustment of the features and new task-specific labels. Language and suffix of 4 character length were added to the feature set of the incremental experiment and kept in the final subset of informative features. 19 different tags were used to label the data with flat structural category of a token, such as POS, citation, cross reference, bibliography, punctuation, etc. Given the fact that the entries were randomly selected, the performance of the trained CRFs model significantly varied from zero, for non frequent labels in the training dataset, to 0.99 and 1.0 for punctuation and line beginning tags.

The CRFsuite (Okazaki, 2007) has been used to train the CRFs models with the annotated data. The learning curve shows that 40% of the data brings the model close to the plateau, which confirms the findings of Crist, 2011, showing the efficiency of CRFs models to quickly learn the hidden states in dictionary text sequences.

Discussion 2: Through the language labelling use case, Bago and Ljubešić, 2015 have demonstrated the positive impact of long range features on the performance of CRFs models. They have also shown the power of text based features for some specific tasks when there are no structure categories to be identified. However, the second use case has exposed the need to have observations describing the physical structure to infer the logical structures in a dictionary.

Statistics about each label in the training and test batches are missing but it is clear that the dataset is imbalanced for the use case of structural aspects learning. Thus, this is another experiment showing the specificity of lexical data and the need to have lexicographic expertise for the selection of a training sample. Another part of the issue of having low performance over all the labels in the second experiment could be explained by the fact that several heterogeneous labels have to be learned by one model.

The approach is reported to speed up the annotation process but still not reliable for full automatic labelling of lexical constructs, even with a sample that has no lexical depth.

3.3.3 Probabilistic Models for Parsing Bibliographic Data

All the related studies presented so far have followed a bottom-up approach for decoding lexical information in print dictionaries. The reviewed machine

learning approaches have manifested difficulties in capturing heterogeneous lexical structures using one model.

Our study of the literature led us to find a top-down approach for parsing bibliographic information in a category of print documents, representing several similarities with the parsing task we are addressing. In what follows we present this approach implemented in a framework called GROBID.

GROBID (GeneRation Of Bibliographic Data)

GROBID (*GROBID 2008–2020*; Romary and Lopez, 2015) is a machine learning framework that implements a modular approach for the analysis and extraction of *bibliographic* constructs in scientific papers and patent documents.

GROBID has used over 10 CRFs models to orchestrate the parsing of such documents in a cascade fashion. Each model is responsible for the identification of a homogeneous set of structures. At its first extraction level, GROBID detects the main blocks of a paper such as the header, the body, the references, annexes, etc. These main parts are further structured at the following level, like the header which is recognised and parsed in a second stage to extract the title, authors, their affiliations, abstract and keywords. The references are also extracted in separate items and then parsed one by one to detect the titles, the authors and the other publication details.

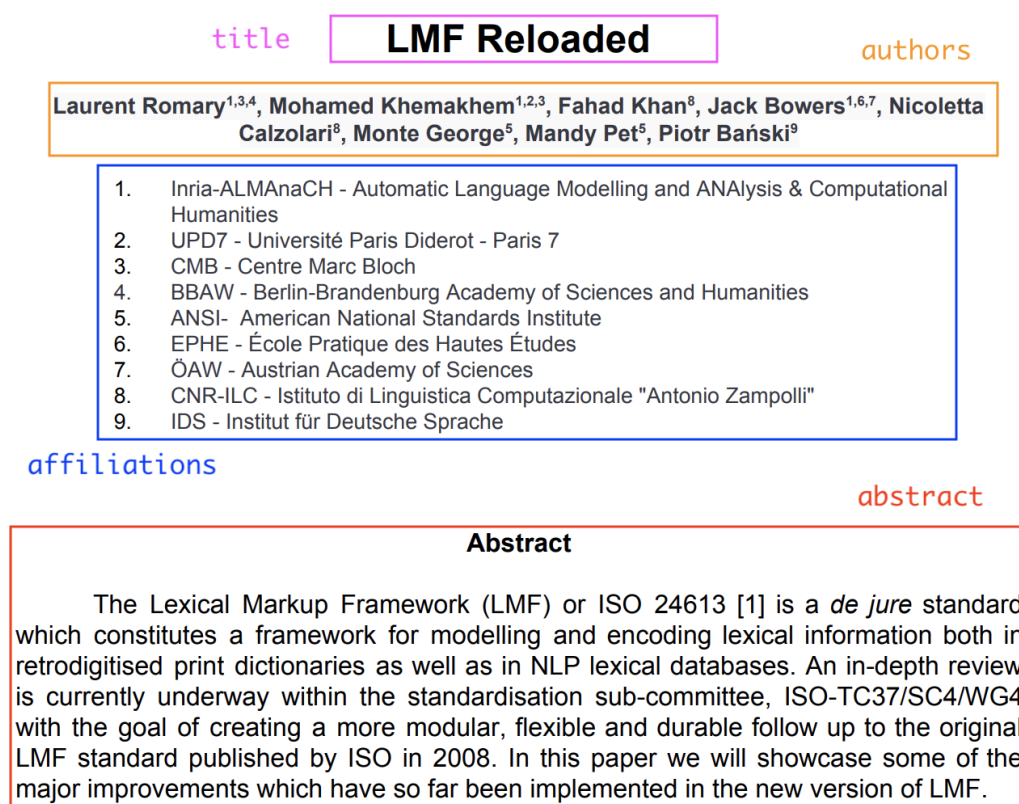


FIGURE 3.7: Illustration of the Hyper-Level zones in a scientific paper recognised by Header Segmentation Model in GROBID

Figure 3.7 depicts an intermediary segmentation level (i.e Header segmentation) linking a general segmentation model and several sub-models for parsing each detected hyper-zone.

GROBID (up to version 0.5.3) relies on an advanced version of CRFs (Lavergne, Cappé, and Yvon, 2010) which allows the modelling of a large number (millions) of features per model. Each model is designed with dedicated features and labels supporting the parsing of a certain level of the bibliographic information. For the first segmentation model, for instance, each line in a document is described with over 30 seed observations³ that are expanded in more local and long range features based on the text, layout and typography.

Such an extensive use of features results in more complex feature engineering and a longer learning process but the outcome of the models shows it pays off. The framework has been ranked for years among the best system for such a task (Saleem and Latif, 2012; Lipinski et al., 2013; Torre et al., 2018; Tkaczyk et al., 2018) with high accuracy information extraction, thanks to its cascading sequence labelling and the leverage of CRFs modelling capacities.

3.4 Chapter Summary

In this chapter, we have shown the bottlenecks for parsing lexical structures in print dictionaries. The difficulties lie mainly in the noisy digital material to process and the complexity and possible inconsistency of the logical structure to dismantle, through physical clues highly impacted by the varying noise in the material.

The study of the literature has shown that it is still of actuality to use ad-hoc approaches (Maxwell and Bills, 2017; Steingrímsson, 2018), based on parsing grammars defined on the basis of human expert observations. HMMs-based methods (Karagol-Ayan, Doermann, and Dorr, 2003; Ma et al., 2003) have tried to relax the dependence on human expertise by relying on probabilistic models to learn the distribution of labelled sequences. The attempt was partly successful as comparable results have been achieved by an adaptive HMMs system for several dictionary samples. However, the approach remained focused on capturing salient structures in a category of resources (bilingual dictionaries) and did not leverage the context of tokens, given the conceptual limits by design of HMMs.

CRFs based approaches (Crist, 2011; Bago and Ljubešić, 2015) have been adopted to overcome the limits of HMMs and provide a scalable framework that integrates noise and uncertainty along with the rich contextual clues that are often present in print dictionaries. The visited CRFs setups have illustrated ways to benefit from the textual and typographic information of tokens in a context. However, the solution for the different sequence labelling tasks suffered from the "one model does it all" strategy. It risks sanctioning certain labels which are less represented in the training dataset. Moreover, the evaluation of such an issue is usually hard given the large number of

³<https://github.com/kermitt2/grobid/blob/master/grobid-core/src/main/java/org/grobid/core/features/FeaturesVectorSegmentation.java>

heterogeneous labels per model. The attempt to reduce labels per model has shown positive results with dictionaries (Bago and Ljubešić, 2015). A larger modular approach (Romary and Lopez, 2015) has also given highly accurate results with a similar complex parsing task, which supports the "divide and conquer" strategy.

We hypothesise that the use of CRFs can be improved and tuned to build models that are generic enough to cover several categories of lexica. Following a waterfall approach for the parsing case of print dictionaries seems to have the potential to solve the issue of overwhelming lexical labels. These assumptions are our starting point to find answers to the genericity and modularity questions.

Chapter 4

Lexical Models for Parsing Print Dictionaries

4.1 Introduction

Previous work introduced in Chapter 3 has left open questions regarding the ability of probabilistic models to parse complex lexical information along with the variations and inconsistencies in different dictionary print material. Parsing print dictionaries using probabilistic models, CRFs more specifically, seems to have the potential to overcome the scalability issues. However, more investigations need to be carried out to find additional assets for using such probabilistic models as well as their limits.

Given the unrealistic short-term goal of structuring all kinds of dictionaries by the click of one button, our aim in this thesis is not limited to a proof of the scalable structuring concept but we also aim at building an end-to-end infrastructure for structuring dictionaries which can be easily adapted and extended. Getting domain experts, who may have limited IT skills, involved in shaping the lexical models is on our agenda for two reasons: first because we do not have the knowledge required to annotate data in several categories of dictionaries in a multitude of languages. Second, we do not have enough resources to perform such large-scale and complex *annotations*, which are the cornerstone for conducting any machine learning experiment.

In this chapter, we present our approach which has been inspired from another task dealing with a different category of text documents. We also give details about building GROBID-Dictionaries¹, a new architecture of machine learning models dedicated to the analysis and extraction of lexical structures from digital and digitised dictionaries, following the novel approach. An overview on the challenges encountered and the solutions to implement our approach, are also discussed in the following sections.

4.2 Cascading Parsing

Lexical information in print dictionaries has several granularity levels which represent the logical structure of the lexicon. Recognising all these elements

¹<https://traces1.inria.fr/grobid-dictionaries/>

at once is not an obvious task as the complexity we want to dismantle involves hierarchies of lexical components and different constructs that do not have the same physical structure. Therefore, we addressed the parsing task by adopting a top-down approach where several parsers are built in cascade to go from the less to the more granular. To be easily adaptable to new dictionary samples, we chose machine learning over rule-based techniques to build the cascading models. And since CRF models have been the more successful in the literature for sequence labelling tasks, we decided to explore them to implement our approach.

4.2.1 Approach

We followed a divide-and-conquer strategy to dismantle text constructs in a print dictionary, based initially on observations of their layout. Main pages (see Figure 4.1) in almost any dictionary share three blocks: a header (green), a footer (blue) and a body (orange). The body is, in its turn, made up of several entries (red).

Each lexical entry can be further broken down (see Figure 4.2) into: form (green), etymology (blue), sense (red) or/and related entry. The same logic could be applied further for each extracted block, as long as the finest lexical structures have not yet been reached.

Such a cascading approach ensures a better understanding of the learning process's output and consequently simplifies the selection process of the machine learning features. Limited exclusive text blocks per level help to diagnose the cause of prediction errors significantly. Moreover, it would be possible to detect and replace at an early stage any irrelevant selected features that can bias a trained model. In such a segmentation, it becomes more straightforward to notice that, for instance, the token position in the page is very relevant to detect headers and footers but has almost no relevance for capturing a sense in a lexical entry, which is very often split over two pages.

Such a generic approach enables the modular creation and the flexible tuning of any required number of models to parse certain constructs. But at the same time, the flexibility should be controlled to avoid ending up with an architecture tailored for one sample or category of lexica. The balance we are aiming to strike should be translated in an architecture of models that is generic enough to be adapted to any dictionary by simply annotating a small sample, a key known asset for CRFs. Therefore, this milestone is highly connected to the modelling decisions and best practices that will be discussed in Chapter 6. Switching between the two processes to adapt either the architecture or the modelling to each other, is necessary to make sure that the outputs of the two stages are compatible.

4.2.2 Bibliographic Information and Lexical Information: Parsing Similarities and Differences

Our study of the literature led us to notice a remarkable analogy between the structures that can be extracted by GROBID (Romary and Lopez, 2015),

CON

Eugène IV à Ferrare en 1438-1439, puis à Florence de 1439 à 1442], de Istran (1512-1517), de Trente (1545-1563) [où fut décidée la réforme générale de l'Église catholique en face de la Réforme protestante], de Vatican I (1870) [où fut défini le dogme de l'Infaillibilité pontificale], de Vatican II (1962-1965) [où fut définie l'attitude de l'Église romaine à l'égard du monde moderne].

conciliarie adj. Qui peut se concilier avec une autre chose.

conciabule [kɔ̃siabyl] n. m. (lat. *conciabulum*). Réunion secrète de personnes soupçonnées de mauvais desseins : tenir des *conciabules*. | Entretien plus ou moins secret et suspect.

conciataire adj. Relatif à un concile : décret *conciataire*.

conciant, e adj. Porté à la conciliation : caractère *conciant*. | Propre à concilier : des paroles *conciantes*.

conciateur, trice n. Personne qui concilie, aime à concilier.

conciliation n. f. Action de concilier ; résultat de cette action. | Accord de deux personnes en litige, réalisé par un juge.

conciatoire adj. Propre à concilier : mesures *conciatoires*.

conciiller [kɔ̃silje] v. t. (lat. *conciillare*). Mettre d'accord ; concilier des adversaires. | — le *conciiller* v. pr. Acquiescer ; laisser se concilier l'estime d'autrui.

concis [kɔ̃si]. e [-s] adj. (lat. *concisus*, tranché). Qui exprime beaucoup de choses en peu de mots. (Syn. : bref, court, dense, *concis*, *succinct*.)

conciision n. f. Qualité de ce qui est concis.

conciroyen, enne n. Qui est du même pays, de la même ville.

conciclave [kɔ̃silav] n. m. (lat. *conclave*). Assemblée de cardinaux pour élire un pape.

conciaviste n. m. Personne qui s'enferme au concile avec un cardinal, pour le servir.

conciuer, e adj. Qui prouve bien ce qu'on a avancé : argument *conciuer*.

conciure [kɔ̃kyr] v. t. (lat. *conciudere*) [con]. 42. Achever, terminer : *conciure* une affaire. | Tirer une conséquence : *conciure* une chose d'une autre. | — V. l. Donner son avis, ses conclusions ; se prononcer : on vous *conciure* de conclure. | Être probant : les témoignages *conciuent* contre lui.

conciusion n. f. (lat. *conclusio*). Arrangement définitif : la *conciusion* d'un traité. | Fin, résultat final : la *conciusion* d'un discours. | Conséquence d'un argument : la *conciusion* d'un syllogisme ne doit pas dépasser les prémisses. | — Pl. Prévisions respectives de chacune des parties dans un procès. | Écrit exposant ces prévisions. | Réquisition du ministre public. | — En *conciusion* loc. adv. En conséquence, pour conclure.

conciuer v. t. Fam. Elaborer avec soin : *conciuer* une lettre de réclamation.

conciombre [kɔ̃kɔ̃br] n. m. (anc. provenç. *conciembre*). Plante potagère de la famille des cucurbitacées, cultivée pour ses fruits allongés que l'on consomme comme légume ou en salade. | Ce fruit.

conciomment adv. De façon concisante.

conciomance [kɔ̃kɔ̃mɑ̃s] n. f. Coexistence, simultanéité de deux ou de plusieurs faits.

conciomitant, e adj. (lat. *conciomitant*). Qui accompagne, qui se produit en même temps : des faits *conciomitants*. • Variations *conciomitantes*, variations simultanées et proportionnelles de certains phénomènes.

conciordance n. f. Conformité, accord ; concordance de témoignages. | Géol. Disposition parallèle des couches sédimentaires. • *Conciordance de phases* (Phys.), état de plusieurs vibrations sinusoïdales de même nature et de même période, dont la différence de phases est nulle. • *Conciordance des temp-râtes* de syntaxe d'après lesquelles le temps du verbe d'une subordonnée varie selon celui du verbe de la principale.

conciordant, e adj. Qui s'accorde : témoignages *conciordants*.

conciordat [kɔ̃kɔ̃rdɑ] n. m. (lat. *conciordatum*). Traité entre le pape et un gouvernement sur les affaires religieuses. | Dr. Accord entre le commerçant qui, ayant déposé son bilan, a été admis par le tribunal de commerce au règlement judiciaire et ses créanciers. | Les plus anciens *conciordats* sont le *conciordat* de Worms (1122), entre Calixte II et Henri V ; le *conciordat* de 1516, entre Léon X et François I^{er}. Le *conciordat* entre Bonaparte et Pie VII, conclu le 10 juillet 1801, a réglé les rapports de la France avec le Saint-Siège, et de l'État avec l'Église jusqu'à la loi du 9 décembre 1905. Au xix^e s. et au xx^e s., de nombreux *conciordats* furent signés par les papes.

conciordataire adj. Relatif à un *conciordat* : loi *conciordataire*. | Dr. Se dit du commerçant qui a obtenu un *conciordat*.

conciorde n. f. (lat. *conciordia*). Accord des sentiments et des volontés : rétablir la *conciorde* entre les citoyens.

conciorder [kɔ̃kɔ̃rdɛ] v. t. (lat. *conciordare*). Avoir des rapports de similitude, de correspondance : dates qui *conciordent*.

conciourant, e adj. Qui converge vers un même point, un même but : droites *conciourantes*.

conciourir [kɔ̃kɔ̃ru] v. t. (lat. *conciourir*) [con]. 21. Tendre au même but, aider à : *conciourir* au succès d'une affaire. | — V. l. Être en concurrence en compétition : *conciourir* pour un prix.

conciours [kɔ̃kɔ̃r] n. m. (lat. *concioursus*). Concidence, conjoncture ; concours de circonstances. | Action de concourir, d'aider : offrir son *conciours*. | Action d'entrer en concurrence avec d'autres, pour prétendre à quelque chose ; examen : se présenter à un *conciours*. | Lutte sportive : *conciours* hippique. • *Conciours général*, *conciours annuel* entre les premiers élèves des classes supérieures des lycées, collèges et écoles normales.

conciort [kɔ̃kɔ̃rt]. e [-t] adj. (lat. *conciortus*). Epais, condensé : huile *conciorte* (vieille). | Qui exprime quelque chose de réel, de positif : obtenir des avantages *conciorts*. | Qui a le sens des réalités précises : esprit *conciort*. | Gramm. Se dit d'un terme qui désigne un être ou un objet pouvant être perçu par les sens. • *Musique concrète*, technique de composition qui utilise les bruits produits par divers objets sonores enregistrés sur bande magnétique et susceptibles de transformation.

conciort n. m. Qualité de ce qui est concret.

conciortement adv. De façon concrète.

conciorter v. t. (conq. 3). Rendre concret, solide.

conciortion [kɔ̃kɔ̃rjɔ̃] n. f. (de *conciort*). Action de s'épaissir : la *conciortion* de l'huile, du sang. | Réunion de parties en un corps solide : *conciortion* saline. | Agrégation solide dans les tissus vivants : *conciortions* bénils.

conciortiser v. t. Rendre concret ce qui est abstrait : *conciortiser* une idée, un avantage.

conciubin, e adj. (lat. *conciubina*). Relatif au concubinage. | — N. Personne qui vit en concubinage.

conciubinage [kɔ̃kybinɑ̃] n. m. État d'un homme et d'une femme qui vivent ensemble sans être mariés. (On dit aussi UNION LIBRE.)

conciupiscence n. f. (du lat. *conciupiscere*, désirer). Penchant à jouir des biens terrestres, particulièrement des plaisirs sensuels.

conciupiscient [kɔ̃kybisjɑ̃]. e [-s] adj. Qui exprime la *conciupiscence* : regards *conciupiscents*. | Attaché aux plaisirs sensuels.

conciurrence [kɔ̃kyrɑ̃s] adv. — *pan* *conciurrence*. | Par un concours mutuel, de concert : agir *conciurrence*ment avec quelqu'un.

conciurrence n. f. Rivalité entre plusieurs personnes qui visent un même but : entrer en *conciurrence* avec quelqu'un. | Rivalité d'intérêts entre commerçants ou industriels qui tentent d'attirer à eux la clientèle par les meilleures conditions de prix, de qualité, etc. • Régime de libre *conciurrence*, système économique qui ne comporte aucune intervention de l'État en vue de limiter la liberté de l'industrie et du commerce, et qui considère les conditions de producteurs comme des délits. | — Jusqu'à concurrence de loc. adv. Jusqu'à la somme de.

conciurrence v. t. (conq. 1). Faire *conciurrence* à.

conciurrent [kɔ̃kyrɑ̃]. e [-ɑ̃] adj. et n. Qui tend au même but : une action *conciurrente*. | Personne qui participe à un concours, à une compétition : les *conciurrents* ont pris le départ de la course. | Celui qui exerce la même profession commerciale qu'un autre.

conciurrentiel [kɔ̃kyrɑ̃sjɛl]. e [-sjɛl] adj. Ou loc. la *conciurrence*.

conciusion [kɔ̃kyzjɔ̃] n. f. (lat. *conciusio*, secousse). Exaction commise par un trésorier public. | Malversation commise dans l'exercice d'une fonction publique, particulièrement dans le manement des deniers publics.

conciusionnaire adj. et n. Coupable de *conciusion*.

conciusable adj. Qui mérite d'être condamné : acte *conciusable*.

conciudation [kɔ̃dɑ̃sjɑ̃] n. f. (lat. *conciudatio*). Décision d'un tribunal imposant à l'un des plaideurs de s'incliner au moins partiellement devant les prétentions de son adversaire. | Décision d'une juridiction prononçant une peine contre l'auteur d'un crime, d'un délit ou d'une contravention. (En cour d'assises, le jury juge la culpabilité de l'accusé, et la cour prononce la *conciudation*.) | La peine infligée : une *conciudation* à la réclusion criminelle. | Bâton, désapprobation : la *conciudation* des abus.

conciudatoire adj. Qui porte *conciudation*.

conciudé, e n. Personne qui a subi une *conciudation*. | — Adj. Qui ne peut échapper à un sort pénal : maladie *conciudée*.

conciudner [kɔ̃dɑ̃nɛ] v. t. (lat. *conciudnare*). Prononcer un jugement contre un plaideur ou un inculpé : *conciudner* un criminel. | Astreindre, réduire à : *conciudner* au silence, à l'immobilité. | Désapprouver, blâmer : *conciudner* une opinion, un usage. | Interdire : la loi *conciudne* la bigamie. | Déclarer perdu, incurable : les médecins l'ont *conciudné*. | Barer, truser : *conciudner* une porte.

conciudensable adj. Qui peut être *conciudé*, réduit à un moindre volume.

conciudensateur n. m. Phys. Appareil servant à emmagasiner une charge électrique : la bouteille de Leyde est un *conciudensateur* électrique. | Lentille servant à éclairer un objet dont on veut former une image.

conciudensation n. f. Action de *conciudner* ou effet qui en résulte. | Liquéfaction d'un gaz. | Soudure de plusieurs molécules chimiques, avec élimination d'eau.

conciudensé n. m. Résumé d'une œuvre littéraire.

conciudenser [kɔ̃dɑ̃sɛ] v. t. (lat. *conciudensare*, rendre épais). Rendre plus dense, réduire à un moindre volume. | Liquéfier un gaz par refroidissement ou compression : le froid *conciudense* la vapeur d'eau. | Fig. Exprimer d'une manière concise, en peu de mots :

FIGURE 4.1: First and second segmentation levels of a dictionary page (Larousse, 1972)

conciudenser [kɔ̃dɑ̃sɛ] v. t. (lat. *conciudensare*, rendre épais). Rendre plus dense, réduire à un moindre volume. | Liquéfier un gaz par refroidissement ou compression : le froid *conciudense* la vapeur d'eau. | Fig. Exprimer d'une manière concise, en peu de mots :

FIGURE 4.2: Example of the segmentation performed by the Lexical Entry model (Larousse, 1972)

in the case of full scientific articles, and the actual constructs we wanted to extract from print dictionaries.

8. References

- Budin, G., Majewski, S., and Mörth, K. (2012). Creating lexical resources in tei p5. a schema for multi-purpose digital dictionaries. *Journal of the Text Encoding Initiative*, (3).
- Khemakhem, M., Foppiano, L., and Romary, L. (2017). Automatic Extraction of TEI Structures in Digitized Lexical Resources using Conditional Random Fields. In *electronic lexicography, eLex 2017*, Leiden, Netherlands, September.
- Lopez, P. and Romary, L. (2015). Grobid - information extraction from scientific publications. *ERCIM News*.

abuse¹ /ə'bjuz/ *noun* **1.** rude words ○ The people being arrested shouted abuse at the police. **2.** very bad treatment ○ the sexual abuse of children ○ She suffered physical abuse in prison. (NOTE: [all senses] no plural)

abuse² /ə'bjuz/ *verb* **1.** to treat someone very badly, usually physically or sexually ○ She had been abused as a child. **2.** to make the wrong use of something ○ He abused his position as finance director. **3.** to say rude things about someone ○ The crowd noisily abused the group of politicians as they entered the building.

FIGURE 4.3: Left: Excerpt from Bibliographic References Section in (Khemakhem, Herold, and Romary, 2018). Right: Excerpt of Lexical Entries in (Publishing, 2009)

Figure 4.3 illustrates visually such an analogy between the structure of *bibliographic* entries in a scientific paper and *lexical entries* in a print dictionary. The underlined fields highlight the analogue structures we noticed respectively between: *author names* versus *headword* (red), *publication year* versus *pronunciation* (green), *publication title* versus *POS* (blue) and *book title* versus *sense* (orange). To these backbone structures in both categories of documents, additional fields can be joined to the description of an entry such as *location* and *month* of a conference versus a *general note* about a dictionary article (purple). Note that the transitions between the fields of bibliographic and lexical entries are marked by a consistent change in either the typographic features (e.g. bold, italic, font change, etc.), or/and textual markers (e.g. parenthesis, slash, number, dot, etc.).

This correspondence is reinforced by the fact that GROBID actually relies on these features to perform a cascading parsing of the text of a scientific paper and, in particular bibliographic references, in the same way shown in Figure 4.3. These facts incite us to investigate the adaptation of such models to the parsing of lexical entries.

Scientific papers processed by GROBID and the print dictionaries we are targeting represent, however, logical and physical differences. First, the logical structure of many categories of dictionaries is more granular than bibliographic information and can support different interpretations of structure classifications, depending on the background of the expert (see Figure 2.3). Consequently, annotating lexical information is more costly and less consistent than bibliographic data in scholarly articles. Second, the vast majority of available scholarly papers are the product of work dating back to few decades ago which consequently means that most of them are born-digital as opposed to the available dictionaries we are targeting, which are mostly digitised. This implies more obstacles for processing and classifying the text they contain.

4.2.3 GROBID

GROBID was initially implemented as a machine learning system for parsing and extracting bibliographic data from scholarly articles, mainly text documents in PDF format. It relies on CRF models (Lavergne, Cappé, and Yvon, 2010) to perform a multi-level sequence labelling of text blocks in a cascade fashion which are then extracted and encoded in TEI elements.

Such an approach has been very accurate for that use case and the system's Java API ² has been one of the most widely used by bibliography research platforms and research bodies worldwide, including ResearchGate³, HAL⁴, Mendeley⁵, CERN⁶ among many others.

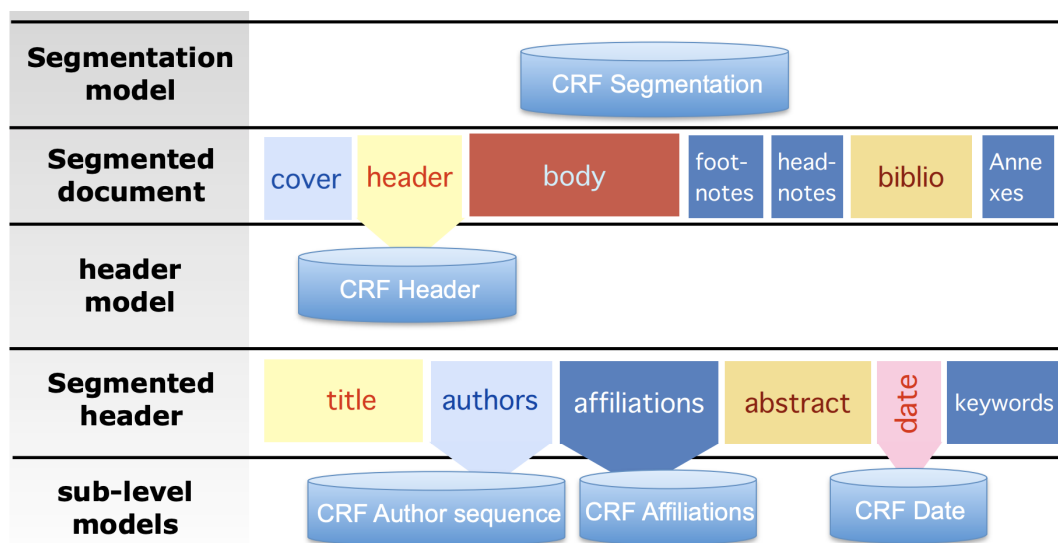


FIGURE 4.4: Excerpt from GROBID's architecture⁷

GROBID as a platform for manipulating PDF documents has powerful functionalities to extract and manipulate text in digital and digitised material. In addition, its API provides a benchmark for measuring the performance of newly developed CRF models and represents a suitable framework for conducting machine learning experiments. The API benefits from an active developer community guaranteeing the sustainability and the constant evolution of the embedded NLP libraries.

On the other hand, the cascading behaviour of the system comes from a modular structure of the API which makes CRF models, implemented following the same logic, portable and easily pluggable to the core GROBID models (Lindemann, Khemakhem, and Romary, 2018).

²<https://github.com/kermitt2/grobid>

³researchgate.net

⁴<https://hal.archives-ouvertes.fr/>

⁵<https://www.mendeley.com>

⁶<https://home.cern/>

⁷the full architecture can be found in GROBID documentation <https://grobid.readthedocs.io/en/latest/grobid-04-2015.pdf>

Given all these conceptual analogies and technical conveniences, we chose GROBID as a core platform to implement our approach. In the following section, we present the novel architecture for parsing lexical information called GROBID-Dictionaries.

4.3 GROBID-Dictionaries

GROBID-Dictionaries (Khemakhem, Foppiano, and Romary, 2017; Khemakhem, Herold, and Romary, 2018) is a machine learning infrastructure which relies on cascading CRF models for parsing text content of dictionary pages. It takes as an input a PDF text document and generates a TEI P5 (Budin, Majewski, and Mörth, 2012) compliant encoding where the various segmentation levels are associated with an appropriate XML tessellation.

4.3.1 Cascading Lexical Models

Our cascading models are designed in a way to support the encoding of the detected structures in multiple TEI constructs. The TEI schemes and the decisions behind modellings granular information will be discussed in detail in Chapter 6. But for the purpose of explaining the scope of each model, we are focusing more on the adequacy of a certain TEI modelling to the implemented cascading mechanism.

After having fully encoded a lexical entry, the task becomes more specific and more challenging when it comes to defining the TEI structures to be extracted by each model. It is a question of finding the appropriate mapping between the TEI elements and the labels to be set for the models that share the task of structuring the text in cascade. In addition, the process is at the same time constrained by the need to avoid having structures from different hierarchy levels being extracted at once. In fact, the CRF models, as they could be used from the GROBID core, do not allow the labelling of nested text sequences. A modification of such an aspect is still possible but it could be costly and is not necessary for implementing our approach, as we want to keep the models the least complex possible to ease the feature selection process.

The matrix in Figure 4.5 represents a set of feature vectors (see Appendix A for the description of each feature) describing a lexical entry *condenser*, which will be labelled by the "Lexical Entry" model. The latter has the task of detecting the main blocks in a lexical entry, if they exist. For the sense information, the model has been trained to extract each parsed text sequence representing a sense. Each vertical column is a specific feature for all the tokens of the lexical entry and each horizontal line corresponds to all the features of each token. The feature vectors and columns serve as the basis for the feature selection process that will be explained in Section 4.3.2.

In the second phase, comes the role of the trained model to give a prediction of a suitable label for each token, based on all its feature values. A structure corresponds then to the sequence of tokens having the same label,

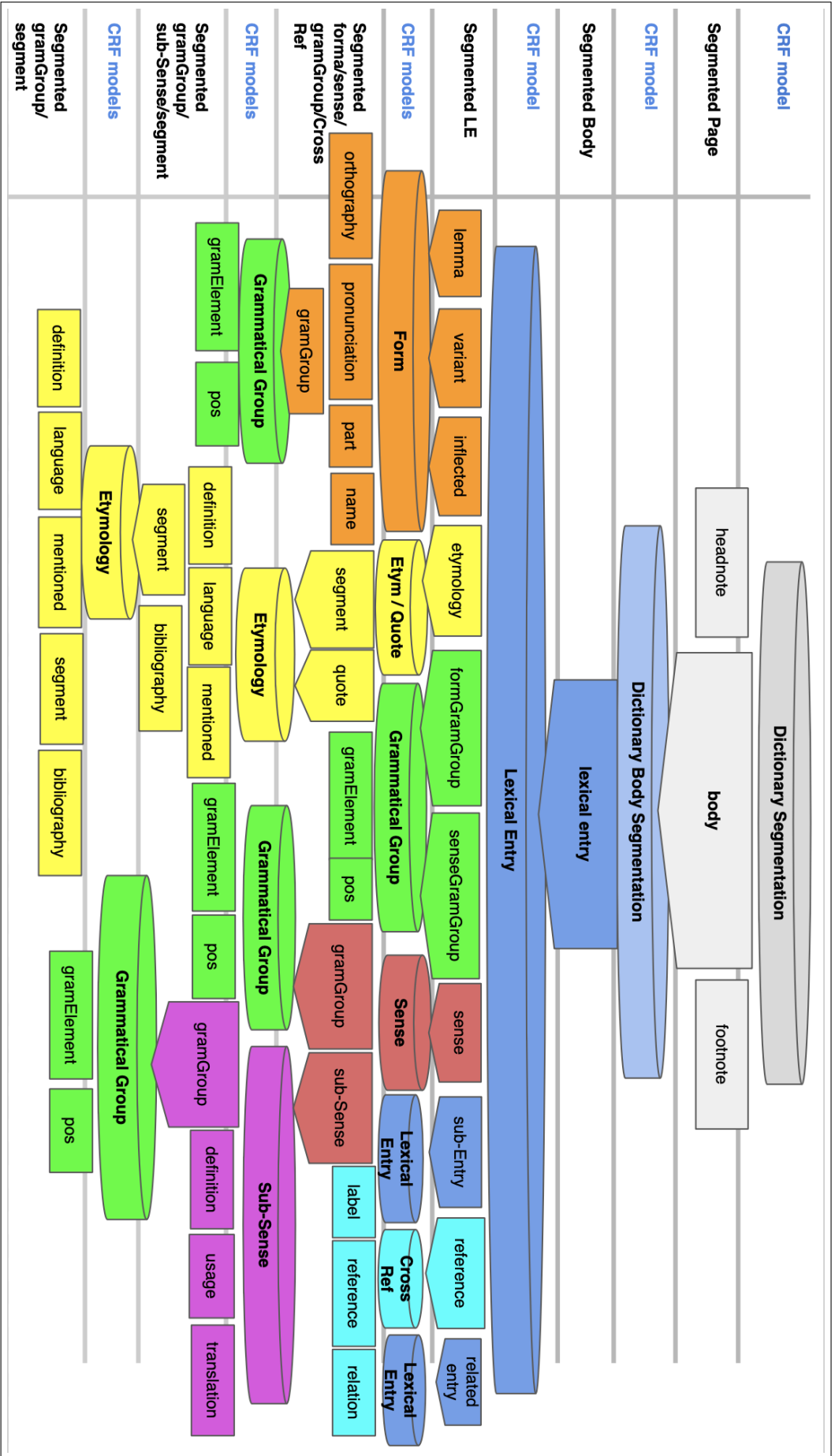


FIGURE 4.6: GROBID-Dictionaries’s Cascading Architecture

- **Dictionary Body Segmentation:** The second model gets the page body, recognised by the first model, and processes it to recognise the boundaries of each lexical entry by labelling each sequence with the *Lexical Entry* label.
- **Lexical Entry:** The third model parses each "Lexical Entry", recognised by the second model, to segment it into: *lemma*, *variant*, or *inflected* for morphological and grammatical information related to a "Form", *etymology* for etymological and diachronic description, *senseGramGroup* for grammatical information related to a "Sense", *sense* for semantic information or a "Sense", *sub-entry* for an embedded entry, *reference* for a reference to another entry, *related entry* for a related entry, *note* for general remarks about the whole entry, and *formGramGroup* for general grammatical description.

Another instance of this model can be used recursively to parse generated labelled *sub-Entry* and *Related entry* text blocks, as the logical structure of these sub-structures follows the same scheme as a *Lexical Entry* construct.

- **Form:** This model analyses any "Form" block such as *lemma*, *variant*, or *inflected*, generated by the **Lexical Entry** model, and segments the information it contains. The list of possible labels for this model contains for the moment: *orthography* to contain the orthography of a "Lemma" or a "Variant", *part* to contain the extent of an orthography of an "Inflected Form", *pronunciation* for pronunciation, *gramGroup* for grammatical information, such as POS, gender, number, etc, *language* for language information about a "Form", *name* for headwords in encyclopedic dictionaries, *description* and *note* for prose descriptions and notes related to morphology or grammar, and *usage* for usage information of the analysed form.
- **Grammatical Group:** This model is plugged at several levels of the architecture and has the task of parsing a group of grammatical information generated by **Lexical Entry**, **Form**, **Sense** or **Sub-Sense** models. The actual list of labels contains: *pos* for POS, *gramElement* for a piece of grammatical information that is going to be typed, *tense* for tense of a verb, *gender* for gender information, *number* for number, *subcategorisation* for information about transitivity, count-ability, etc., and *note* for prose notes about grammar.
- **Cross Ref:** Such a model makes it possible to parse cross reference constructs generated by the **Lexical Entry** model and potentially at other levels where the logical and physical structures are represented in a similar way. The main labels predicted by this model are: *label* for textual triggers of a reference (e.g. "See", "voir" or "V.", etc), *reference* for a internal or external reference, and *relation* to type the reference instance

(e.g. synonym, antonym, false friend, etc). Other labels are also enabled for this segmentation level such as *bibliography* for bibliographic information or *note* for further notes.

- **Sense & Sub-Sense:** These two models orchestrate the decomposition of the hierarchy of senses, if sense nesting occurs. For the **Sense** model, the supported labels are *subSense* for an embedded sense, *gramGroup* for existing grammatical information, *num* for sense numbering, and *note* for any prose description related to the upper sense. The mission of this segmentation model is to extract senses that will be parsed by the **Sub-Sense** model to recognise possible *gramGroup* for embedded grammatical information, *definition* containing a definition of a sub-sense, *example* for sense illustration, *translation* for translation equivalents, *usage* for usage information, *related entry* for possible embedded related entries, *etymology* for diachronic information related to the sense, and finally *reference* for recognised cross references. Note that, if no sense nesting occurs, these two steps still need to be followed one after the other. For an entry that has only one sense, this sense needs to be annotated first as a *sense* for the **Lexical Entry** model and then as a *subSense* for the **Sense** model. A control for displaying the final output will make sure that redundant *sense* tags are removed. Finally, an additional sense nesting level can be triggered when the *senseGramGroup* label is used and all the *sense* elements recognised by the **Lexical Entry** model along with *senseGramGroup* will be wrapped in a *sense* construct inside a *Lexical Entry*.
- **Etym/Quote & Etymology:** parsing etymological and diachronic information is carried out through two successive steps. The first makes use of the **Etym/Quote** model to differentiate *quote* text blocks from the rest of the etymological description, which is labelled as *segment*. Then both recognised blocks are parsed in the second step with an **Etymology** model to extract *definition*, *mentioned* for etymologically related words, *language* for information about the related word's language, *bibliography* for bibliographic details, and possible nested *segment(s)*. Such cascading processing is necessary as *quote* could have a complex structure that needs to be processed in the same way as *segment* constructs. These cascaded models can be used to parse diachronic information wherever it appears (e.g. under *subSense*, *related entry*, etc).

For all the models presented above, two more labels are required: *punctuation*, which represents an abstract label for any punctuation or symbol marking the separation among fields recognised by a CRF model, such as the full stop marking the end of a lexical entry's description, bullets marking the beginning of a definition or an example, etc. Such an abstraction helps the models to converge quickly, especially for some feature setups where the label of the previous token in a sequence is used as a feature to predict the label of a current token. This aspect will be explained further in Section 4.3.2.

The second label we added to the set of elements to be recognised by a model is *dictionary scrap*, which can be considered as a trash collector used to encode any text sequence that can not be labelled with any of the tags permitted for a model. Such a label is very useful when it comes to labelling a noisy text resulting from an OCR output or meta-data information that could figure in the text of the dictionary at any random position.

Finally, all these tags can be used more than once per level, as for multiple *lexical entry* to segment a *body* or *definition* to structure a *subSense*.

4.3.2 Feature Engineering

Given the impact of selected features on the learning of CRF models explained in Section 3.3.2, the feature *engineering* or *selection* step is crucial.

The process begins with preparing vectors describing the text sequence we want to structure (i.e. the text of a whole page, the text of a lexical entry, the text of a sense, etc). To do so, we designed the vectors based on two variations:

- **Token vs Line based clues:** The descriptive vectors are based on text and layout features of single tokens (see Figure 4.5). The vector generation mechanisms we implemented can adjust the level on which the *descriptive matrix* is focused. For the first model, **Dictionary Segmentation**, which has the task of parsing all the text of the document, is more efficient in terms of learning the distributions and guaranteeing a fast processing to use the first two tokens (punctuation marks with their preceding token are considered as one token) of each line rather than using all its tokens. Our decision was initially based on the observation of similar behaviour of analogue models in GROBID. Moreover, the experiments on several samples confirmed the rapid and efficient prediction results. This choice was dropped starting from the second model, **Dictionary Body Segmentation**, where lexical markers at different positions of the lines are often crucial for the model to learn the triggers for starting labelling a new field. For that, each vector describes each token of the text sequence.
- **Generic vs Lexical clues:** For the **Dictionary Segmentation** model, we used features based on a *descriptive matrix* analogous to the one defined for GROBID's first segmentation model (a detailed description of the vectors is provided in A.1.1). This choice was motivated by the power of such a model to detect macro areas in a document, a task which is similar to the scope of the first model in our architecture. For the rest of the models in GROBID-Dictionaries, we chose to rely on restricted descriptive vectors where we drop the information that is unlikely to be useful. The token position in the page, for instance, is very informative for the **Dictionary Segmentation** model to differentiate a page body from headnotes and footnotes but has almost no relevance, and is probably misleading for capturing a sense in a lexical entry that is very

often split over two pages. Excluding such input for the CRF models results in reducing the number of feature functions of a model and consequently making its training process and its size more compact. Therefore, we extracted only 16 descriptive features (see columns in Figure 4.5): 8 based on the text and the rest carrying layout and typography clues about each token, such as a change of font or line breaks (for more details, see Appendix A.1.2). However, we tried to maximise the abstraction over key lexicographic markers, namely *field separators*. In fact, we dedicated one flag feature to mark the tokens that are a punctuation, opening or closing brackets (parenthesis, square brackets, and braces are considered brackets in this case).

From the resulting *descriptive matrix* we used Wapiti⁸, a CRF library implemented by Lavergne, Cappé, and Yvon, 2010 and adapted within GROBID to perform the feature engineering process by experimenting combinations of selected features from each descriptive vector and its neighbouring tokens. Wapiti allows feature tuning to be performed through dedicated files specifying a set of *feature templates* (i.e. selected features). We studied the tuning of *feature templates* empirically to find the best combination for each model, based on the various samples we collected and annotated. Given the fact that the templates select the features from the descriptive matrix, the feature engineering process follows the same separation between the first model and the rest of GROBID-Dictionaries' architecture. In other terms, we prepared three classes of template combinations and for each class we define one combination for the first model of the architecture and another one for the rest of the models. We detail the combinations in the following:

- **Unigram and Bigram Feature Templates:** These two classes of templates differ in the possibility to take into consideration or not the transition probability from one label to another. A template is called a *Unigram feature template* when a label predicted by a trained model is based only on the features of the input sequence, whereas a more complex variation called *Bigram feature templates*, also takes the predicted label of the previous token into account. For the **Dictionary Segmentation** model we used the templates of GROBID's first model in their *Unigram* (see Appendix A.2.1) and *Bigram* variations (see Appendix A.2.3). Starting from **Dictionary Body Segmentation**, we used the newly defined descriptive matrices to define own *Unigram* (see Appendix A.2.2) and *Bigram* templates (see Appendix A.2.4). We defined these templates based on those used for the **Dictionary Segmentation** model and followed the same logic to define the descriptive matrices, by dropping information like the position of a token within a block or a page, or the description of the current line. For the remaining templates, we defined a combination that has a restricted *bidirectional window* of information about neighbouring tokens.

⁸<https://wapiti.limsi.fr/manual.html>

- **Engineered Feature Templates:** As a third class, we used *Engineered feature templates* that we define based on rounds of feature selection. Enlarging the *bidirectional window* mentioned above is the main engineering action for this class. For the **Dictionary Segmentation** model we were not able to find a better combination that could significantly improve the existing Bigram combination. For the rest of the models, the idea was to find an advanced set of feature templates that could operate on data which have different levels of complexity and for a mid-level complexity sequence labelling model. For that, we relied on the corresponding set of *Bigram feature templates* and we tried to tweak it (see Appendix A.2.5) using two different dictionaries, a modern English born-digital dictionary (see Section 7.2.3) and a legacy French digitised dictionary (see Section 7.2.3), and the **Lexical Entry** model, which we consider to be the component of our architecture that has the average complexity level of parsing (i.e. in terms of granularity). For all features focusing on information about neighbouring tokens, we enlarged the window by 3 in both directions to provide longer range clues for predicting the class of a token. For instance, the window of templates focusing on the token and its neighbours goes from 4 to 7 and templates pointing to the typography of tokens in the sequence go from 1 to 4 neighbouring words.

For *Unigram templates*, the total T of generated distinct features is $L \times N$, where L is the number of labels and N is the number of unique features generated by the templates. For *Bigram* and *Engineered* templates, T equals $L \times L \times N$. As a result of feature engineering, a larger L in the case of *Bigram templates* and a larger N in the case of *Engineered templates* produce a larger number of distinct features T . It is worth pointing out that the cost of a growing T is a slower training process and heavier resulting models which could slow similarly heavy models when they are called in cascade.

The impact of the presented set of template combinations on the labeling performance of the models will be exhaustively presented in Chapter 7.

4.3.3 Model Activation & Call

To use the models of the architecture, two stages need to be followed through the functionalities available in the two facets that the system represents.

Models Activation: MATTER Workflow

The activation of the architecture presented is enabled by following the MATTER methodology (Model–Annotate–Train–Test–Evaluate–Revise, see Figure 4.7) introduced by Pustejovsky and Stubbs, 2012. Projected onto GROBID-Dictionaries and the processing of lexical resources, the individual steps are as follows:

Model: define a CRF model for predicting different text structures at one stage and determine the corresponding feature set. This phase requires a programmer to create the defined models and integrate them into the cascading architecture.

Annotate: assign a TEI tag to each text block representing a lexical entity defined within a model's scope. This task must be performed on an XML representation of the data and must be strictly synchronised with the corresponding feature matrix file. The annotation guidelines⁹ need to be respected.

Train: use each annotated batch of data to train a corresponding model. The cascading architecture of the models should be respected here.

Test: this step gives just a rough idea about how the trained model behaves on unseen data. There are many ways to accomplish this goal. The easiest one is to run the corresponding web service from the web application on a held-out sample.

Evaluate: a precise evaluation with different measures is possible at the end of the training process as long as annotated data are provided under the dedicated location in the dataset.

Revise: the last stage concerns reviewing the modelling and annotation steps that have been described in the guidelines. Four possible measures are the outcome of this step:

- annotate more data when an improvement in the results was achieved,
- refine the annotation guidelines for new variations observed in the last training batch
- proof-read the performed annotations when minor anomalies are noticed
- think about redefining the modelling when the results represent unexplainable anomalies. This could be translated either into a simple feature engineering process or into a change of the logic behind and the scope of the models or their architecture.

Models Call: Cascading REST Services

After activation/training of the selected parsing models, these are called through REST services of the system's web application facet¹⁰. After calling the macro levels parsing service, the micro level parsing is organised according the required depth of the analysis of the lexical information.

After giving an input dictionary file, calling the *Parse dictionary* service triggers the **Dictionary Segmentation** model and the parsing result is then displayed in the web navigator. If the result is good enough, calling the second service on the list is then meaningful and the second model in the architecture is launched to parse the extracted body. Upon having good segmentation results, the **Lexical Entry** model can be called through the third service on the list. The final service on the list, *Parse full dictionary*, triggers

⁹<https://github.com/MedKhem/grobid-dictionaries/wiki/How-to-Annotate%3F>

¹⁰<https://traces1.inria.fr/grobid-dictionaries/>

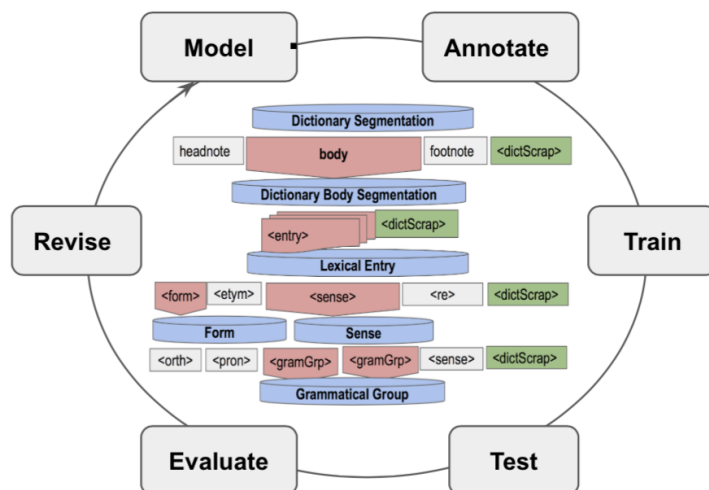


FIGURE 4.7: Implemented MATTER Workflow

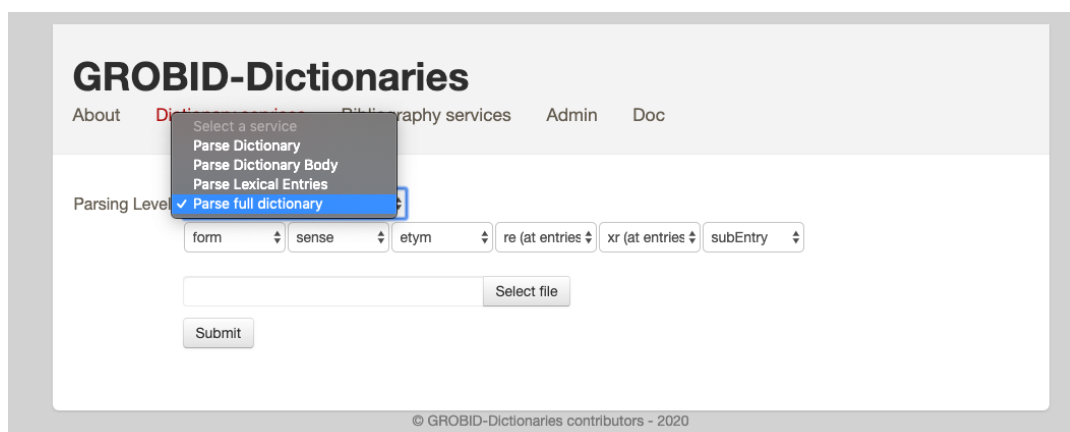


FIGURE 4.8: Cascading Model Selection in GROBID-Dictionaries

the customisable orchestration process of the rest of models. At this stage, the users have full control over calling the models they need for the parsing depth they want. Figure 4.8 shows a case of the cascading model activation process, where the users have the choice of calling one or several models for parsing the constructs of detected by the **Lexical Entry** model. They can then combine the output of the models they choose to get the best possible parsing result. More screenshots of the web application are provided in Appendix B.

It is worth pointing out that after each step at any parsing level, the users have the option to export the output they find the best as a TEI valid document. They can then continue the encoding of the structures in the exported document.

4.4 Lexicographic Knowledge Acquisition

Throughout the development of our approach and the implementation of the models in GROBID-Dictionaries, we faced several challenges that impacted on our priorities and pushed us to carry out more experiments in order to identify the nature of the issues and think about possible ways of overcoming them.

Extensive lexicographic expertise was required to define generic cascading models that satisfy the maximum of the lexicographic needs at each segmentation level. Making sure that the lexical models are applicable to different categories of dictionaries and languages that we do not master, was also a priority for us after encountering several modelling challenges. Therefore, we thought of collecting early feedback from domain experts through three measures. First, easing the setup of the system to attract more users, in particular those who have no IT knowledge or support to install and operate relatively complex tools in different running environments. Second, relaxing the complexity of the annotation workflow with more user-assistance and steps controls. Finally, offer collective training sessions to test the enhanced setup and collect feedback directly from the end users.

4.4.1 Easing Setup

To explain the improvement targeting the setup of the tool by a user with little IT skills, one needs to have an idea about the initial configuration of the tool.

Initial Configuration: IT Expert Use

GROBID-Dictionaries depends on core utilities and libraries provided by GROBID. The installation of the system must be preceded by the installation and setup of the parent project. Therefore GROBID-Dictionaries needs to be cloned as an extension module within GROBID's project structure and must be built after its parent project.

Due to differences in technical preferences of the project leaders, two different automation build technologies need to be used to build each project:

Gradle¹¹ for GROBID and Maven¹² for GROBID-Dictionaries. Successful builds of the system are packaged as Java libraries in two formats:

- a JAR (Java ARchive): this file is required for all processing stages which precede the training of each model, and
- a WAR (Web Application Resource or Web application ARchive): in the case of GROBID-Dictionaries this is not only a standalone web application but also a self-contained one that can be run after the training of the CRF models. It provides a graphical user interface to the existing web services, each corresponding to one or more of the cascading classification models.

GROBID-Dictionaries has been developed, tested and documented for the Linux and Mac operating systems. The behaviour of the resulting libraries is expected to be the same when run on other operating systems. However, there is no explicit guarantee for such uniform behaviour.

Enhanced Usability & Unified Execution Environment: DH Use

As a first measure, we investigated different ways to streamline the setup process and to guarantee a unique behaviour of the system across different execution environments.

One possible solution would have been to use a system image runnable on a virtual machine. Such an image should have a Linux-based operating system, a Java development kit (JDK) and the different automated build systems installed. GROBID and GROBID-Dictionaries should also already be cloned and built correctly. This type of solution suffers from two main issues. Firstly, the size of the image would be huge as it would include several unnecessary tools and system files that are still part of the operating system. Secondly, the static nature of such an image would make it complicated to update after a new version of GROBID-Dictionaries has been released. Updates to GROBID-Dictionaries are published frequently since the tool is under continuous development.

However, a system image containing the above-mentioned components can be built in a more efficient way using a different technique. Docker¹³ is a state of the art software technology which is also based on the virtualisation of the execution environment. In contrast to the static image approach sketched out initially, Docker allows for the flexible composition of an image. An image is shaped by instructions written in a Docker file¹⁴. These instructions ensure that only the required components are included in the image. Moreover, several alternatives are available to efficiently update a build within an image starting from pushing a newly created image to the online Docker Hub repository¹⁵, to linking the corresponding GitHub and

¹¹<https://gradle.org>

¹²<https://maven.apache.org>

¹³<https://www.docker.com>

¹⁴<https://github.com/MedKhem/grobid-dictionaries/blob/master/Dockerfile>

¹⁵<https://hub.docker.com/r/medkhem/grobid-dictionaries/>

Docker Hub repositories coupled with activating the automatic build to synchronise the image after each update of the code.

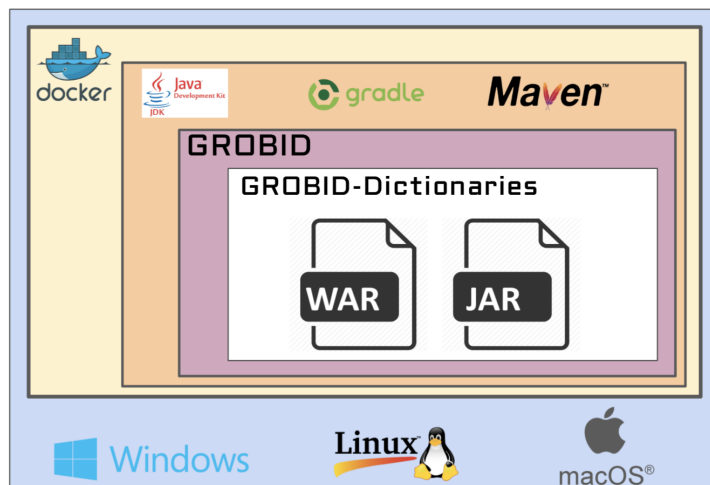


FIGURE 4.9: A GROBID-Dictionaries image in a Docker container

To run a Docker image of GROBID-Dictionaries (see Figure 4.9), a user needs to install the version of the Docker software corresponding to the user's operating system and pull the latest image of the tool from Docker Hub. The pulled image (orange box) will not be run directly on top of the operating system of the host machine but rather inside a Docker controlled container (yellow box). Thus testing the tool on Docker is enough to guarantee a unified behaviour, regardless of the particular system configuration of a user's computer environment.

It is also possible to synchronise files on the host machine with a running image in the Docker container. This feature allows the tool hosted inside a Docker container to directly interact with files stored on the host machine. We took advantage of this alternative to make the dataset directory shared between the two environments. With this mechanism, the user can exploit the full functionality of the tool living in the Docker image to train the machine learning models on the data residing locally on the user's machine.

In addition, thanks to the self-contained nature of the tool's web application coupled with its fluid setup and manipulation through the Docker image, using the GROBID-Dictionaries image enables both the desktop and web based functionality to be run on the user's local machine. Such a feature represents an asset for researchers who are concerned about the security of their data and experiments.

4.4.2 Lightening Annotation

The second major category of improvements specifically targets the annotation workflow. Annotating data for the training process involves challenging manual work and requires precautionary measures to ensure data integrity and validity.

Creating Training Data

To train a model in GROBID-Dictionaries based on a PDF file containing the raw text and the typographical features of a lexical resource, two additional files are necessary: a TEI document containing the corresponding reference encoding and a feature file describing textual and typographical information of each printed line or token.

To generate the training files, embedded functionalities of the tool should be used following one of the two following options:

- *pre-annotated training data*: this used to be the default mode for automatically creating training data, inherited directly from GROBID's core functionality. This mode is useful when a model was trained on a substantial amount of data. The task of the annotator is then to correct the automatically placed TEI tags by moving, adding or removing them.
- *raw training data*: this constitutes new functionality we have implemented to shortcut the checkout and cleaning of the tags automatically generated by using the default mode. The idea is simply to create training data without pre-annotations. Despite being obvious, starting to annotate a document from scratch was not possible before integrating this new feature. Such a mode breaks with the old practice of correcting the predictions made by a model trained on different samples, to make it possible to start annotating totally fresh data. Besides giving more choices to the annotator, such a mode saves time and effort, especially if an old model was trained with multiple TEI elements.

A legitimate question remains as yet unanswered: how can a user generate training data based on a selection of specific pages from the possibly hundreds of pages a dictionary may comprise?

After annotating different lexical samples in PDF format, we could qualify splitting an existing document into separate pages, or sequences of pages, as a very critical step. With some supposedly dedicated PDF manipulation tools producing damaged pages, we found only one tool reliably useful for the purpose of separating PDF pages¹⁶ which seems to produce a quality split as good as the original document. Using workaround solutions for this purpose, such as the print-to-file functionality in web browsers, is also not recommended.

Training Data Annotation

As previously stated, GROBID-Dictionaries generates a preprocessed XML representation from PDF files containing the raw text of a lexical resource. To create training data for the tool, the user is then required to introduce semantic mark-up for the different models. Typically, an XML aware editor should be used to perform this task. Some advanced editors such as oXygen¹⁷ allow for the visual annotating of XML files (see Figure 4.10 for an example).

¹⁶<http://community.coherentpdf.com>

¹⁷<https://www.oxygenxml.com/>

We aimed to take advantage of the visual feature to avoid performing in-line annotation directly on the text of the XML elements. This is catered for by a new feature in GROBID-Dictionaries that for each model now provides both a schema description (in Relax NG)¹⁸ and a presentational stylesheet (in CSS). The schema description enables the editing software to check or even enforce schema compliance of the training data. The stylesheet can be exploited by the editing software to allow users to mark up the training data semantically by highlighting portions of the text and then enclosing the highlighted portion with a suitable XML tag. The colours attributed to each element can be customised by a simple modification in the stylesheet.

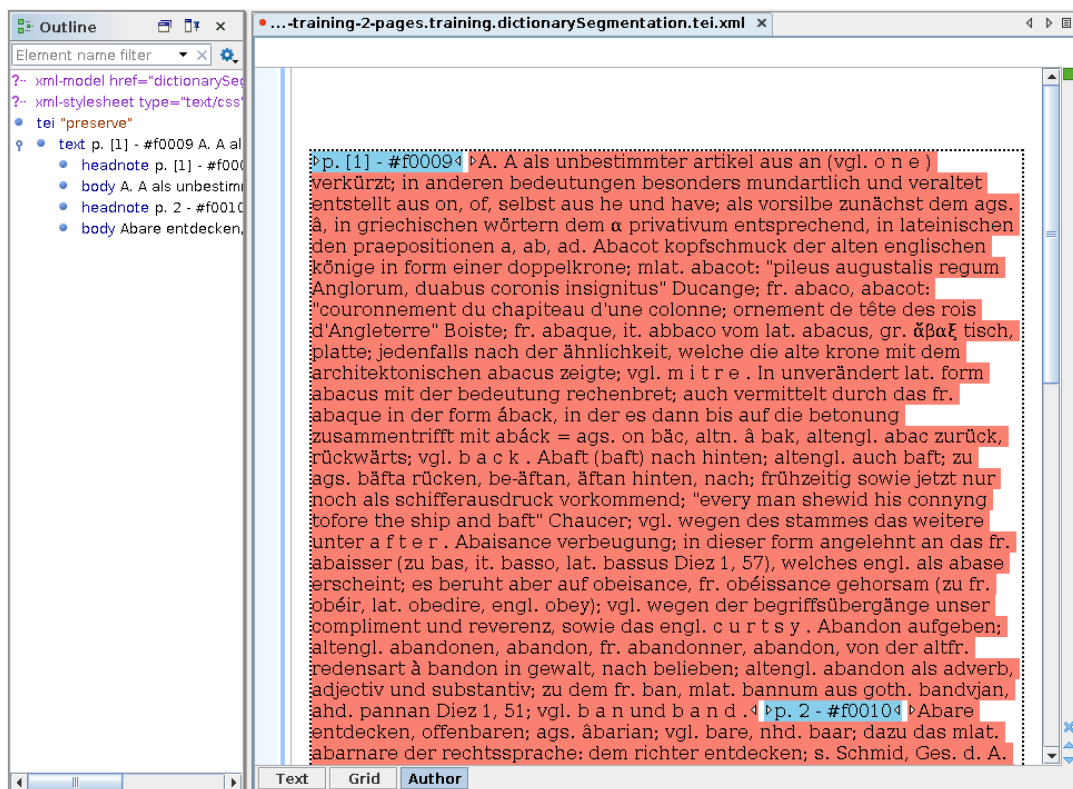


FIGURE 4.10: Training data annotation in oXygen author mode for the first model: page headers vs. page body

Train, Test and Evaluate

For this segment of the MATTER workflow, the user is provided with straightforward shell commands to execute, a graphical mode to test and varied measures to evaluate and decide whether a model has reached an acceptable level of accuracy. A simple but effective trick could however be employed at this stage to verify the accuracy of the annotations performed in the previous step. Where in a normal case the annotated data should be split between training and evaluation datasets, the training dataset could be also used as an evaluation dataset to verify any inconsistencies that might have accrued during the annotation process. In such a setup, and when a reasonable number

¹⁸<http://www.relaxng.org>

of pages have been annotated, a correct annotation should give almost 100 % accuracy, which means that model could reproduce what it has learnt correctly. Any other result should lead to the last step described in Section 4.3.3.

4.4.3 Training Domain Experts and Collecting Feedback

To test and adjust the two previous categories of enhanced usability measures, we chose to collect feedback from domain experts while and after manipulating and getting familiar with the GROBID-Dictionaries ecosystem.

Hands-on Sessions with End-Users

Besides the motivation, we had the opportunity to expose the under-development infrastructure, during different periods of its implementation process, to real users. These user experiences were organised as workshop series in the context of Masterclasses for lexicographic data^{19,20}, inter-disciplinary consortium gathering²¹, and dedicated events for promoting the new system^{22,23,24}.

The goal of the workshop was to familiarise the participants with the MATTER workflow as implemented in GROBID-Dictionaries (see Section 4.3.3), while excluding the first modelling step which requires programming skills. Note that none of the participants was familiar with the tool prior to the tutorial.

The average duration of a workshop is 5 hours. After a short introduction to the architecture of the system, the users were guided through the process of installing and running the docker image²⁵. Once the docker image was running, the participants were then able to reproduce the results reported in Khemakhem, Foppiano, and Romary, 2017 which are based on a modern English monolingual dictionary (Publishing, 2009). As the next step, several users used the possibility to experiment with their own lexical samples by repeating the workflow they had learnt and crafting new models for their individual datasets²⁶.

¹⁹<https://digilex.hypotheses.org/250>

²⁰<https://lexmc18.sciencesconf.org/resource/page/id/3>

²¹<https://cahier.hypotheses.org/3640>

²²<https://www.eventbrite.com/e/sadilar-grobid-dictionaries-workshop-pretoria-tickets-49730494247>

²³<https://www.eventbrite.com/e/sadilar-grobid-dictionaries-workshop-potchefstroom-tickets-49732031846>

²⁴<https://www.eventbrite.com/e/sadilar-grobid-dictionaries-workshop-stellenbosch-tickets-49731001765>

²⁵see instructions at https://github.com/MedKhem/grobid-dictionaries/wiki/Docker_Instructions

²⁶A more detailed description of the conditions of the experiment can be found in a blog-post at <https://digilex.hypotheses.org/250> and <https://digilex.hypotheses.org/category/posts> as shared by several participants.

Participants with Diverse Backgrounds

The tutorial groups consisted of users with various profiles, such as lexicographers, linguists, computational linguists, computer scientists, philologists and translators. Most of the participants had not previously trained machine learning tools.

After the very first workshop, we asked the participants of our tutorial to respond to a questionnaire created as a Google Form²⁷. The questionnaire gives an overview of the typical profiles and user experience, which we tried to improve after collected feedback. Although it was our first tutoring experience and the usability improvements were still freshly implemented, all the participants reported being confident that they were able to re-apply what they had learnt on other lexical resources.

The workshop series was held 8 times with pending ones to be organised after writing this thesis. The workshop series allowed us to train and collect feedback from over 100 users from different academic and industrial institutions, who also succeeded in tutoring new users in their institutions. Over 10 users are actively using the system and providing us with feedback and samples.

Various tested Material

The lexical resources brought to the tutorials were considerably varied. They included different types of dictionaries (some digitised, some born-digital with no explicit semantic markup) such as general monolingual, bilingual and etymological dictionaries as well as a dictionary from a language documentation field project (see Table 4.1). The tested samples confirmed the generic nature of our models and the fact they are language agnostic, as long as the samples are from alphabet-based languages.

Some separate experiments with arabic samples were not successful given a problem originating from the library responsible for extracting the text from the original document. We assume that the problem would be the same for other languages sharing the same writing system (from the right to the left).

Outcome & Gathered insights

Having motivated inter-disciplinary experts participating in the tutorial as well as testing the tool on new lexical samples provided us with the opportunity to spot some issues and several possible improvements.

We were able to fix some of the minor triggered implementation issues in the course of the tutorials. Other issues have been filed as new tickets on GitHub, e. g. issues concerning the treatment of lexical entries that stretch over more than two pages in print. Some technical issues related to the GRO-BID core still need to be resolved such as support for some classes of special characters which are wrongly encoded in the pre-processing of the raw input text. The annotation guidelines were also further refined to provide clearer

²⁷<https://goo.gl/Zt2gDy>

Type	Language(s)	Size
general, bilingual	Greek, English	≈ 17 000 entries
general, monolingual	Basque	≈ 16 000 pages
etymological, bilingual	Hittite (a language of the ancient Near East), English	≈ 470 pages
lang. documentation	French, Yemba (an African language family)	≈ 2 100 entries
lang. documentation	German (Bavarian dialects in Austria)	≈ 75 000 entries
general, monolingual	English	≈ 370 pages
Dialectal	Serbian	≈ 320 pages
Domain specific, Bilingual	German, Serbian (Mining Dictionary)	≈ 4,000 entries
general, multilingual	Macedonian, Serbian, English, French, Russian and German	unknown

TABLE 4.1: Some of the Dictionaries experimented with in some sessions of the workshop series

definitions of constructs to be annotated. But we made sure that the changes are not great to avoid confusing the trained users after the tutorials.

Lexicographers and linguists among the participants who gave us useful domain expertise comments, were many of which were taken up and resulted in implementing new labels for our models or even modifying the models behaviour. The limits of our modelling were exposed when the system had the challenge of parsing new samples according to different lexicographic practices.

Such a user experience helped us not only to verify our assumptions and improve our implemented approach but also to provide us with data we can use for our advanced experiments (see Chapter 7) and initiate cooperation that allowed us to widen the scope of our lexical models (see Section 7.6.1).

4.5 Chapter Summary

This chapter provided an in-depth presentation of our approach and the different solutions we found for the theoretical and practical challenges that we encountered.

We have presented GROBID-Dictionaries, a double-faceted ecosystem for parsing print dictionaries, which allows users with no advanced IT skills to train and customise the use of an architecture of machine learning models. We have also sketched out possible ways of leveraging the training of CRF models from an engineering perspective. This aspect will be showcased in Chapter 7 where we will present in detail the performances of the different models, given different combinations of features and varied tested samples.

Solutions to overcome the lack of lexicographic knowledge were also explained through practical and empirical measures. Our investigations in this direction may not be qualified as a user study but we managed to bridge the gap between engineers and domain experts by foreseeing such an obstacle. The positive results of such an early measure and the common issues discovered in related fields in humanities encouraged us to dig in new directions and open up new perspectives for our endeavours. We will further develop this aspect of our work in Section 7.6.1.

We have also highlighted some decisions regarding the encoding of lexical structures with respect to the implemented cascading mechanism. In Chapter 6 we will provide more details about the challenges of using a standard to encode dictionary resources and coming up with generic schemes that ease exchange and automatic processing. But before we dive into new standardisation schemes for lexical resources, the following chapter gives an overview of the state of the art of the standards for such resources and the obstacles facing the community regarding scalability and exchange.

Chapter 5

Standards for Structured Lexical Resources

5.1 Introduction

The dictionary parsing architecture we are presenting in this thesis aims not only at generating structured lexica for NLP downstream tasks, but also at allowing high exchange and interoperability with existing resources and querying tools. Thus, awareness and compliance with the existing practices and standards for structured lexical resources is required for the purpose of scalability.

Our study of the literature has shown a dominance of two standards, namely Text Encoding Initiative (TEI), Lexical Markup Framework (LMF) and one standardisation initiative, OntoLex-Lemon, for modelling lexical resources. Each of these modelling frameworks has been initiated to satisfy specific needs that represent several overlappings.

In this chapter we present the history and the motivations behind the foundation of these standards. We then provide a comparative study¹ of all of them with a focus on the strengths of each framework, and how it could fit the requirements of the structured output we wish to deliver.

5.2 TEI

TEI is a well-established standard that has proved popular within the lexicographic community. In this section, we give an overview of the standard from an engineering and lexicography newbie perspective.

5.2.1 Background

TEI (Sperberg-McQueen and Burnard, 1994) is a standardisation initiative that was launched in the late 80's with a view to finding a common framework for encoding text documents. The initiative has become a *de facto* standard and has been widely adopted in major humanities and documentation fields for academic, governmental and industrial projects².

¹Our survey is based on the state and the guidelines of these standards at the time when the work on this thesis started, specifically, September 2016

²<https://tei-c.org/activities/projects/>

In fact, the TEI guidelines give flexible and effective alternatives to structure texts for various purposes. They offer a strong document representation framework, with over 600 XML elements covering the needs to model almost any text. Its flexibility also lies in its specification language, ODD - One Document Does it all, which makes it straightforward to adapt the existing guidelines for close structuring requirements and supports the creation of new extensions for totally new needs.

The TEI guidelines are actively maintained and the P5 revision (Budin, Majewski, and Mörth, 2012) devotes an extended chapter to encoding lexical resources of different kinds. The impact of this initiative is clear in the lexicography domain, where it has been adopted as a main encoding format for large scale projects and resources in the field. Such success comes from the fact that a varied community of contributors is involved in shaping the standard and the reviewing process is pretty flexible and quicker than those for *de jure* standards.

5.2.2 Modelling Perspectives

The TEI guidelines provide a formal modelling of text in documents through a set of categories gathering related XML elements, called modules. The P5 version of the standard comprises 21 modules³ for marking up almost any piece of text, where the ninth is dedicated to encoding dictionaries.

Encoding Level

The TEI guidelines allow two levels of text encoding in print documents, where often one or the other is targeted by a user of the standard. The first level aims at reflecting the physical structure of a document by using elements from the “core module”. Such TEI elements can be used to encode paragraph and line beginnings, highlighted words, etc. Some elements can also be typed in to provide more precision on how they are typographically presented in the original print document. Figure 5.1 (C) illustrates the use of elements from the core module (i.e. `<p>`⁴, `<lb>`⁵, and `<hi>`⁶) to encode respectively a paragraph beginning, a line beginning and highlighted text segments in the lexical entry. Note the typed `<hi>` elements to markup the italic - *i* - and bold - **b** - in the text. Such a use is very common in the documentation and archive projects where the goal is to preserve every detail of the physical aspects of the text material.

The second level of encoding enabled by the TEI guidelines deals with the semantic and logical function of text structures. For lexica encoding, the “dictionaries module” provides a lexicon designer with an exhaustive set of TEI elements modelling different linguistic levels of the lexical information.

³<https://tei-c.org/release/doc/tei-p5-doc/en/html/ST.html#STMA>

⁴<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-p.html>

⁵<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-lb.html>

⁶<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-hi.html>

Such an encoding is also influenced by the linear description of print dictionaries, where common practices involve respecting the order of fields as they appear in the original document. Figure 5.1 (B) depicts the use of the elements of the specific module and shows the difference between this and the previous encoding level. Such a use is more popular within the lexicographic community and projects.

<p>Cabbage 1. <i>kohl</i>; <i>altengl.</i> cabage, <i>bei</i> Hal. 226 cabes, cabishes: <i>mlat.</i> gabusia, <i>fr.</i> cabus, <i>it.</i> cappuccio; <i>vgl. ndl.</i> cabuis, cabuyscoole, <i>nhd.</i> kappes, <i>worüber</i> Weigand 1, 562: „<i>Im</i> vocab. incip. teut. ante lat. kabbas, <i>mhd.</i> der kapaꝛ, kapeꝛ, <i>spätahd.</i> kabuꝛ, capuꝛ. <i>Aus fr.</i> der cabus, <i>it.</i> capuccio, <i>welches wie russ.</i> die kapusta kohl, <i>aus mlat.</i> caputium kapuze <i>hervorging und der geschlossene kohl schien einer mönchskappe ähnlich</i>;<i>“ vgl.</i> Diez 1, 110 <i>und unter den nhd.</i> kabisz, kabis Grimm 5, 9.</p>	<pre> <entry> <form> <orth>Cabbage</orth><label>1.</label> </form> <etym> <seg>kohl; altengl. cabage, bei Hal. 226 cabes,cabishes: mlat. gabusia, fr. cabus, it. cappuccio; vgl. ndl. cabuis, cabuyscoole, nhd. kappes, worüber Weigand 1, 562: </seg> <quote>„Im vocab. incip. teut. ante lat. kabbas, mhd. der kapaꝛ, kapeꝛ, spätahd. kabuꝛ, capuꝛ. Aus fr. der cabus, it. capuccio, welches wie russ. die kapusta kohl, aus mlat. caputium kapuze hervorging und der geschlossene kohl schien einer mönchskappe ähnlich;“ </quote> <seg>vgl. Diez 1, 10 und unter den nhd. kabisz, kabis Grimm 5, 9. </seg> </etym> </entry> </pre>
(A) Original PDF Excerpt	(B) TEI Encoding of the Logical Structure

```

<p><hi rendition="#b">Cabbage 1.</hi>
  <hi rendition="#i">kohl</hi>; altengl.</hi> cabage, <hi rendition="#i">bei</hi>
  Hal. 226 cabes, cabishes: <hi rendition="#i">mlat.</hi> gabusia,
  <hi rendition="#i">fr.</hi> cabus, <hi rendition="#i">it.</hi> cappuccio;
  <hi rendition="#i">vgl. ndl.</hi> cabuis, cabuyscoole,
  <hi rendition="#i">nhd.</hi> kappes, <hi rendition="#i">worüber</hi> Weigand
  1, 562: „<hi rendition="#i">Im</hi> vocab. incip. teut. ante lat. kabbas,
  <hi rendition="#i">mhd.</hi> der kapaꝛ, kapeꝛ, <hi rendition="#i">spätahd.</hi>
  kabuꝛ, capuꝛ. <hi rendition="#i">Aus fr. der</hi> cabus, <hi rendition="#i">it.</hi>
  capuccio, <hi rendition="#i">welches wie russ. die</hi> kapusta
  <hi rendition="#i">kohl, aus mlat.</hi> caputium <hi rendition="#i">kapuze
  hervorging und der geschlossene kohl schien einer mönchskappe ähnlich;“ vgl.</hi>
  Diez 1, 110 <hi rendition="#i">und unter den nhd.</hi> kabisz, kabis Grimm 5, 9.</p>
<pb n="171" facs="#f0179"/>

```

(C) TEI Encoding of the Physical Structure

FIGURE 5.1: Entry CABBAGE in (Mueller, 1878) and examples of its two Levels of TEI encoding

Encoding Workflow & Choices

Each module is described in the guidelines in a chapter that contains a formal description of the use of its elements. The use of TEI elements of a certain module does not exclude those from other modules.

A newbie willing to follow these guidelines needs to verify the adequacy of an element he/she chooses for the piece of information he/she wants to model by:

- reading the prose description provided in the guidelines for the chosen element (e.g. <entry>⁷)
- exploring the different instances of the use of the element in different contexts of the text to encode, which are often provided in more than one language

⁷<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-entry.html>

- verifying the validity to use a certain element as a child of one of its possible containers, indicated in the section “contained by”
- anticipating the modelling of child components by checking out the allowed elements for lower level, specified in the section “may contain”

In addition, a lexicon designer can stop at any depth of encoding when he/she considers that the necessary needs have been met. Figure 5.2 illustrates two valid modellings with different encoding elements and depths for the same dictionary article.

```

<entryFree>
  <orth>Cabbage</orth> 1. <etym>kohl; altengl. cabage,
  bei Hal. 226 cabes, cabishes: mlat. gabusia,
  fr. cabus, it. cappuccio; vgl. ndl. cabuis,
  cabuyscoole, nhd. kappes, worüber Weigand 1, 562:
  „Im vocab. incip. teut. ante lat. kabbas, mhd. der
  kapaz, kapez, spätahd. kabuz, capuz. Aus fr. der
  cabus, it. capuccio, welches wie russ. die kapusta
  kohl, aus mlat. caputium kapuze hervorging und der
  geschlossene kohl schien einer mönchskappe ähnlich;“
  vgl. Diez 1, 110 und unter den nhd. kabisz,
  kabis Grimm 5, 9.</etym>
</entryFree>
<entry>
  <form>
    <orth>Cabbage</orth> <label>1.</label>
  </form>
  <etym><def>kohl</def>; <lang>altengl</lang>.
  <mentioned>cabage</mentioned>,
  bei <bibl>Hal. 226</bibl> <mentioned>cabes</mentioned>,
  <mentioned>cabishes</mentioned>; <lang>mlat</lang>.
  <mentioned>gabusia</mentioned>, <lang>fr.</lang>
  <mentioned>cabus</mentioned>, <lang>it.</lang>
  <mentioned>cappuccio</mentioned>; vgl. <lang>ndl.</lang>
  <mentioned>cabuis</mentioned>,
  <mentioned>cabuyscoole</mentioned>,
  <lang>nhd.</lang> <mentioned>kappes</mentioned>,
  <bibl>worüber Weigand 1, 562</bibl>;
  <quote>„Im vocab. incip. teut. ante lat. kabbas, mhd.
  der kapaz, kapez, spätahd. kabuz, capuz. Aus fr. der
  cabus, it. capuccio, welches wie russ. die kapusta
  kohl, aus mlat. caputium kapuze hervorging und der
  geschlossene kohl schien einer mönchskappe ähnlich;“
  </quote>
  <seg>vgl.</seg> <bibl>Diez 1, 110</bibl> und unter den
  <lang>nhd.</lang> <mentioned>kabisz</mentioned>,
  <mentioned>kabis</mentioned> <bibl>Grimm 5, 9</bibl>.
  </etym>
</entry>

```

FIGURE 5.2: Different Depths of TEI encoding for Logical Structure

Note the use of `<entry>` and `<entryFree>`⁸ to encode the same macro-structure. `<entryFree>` is more favoured when the encoded entry has no clearly structured description. Moreover, using `<entryFree>` can substitute, to a large extent, the use of `<entry>`, and even gives more options to encode child components not allowed within `<entry>` (e.g. `<orth>`⁹ for encoding orthography or `<superEntry>`¹⁰ for certain cases of nesting entries). The modelling of orthography is also different in the two TEI samples, where the encoding depth varies. In fact, the use of `<entry>` forces the use of `<form>` to enable the encoding of orthography with `<orth>` element, where the same information can be encoded by directly using the `<orth>` element within `<entryFree>`. The granularity of the `<etym>`¹¹ block is also different in the two encoding examples. The designer of the lexicon on the left seems to be interested only in differentiating the lemma and the etymological information, while in the example on the right, more focus is given to identifying the micro-structure of the `<etym>` block.

⁸<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-entryFree.html>

⁹<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/examples-orth.html>

¹⁰<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-superEntry.html>

¹¹<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-etym.html>

5.2.3 TEI-based Lexical Resources

It is hard to enumerate all the existing lexical resources encoded in TEI, given its previously mentioned popularity. Nevertheless, we can differentiate between TEI resources that describe the physical structure and those which focus on the logical constructs.

For the first category, we can cite the work¹² of the Berlin-Brandenburg Academy of Sciences for digitising and transcribing legacy dictionaries such as Mueller, 1878; Uhlenbeck, 1900; Goedel, 1902; Kluge and Lutz, 1898.

For the second, we can cite EDBL (Aduriz et al., 1998), MULTEXT-East (Erjavec, 2004) or the work by Declerck, Mörth, and Lendvai, 2012 to convert Wiktionary data¹³ into a TEI compliant resource.

It is also worth mentioning that some OCR systems, such as Transkribus (Kahle et al., 2017), offer the option to export the OCR'd documents in TEI format. Such an output carries only markups for the physical description of an input print dictionary and can be referred to as “TEI resource” or “TEI version” of the original dictionary.

5.2.4 Discussion

Given the flexibility to choose from different encoding options for the same piece of information, lexica encoded in TEI can have different schemes. This freedom coupled with the wide adoption among lexicographers, having their own backgrounds and visions of the logical structure of a dictionary, has yielded a big bang of lexicon schemes and practices. Ironically, a standard that was supposed to unify the encoding formats under the umbrella of a common framework has turned into an uncontrolled modelling space.

TEI schemes are influenced by the documentation aspect of the standard given the fact that the linear aspect of the encoded constructs highly impacts the possible modelling options. Such a factor constrains the choices for a lexicographer and results in a non-optimal scheme dictated by all the possible orders of the lexical information in a sample, even the inconsistent ones. Such a fact adds a layer of complexity to the diversity of practices and schemes. Consequently, interoperability and exchange between resources compliant to the same standard have been significantly reduced.

5.3 LMF (2008)

LMF is a *de jure* standard published by the International Organization for Standardisation (ISO) for modelling lexical databases and MRDs. It was launched as ISO 24613 and is maintained by the standardisation sub-committee ISO-TC 37/SC 4¹⁴, and more specifically Working Group (WG) 4. In this section, we present a brief history of this standard as well as a theoretical and technical survey concerning its usage.

¹²<https://gitlab.com/xlhrld/retro-dict>

¹³<https://www.wiktionary.org/>

¹⁴<https://www.iso.org/committee/297592.html>

5.3.1 Background

LMF was initiated a bit later than TEI, as a followup initiative to multiple international projects such as ACQUILEX (Copestake, 1992), EAGLES/ISLE (Calzolari, Zampolli, and Lenci, 2002) and MILE (Bertagna et al., 2004). The most stable and well-known version of LMF, published by ISO in 2008 (Francopoulo et al., 2006)¹⁵, has offered a framework for modelling, publishing and sharing lexical resources with a special focus on requirements arising from the NLP domain.

The LMF workflow follows the standardisation practices within ISO where proposals and decisions are discussed and made within a high-level committee (i.e. ISO-TC 37/SC 4/WG 4) of experts and then validated/modified through an international balloting process. Compared to TEI, there is less transparency of the “in-between decisions” and less interaction with the future users. The use of the LMF standard is also charged, unlike TEI which is free. Nevertheless, it has gained a considerable number of users from academia and governmental institutions who have adopted the standard to take advantage of the combination of lexicographic and engineering expertise for the purpose of building universal and sustainable lexical databases.

5.3.2 Meta-model

LMF has proposed a meta-model for designing lexical resources formalised mainly in Unified Modelling Language (UML) (Fowler and Scott, 2004). The meta-model is composed of a core component and pluggable interlinked extensions defining an abstract model for different linguistic levels of the lexical information (see Figure 5.3). Examples of such extensions are Machine Readable Dictionaries, Morphology, Semantic and Syntax extensions. The meta-model is linked to data category registry (ISOCat - ISO 12620)¹⁶ that represents the elementary linguistic properties of the model components.

The meta-model introduces a number of interlinked classes and relationships, which are categorised into one of the aforementioned extensions. An instance of the diagram can be enabled by the instantiation of these classes and relationships and respecting their specified multiplicities. Figure 5.4 depicts an instance of the LMF meta-model.

Lexical Entry is the key class of any LMF meta-model and it holds the backbone of the lexical description. Morphological information and semantic information are respectively presented by means of *Form* and *Sense* classes and their sub-classes. *List Of Components* and *Components* classes, belonging to several overlapping extensions, represent the core modelling mechanism

¹⁵the academic publication about the standard was a mature draft of the main aspects of LMF that were accepted within the ISO-TC 37/SC 4/WG 4 and later went through the ISO validation and publication processes to be published later on in 2008

¹⁶ISOCat is no longer an ISO project after the decision to make such a project open to the user community. DatCatInfo took up the mission of representing such categories. It is maintained by LTAC Global / TerminOrgs. Further details can be found in: <http://www.datcatinfo.net/#/history>

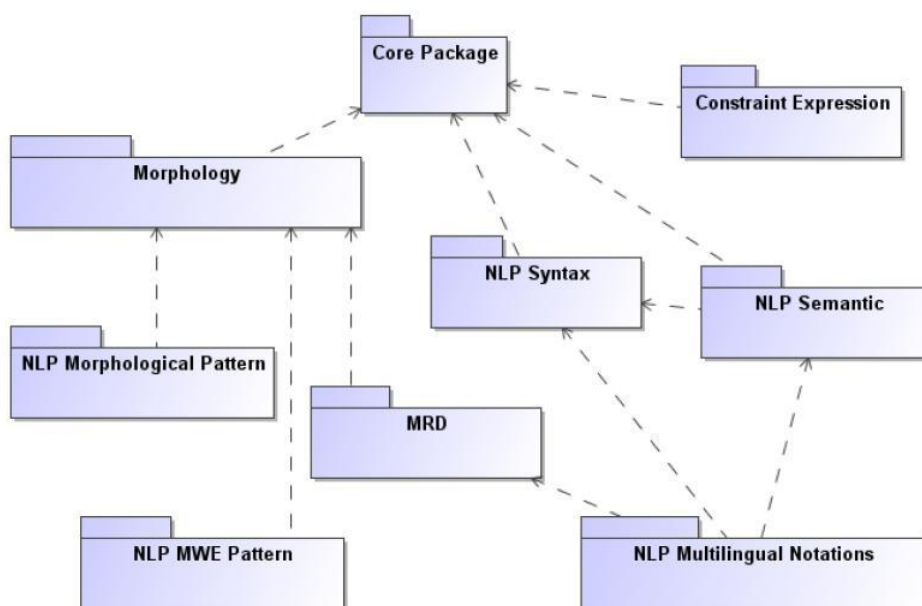


FIGURE 5.3: Dependencies between the LMF (2008) core and extension packages (Soria, Monachini, and Vossen, 2009)

of MWEs. Further details about LMF modelling principles are highlighted by Francopoulo et al., 2006.

5.3.3 Serialisation

LMF recommends an ad-hoc serialisation that represents a direct mapping between the class names in the meta-model and the XML element names or attributes. This serialisation figures in the standard as an Annex to the meta-model and through some scattered possible serialisation examples. But there is no comprehensive guide for serialising each element, defined in the standard's manual. Note that the serialisation can be carried out in any other format, since the XML option remains the recommended alternative but is not exclusive.

The standard is more NLP-oriented, which means that it gives almost no importance to the linear structures of lexical information in print dictionaries. The physical structure is also not considered as an aspect to be represented with the LMF meta-model. These facts give an idea about the scope of the resulting serialisation, in particular in comparison to a standard like TEI.

5.3.4 LMF-based Lexical Resources

A number of projects have adopted LMF as the main modelling framework to build lexical databases in different formats. The most known publicly available project resources is probably UBY (Gurevych et al., 2012) which



FIGURE 5.4: LMF (2008) Object Diagram for Modelling the MWE "DEAD CENTER"

is a database composed of 12 lexica in two languages (English and German). Commonly used lexical resources and knowledge bases like Wiktionary, WordNet, GermaNet and Wikipedia are among the lexica in UBY, having pairwise links on the sense level (Matuschek and Gurevych, 2013).

UBY has been natively implemented as an SQL Database but the UBY API enables the export of lexica as an XML serialised resource. Several downstream applications have been using UBY to carry out different tasks such as semantic annotation (Miller et al., 2016), machine translation (Beinborn, Zesch, and Gurevych, 2013), personality profiling (Flekova and Gurevych, 2015), and many others. The high coverage and the usability alternatives of a resource like UBY are to a great extent enabled thanks to the LMF meta-model behind.

El Madar (Khemakhem et al., 2016), a lexicon for the Arabic language modelling of MRDs, is another LMF-based resource that shows the meta-model's lexical coverage and extendibility. In fact, Khemakhem et al., 2016 managed to model over 37,000 lexical entries using the existing extensions and building on them new components, extending the existing meta-model. Using LMF to model dictionary entries in a highly inflectional and syntactically rich language like Arabic proves the genericity and the flexibility of the meta-model.

The list of resources instantiating the standard goes beyond these two resources and includes lexica in several languages or pairs of languages (Maks, Tiberius, and Veenendaal, 2008; Mykowiecka, Rychlik, and Waszczuk, 2012).

5.3.5 Discussion

LMF managed to overcome several obstacles towards a uniform representation of lexical resources for the NLP usages. The standard left, however, different gaps that restrained its wide adoption and becoming the de facto framework for modelling lexica. On the form level, the standard has an imbalanced structure with just 20 pages for the main content and over 60 pages for normative and informative annexes. In addition, only a few modelling examples come with a recommended serialisation. Consequently, the balance and the analogy of the content, required for an intuitive understanding are highly questionable for the case of this standard.

On the modelling level, the potential richness and the multi-layered nature of linguistic descriptions in lexical resources has resulted in the LMF meta-model taking on a great deal of complexity in its attempt to reflect these various different linguistic facets. Complex relationships between classes (e.g. useless *Component* class in Figure 5.4) and redundant mapping mechanisms (different mechanisms for the same abstract phenomenon e.g. synonymy, MWE, etc) were the side effects of the modelled lexical complexity. The latter has been propagated to the ad-hoc serialisation, which made the querying and the enrichment of the resulting resources challenging. Finally, key areas of linguistics such as etymology (and diachronic lexical information in general) were not covered by the meta-model which sanctions a whole

category of valuable resources that are supposedly within the scope of such a standard.

5.4 OntoLex-Lemon

OntoLex-Lemon is an evolving initiative for the representation of lexical data on the Semantic Web. In this section we present the history of its development, the theoretical and technical foundations, and the limitations of the resulting modelling.

5.4.1 Background

The rise of the Linked Open Data movement, mirrored on the linguistics field as Linguistic Linked Open Data ¹⁷, has created the need to represent lexical data as an ontology for the semantic web usage. LEXicon Model for ONtologies (Lemon) (McCrae, Spohr, and Cimiano, 2011) was the first attempt to find a dedicated model. After the foundation of the OntoLex Community ¹⁸ at the end of 2011, the group took on the improvement of Lemon and its upgrade to create the so called OntoLex-Lemon (McCrae et al., 2017). W3C, however, frames its role into a host of the discussions of the group and remains distant from the views and decisions of the latter ¹⁹

Since its creation, the OntoLex group has focused on more structuring of the model and the collection of several use cases for the purpose of widening the coverage of the Lemon model. In parallel, the proliferation of the tools that facilitate querying, linking and visualising of such resources has been attractive for many projects. The transparency of the workflow has to some extent helped the standard to strengthen interest in its use within the lexicographic community, aiming at representing their resources on the Semantic Web.

5.4.2 Modelling

The most recent version of OntoLex-Lemon, published in the official report of the OntoLex Community Group, defines a model structured into four main modules, besides a core model. The latter represents the description of a lexical entry as a constellation of concepts (see Figure 5.5) such as *Lexical Concept* and *Concept Set*, and main components, like *Form*, *Lexical Sense* and *Affix*. The other modules carry the representation of *Syntactic and Semantics*, the *Decomposition* of multi-word lexical entries, *Variation and Translation* aspects, and *Linguistic Metadata*.

¹⁷<http://linguistic-lod.org/>

¹⁸<https://www.w3.org/community/ontolex/>

¹⁹An explicit note from the consortium in the community page clarifies its position regarding the work of the Ontolex Community group: “Community Groups are proposed and run by the community. Although W3C hosts these conversations, the groups do not necessarily represent the views of the W3C Membership or staff.”

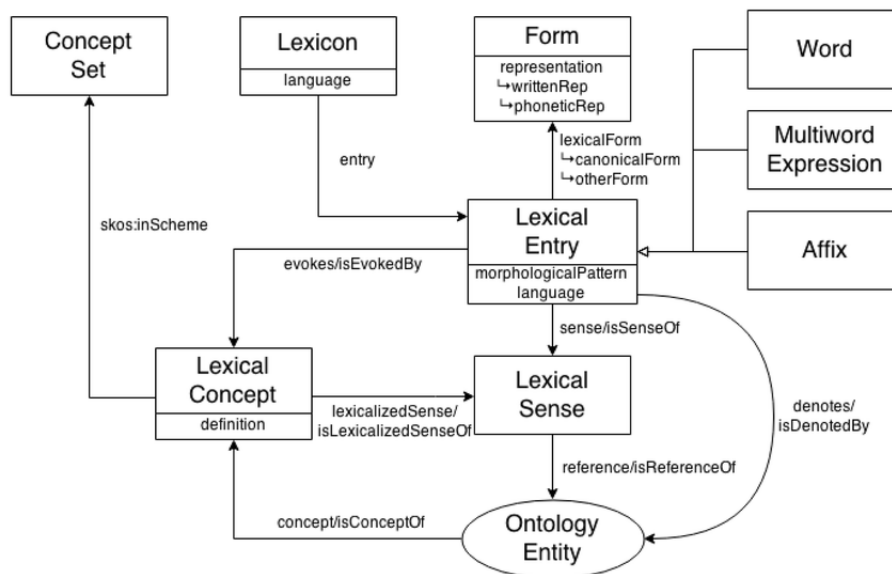


FIGURE 5.5: OntoLex-Lemon Core Model (McCrae et al., 2017)

OntoLex-Lemon’s modelling remains faithful to the semantic web formalism which is based on representing concepts and their relationships using Resource Description Framework (RDF) triplets: *subject*, *predicate* and *object*. The resulting model follows the principle of semantics by reference (McCrae et al., 2012), where the logical structure of a lexical entry is expressed by reference to an individual, class or property defined in the ontology. In some cases, the lexicon itself can reference object properties that belong to other ontological models such as *LexInfo*²⁰. In fact, the defined triple-based vocabulary specifies lexical concepts and possible internal links, forming the definition of a lexical entry, and external relationships that represent links among lexical entries. The modelling principles are heavily inspired by those implemented in other models, like LMF’s.

In Figure 5.6, we show an instance of the OntoLex-Lemon model for representing the same compound (i.e MWE) “dead centre”, already modelled using LMF and depicted in Figure 5.4. We can notice the LMF inspiration in modelling the components of the MWE and in the representation of Sense and Form information. The modelling diagram is however ad-hoc, despite a great resemblance to UML schematisation.

The triple-based vocabulary is manifested in each concept, as for instance: **centre_n is a Lexical Entry**, and between two concepts, such as: **centre_n is a noun**.

Note the complex modelling of the *geographical variant* forms of the *canonical form* (i.e. lemma) in **centre_n_form** concept, where the *written* and *phonetic representation* have composite values. In addition to being less intuitive, in comparison to other modelling frameworks like LMF, such a representation makes it more complicated for automatic systems to enrich or query atomic values for each attribute of a property.

²⁰<https://www.lexinfo.net/>

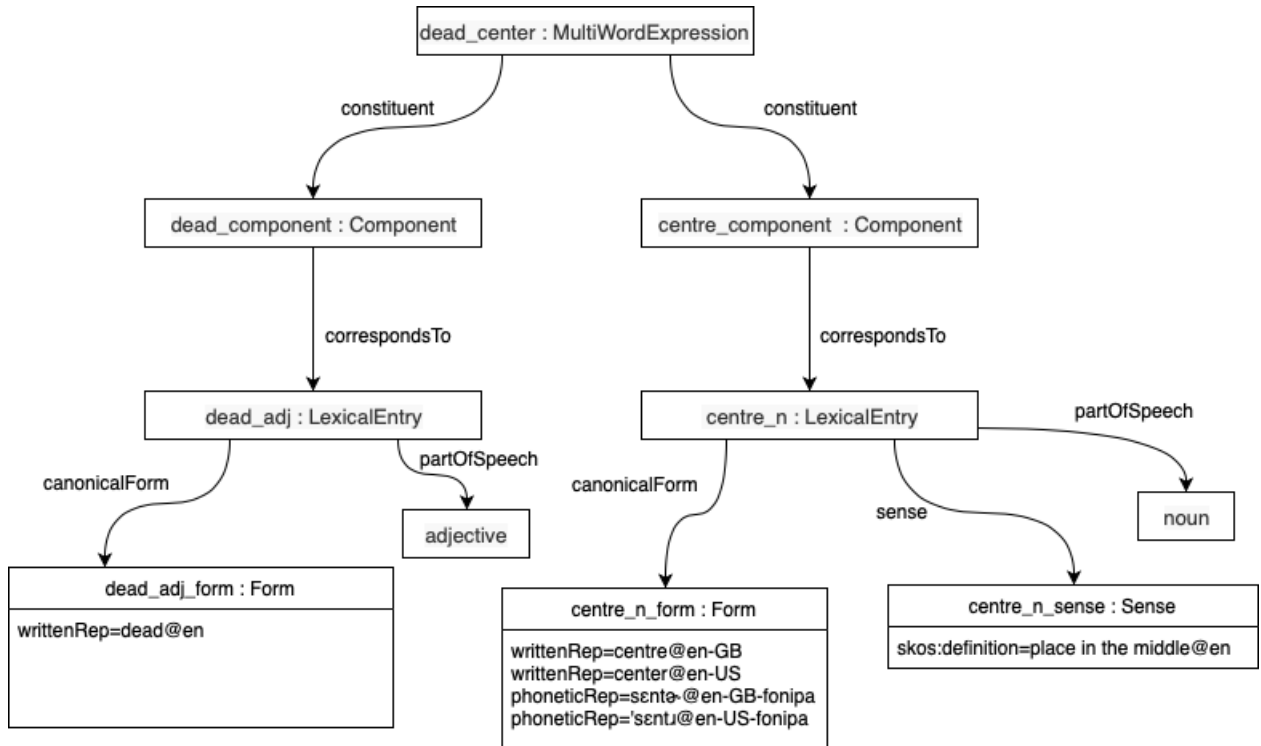


FIGURE 5.6: First Instance of OntoLex-Lemon Model for the MWE "DEAD CENTRE"

OntoLex-Lemon offers the possibility to encode a MWE differently, by considering it an elementary form and consequently short-cutting the need to model its *components*. Figure 5.7 depicts such a different design.

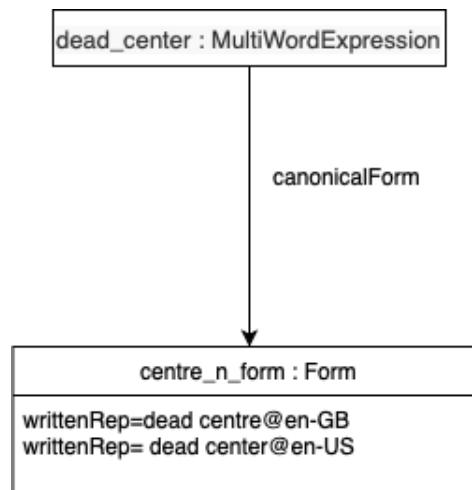


FIGURE 5.7: Another Example of Instantiating OntoLex-Lemon Model for the MWE "DEAD CENTRE"

We noticed that the official report of OntoLex-Lemon does not provide guidelines for modelling other categories of lexical information like deeper sense relations or Etymology. But defining customised extensions for supporting information, having such granularity and category, is also possible.

And as long as the design in the *OntoLex-Lemon* report and the triplet modelling principles are respected, a new extension can be qualified as “*OntoLex-Lemon compliant*”.

Related work has succeeded in modelling such advanced extensions like *polyLemon* (Khan et al., 2017) for deeper sense information, and *lemonEtym* (Khan, 2018) for etymological description. The ongoing revisions of the model are also focused on its enrichment by improving existing extensions as for *morphology*²¹, or defining new ones as for *frequency*²².

5.4.3 Serialisation

The serialisation of *OntoLex-Lemon* relies mainly on the syntax and vocabulary of RDF (Klyne, 2004) and RDFS (McBride, 2004). Another Semantic Web framework, OWL (McGuinness and Van Harmelen, 2004), is also used within *OntoLex-Lemon*’s serialisation to express more complex relationships along with constraints on data values, such as cardinality and data range.

The defined serialisation can be created in different formats, such as RDF/XML and JSON, while the *OntoLex* documents uses Turtle²³ as the main language to serialise the examples of modelling diagrams.

Figure 5.8 illustrates a Turtle serialisation of the entry, modelled in Figure 5.6, by following the *OntoLex-Lemon* vocabulary. Note that every line represents a triplet expressed in the diagram. The referencing to the internal modules is manifested in, for example, the definition of components of the MWE:

```
centre_component a decomp:Component
```

There are also instances of external links to object properties, such as the one to *LexInfo* in:

```
lexinfo:partOfSpeech a lexinfo:adjective
```

The choice of a certain serialisation format does not exclude a straightforward migration to the others. In fact, through the existing translation tools^{24,25} for semantic web technologies, it is possible to transform any RDF serialisation into other formats, like JSON and Turtle, upon the click of a button. Such a feature ensures more interoperability and exchange for the *OntoLex-Lemon* resources on the Semantic Web.

5.4.4 *OntoLex-Lemon*-based Resources

Theoretically, *OntoLex-Lemon* allows the modelling of unstructured lexical data in print dictionaries. In practice, the ontological model has been mostly

²¹<https://www.w3.org/community/ontolex/wiki/Morphology>

²²<https://acoli-repo.github.io/ontolex-frac/>

²³<https://www.w3.org/TR/turtle/>

²⁴<https://rdf-translator.appspot.com/>

²⁵<http://www.easyrdf.org/converter>

```

:centre_n a ontalex:LexicalEntry ;
  ontalex:sense :centre_n_sense;
  ontalex:canonicalForm :centre_n_form ;
  lexinfo:partOfSpeech lexinfo:noun .

:centre_n_form a ontalex:Form;
  ontalex:writtenRep "centre"@en-GB, "center"@en-US;
  ontalex:phoneticRep "sɛntəˈcentre"@en-GB-fonipa;
  ontalex:phoneticRep "'sɛnt.ɹ"@en-US-fonipa.
  |
:centre_n_sense a ontalex:Sense ;
  skos:definition "place in the middle"@en .

:dead_adj a ontalex:LexicalEntry ;
  ontalex:canonicalForm :dead_adj_form ;
  lexinfo:partOfSpeech lexinfo:adjective .

:dead_adj_form a ontalex:Form;
  ontalex:writtenRep "dead"@en .

:centre_component a decomp:Component ;
  decomp:correspondTo :dead_adj .

:dead_component a decomp:Component ;
  decomp:correspondTo :centre_n .

:dead_center a ontalex:MultiWordExpression ;
  decomp:constituent :dead_component;
  decomp:constituent :centre_component .

```

FIGURE 5.8: Turtle Serialisation of the MWE "DEAD CENTRE" and its Components

used as a scheme to transform already structured lexical data into lexical databases. To our knowledge, only a very small number of small sized lexica have used it as a native model/serialisation. From our newbie experience, our explanation for such a phenomenon is the need to break down everything in a lexical description into at least one triplet and often into more than one. Such a modelling principle is not very intuitive and not practical, especially when it comes to encoding from scratch complex lexical information that involves composition, rich atomic values (e.g. the phonetic and written representations in Figure 5.8) or structure nesting.

A number of transformation experiments have been successfully carried out with TEI and LMF-based databases among other native formats. The best known examples of such a conversion are probably the resources resulting from the transformation of NLP lexica, widely used in field, like *UBY* (Gurevych et al., 2012) transformed into *UBY-Lemon* (Eckle-Kohler, McCrae, and Chiarcos, 2015), and *BabelNet* (Navigli and Ponzetto, 2012) converted into *lemon-babelnet* (Ehrmann et al., 2014), or the TEI version of *Liddell-Scott* (Liddell and Scott, 1896) compiled by the *Perseus project* (Magazine, 1998) which has been transformed into an OntoLex-Lemon lexicon .

5.4.5 Discussion

OntoLex-Lemon managed to open new perspectives for lexical data by offering a framework for its representation on the Semantic Web and consequently overcoming ad-hoc serialisation and format issues.

Nevertheless, this initiative represents several limitations for the use of its scheme, as a native format for modelling lexical structures and their relationships. Firstly, this is due to the relative complexity of its triplet-based vocabulary, compared to standards like TEI and LMF. The use cases, based mostly on conversion scenarios, support this claim. Secondly, the different possibilities to design the same lexical concept, such as the case of MWEs (see Figures 5.6 and 5.7), represent an obstacle towards reaching a unified representation of the lexical information which results in reducing exchange and comparability alternatives. Moreover, the standard is under active revision and great progress has been already achieved to support the modelling of more granular and new classes of information. However, it remains insufficiently mature to cover the modelling requirements in print dictionaries.

5.5 Chapter Summary

This chapter gave an overview of the state of the art of the leading standards and initiatives for modelling structured lexical resources and we highlighted their strengths and shortcomings.

The TEI guidelines, being the most adopted within the community, have provided lexicon designers with multiple mechanisms to model similar lexical structures according to different perspectives and practices. The flexibility offered, however, needs to be restrained by finding more compromising

encoding schemes/practices to ensure a maximum interoperability of resulting resources. Our aim is to find a modelling that converges towards a unified TEI scheme for our parsers.

LMF has more control on the encoding practices and provides a stronger formalism for modelling the lexical information, making the approach of the standard a candidate framework for defining the target output we want to generate from the parsed dictionary material. However its complexity, lack of coverage and NLP orientation need to be relaxed in order to support the specificity of our target dictionaries.

Ontolex-Lemon has great potential for enabling high interoperability and exchange allowed by the Semantic Web technologies. But the lack of coverage and the differences in representing the same information are not in favour of directly adopting it for modelling the lexical output of our parsers. Nevertheless, the active revisions and the previous conversion experiments leave the door open for the use OntoLex-Lemon to represent the output of our models on the Semantic Web, by using TEI or LMF serialisation as a pivot format. Such a choice has been already adopted in ELEXIS²⁶, a major European project for electronic lexicography that shares with us the same parsing and modelling goals, which confirms the limitations of the ontological model.

In Chapter 6, we present the endeavours targeting the improvement of our two candidate standards TEI and LMF. We show how these initiatives overlap with our work and how our models can generate a comparable output.

²⁶<https://elex.is>

Chapter 6

Novel Standardised Schemes for Encoding Dictionaries

6.1 Introduction

The issues highlighted in Chapter 5 regarding the modelling aspects of two leading standards for lexical resources, TEI and LMF, have been already reported by the lexicography community and some concrete proposals have consequently been made to find better modelling frameworks with mutual improvements (Romary, 2010; Czaykowska-Higgins, Holmes, and Kell, 2014; Romary, 2015). But these observations and suggestions lacked the context that moves towards gathering individual interests in such a direction under one umbrella.

The two standards have different approaches and revision workflows. Thus, combining the best of the two standards represents a considerable modelling challenge and requires an active involvement in both standardisation ecosystems. Fortunately, this thesis began when there was already consensus within European and International projects on the need to act and find better TEI and LMF. In addition, we had the opportunity to be closely involved in such revision endeavours, which helped us to be aware of standardisation issues and important modelling challenges of lexica that we took into consideration to shape the output of GROBID-Dictionaries.

In this chapter, we present two novel standardised schemes for encoding lexical resources: TEI-Lex-0 (Romary and Tasovac, 2018) and the recently revised version of LMF (Romary et al., 2019). We report on the modelling aspects brought by these new frameworks and the main challenges which impacted on some key encoding choices. We then show the overlap with the output of GROBID-Dictionaries and the main differences.

6.2 TEI-Lex-0

The definition of a baseline encoding that enables easier exchange and exploitation of TEI lexical resources was the key driving force behind the initiation of TEI-Lex-0. Such an encoding scheme in no way replaces the TEI dictionaries module. In fact, it has the goal of defining guidelines towards a more systematic use of TEI elements, based on observations of common

practices followed to encode existing TEI-based lexical resources. The guidelines should support newly created TEI resources, as well as, the unequivocal transformation of existing TEI dictionaries into a unified baseline encoding. In this section, we refer by *minimal encoding* to the representation of macro structures which does not present the exhaustive encoding of fine grained lexical information, such as <orth> within <form> or <def> within <sense> constructs.

6.2.1 Context

The preparations for the TEI-Lex-0 initiative were initially discussed within the “Retrodigitised Dictionaries” Working Group¹ under the umbrella of the COST Action European Network of e-Lexicography (ENeL). The DARIAH Working Group “Lexical Resources”² took up the work of establishing the TEI-Lex-0 guidelines and support from the H2020 funded project PARTHENOS³, which has been accorded within its standardisation Work Package 4 (Romary et al., 2017). The TEI-lex-0 has been chosen to be a pivot format for comparing, querying and visualising structured lexical resources. Upon the kick-off of ELEXIS⁴ and given high overlapping interests, partners have also been involved in the discussion around the expected interoperability requirements expected from TEI-Lex-0.

Technically, the main discussions are carried out through several face-to-face meetings, organised every few months. Experts, mastering several languages and having different lexicographic backgrounds, bring and discuss different raw or already encoded lexical samples. Critical and open questions are publicly posted as github tickets⁵ to push the discussion forward and collect feedback from the community. Finalised decisions are then implemented in dedicated public guidelines⁶.

A number of simplification principles were obvious and easily got the green light from the majority of the experts around the table (e.g. main elements of a lexical entry, lemma encoding, sense made mandatory for each dictionary article). However, several recommendation decisions were not straightforward to reach, as compromising comes at a cost and each expert has his/her own legitimate vision and preferences for modelling TEI structures.

In the following section, we sketch out some challenging modelling choices to illustrate the obstacles encountered and how they were overcome.

¹<http://www.elexicography.eu/working-groups/working-group-2/objectives/>

²<https://www.dariah.eu/activities/working-groups/lexical-resources/>

³<http://www.parthenos-project.eu>

⁴<https://elex.is>

⁵<https://github.com/DARIAH-ERIC/lexicalresources/issues>

⁶<https://dariah-eric.github.io/lexicalresources/pages/TEILex0/TEILex0.html>

6.2.2 Modelling challenges

The complexity of modelling logical structures in TEI dictionaries was dismantled based on the main macro structures of a lexical entry in a print dictionary. These structures are then studied individually based on their components as well as their belonging to higher level constructs (i.e., *contained by* and *may contain* sections in the description of each TEI element).

We present encoding use cases that resulted in major recommendations reached within TEI-Lex-0 and which were propagated to the TEI Dictionaries module itself. For the remainder of this thesis, we denote by *element@attribute* the attribute of an XML element.

Recursive entries

<entry> is a TEI element containing the description of a lexical entry in a dictionary. The TEI guidelines provide a lexicon designer with additional elements to encode specific categories of lexical entries and entry-like constructs like <entryFree> for less structured entries, <superEntry> for higher level entries (e.g. roots in Arabic dictionaries) and <re>⁷ for related entries (e.g. collocations, idioms, etc).

To ease search in multiple resources and extract comparable structures, the TEI-Lex-0 guidelines recommend the use of only <entry> to encode all entry and entry-like constructs. The differentiation among these different constructs is specified by means of type values on the <entry> elements, where these values denote properties of the entry as a whole and not expressed by means of other elements within the encoding of the entry. Such a classification can be seen from different perspectives. Therefore, TEI-Lex-0 guidelines recommend several domain lists for such values.

The unified entry-like structure is vowed to be an autonomous and referenceable construct in a number of scenarios, such as internal mapping, indexing, inter-resources exchange or/and language disambiguation of lexica. Therefore, a recommendation, of making *@id* and *@language* attributes mandatory for each entry construct, reached the absolute agreement among the TEI-Lex-0 experts.

Figure 6.1 shows an instance of such a scheme in a typical case of nested entry-like constructs in Arabic dictionaries, where dictionary articles are roots that gather all their morphologically derived forms.

In such an example, nesting typed <entry> is required to reflect the hierarchy within the lexical entries and avoid using different representations like <superEntry> or <re>. Such a decision triggered the need to allow such modelling within the TEI Dictionaries modules. Upon exhaustive argumentation in favour, the proposal was adopted and the <entry> model in the TEI guidelines was extended to allow recursive entries and <entry> has been made possible within <sense> construct as related entries might be related to a sense and not the whole entry⁸.

⁷<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-re.html>

⁸<https://github.com/DARIAH-ERIC/lexicalresources/issues/43>



FIGURE 6.1: Left: Dictionary Article in an Arabic Dictionary (Almonjid, 2014). Right: Corresponding Minimal TEI-Lex-0 Encoding

As can be noticed, the values of the <entry> containers provide another classification of the constructs encoded in the nested <entry>s (i.e. *entry@type="wordFamily"* vs *form@type="root"*). It is also worth mentioning that each entry that is not typed has by default "main" value, usually meaning that it carries at least a lemma and other sense information.

Revising the entry model

The existing <entry> model, prior to the TEI-Lex-0 discussion, required further revision to unify the representation of its macro-structures. Besides the required nesting of <entry> and the deletion of <re>, the use of <hom>⁹ for the representation of homographs was deprecated. Two recommendations for a more unified representation of such a phenomenon, by means of either nested <entry> or <sense> elements, were proposed as alternatives. Figure 6.2 shows an example of the first encoding proposal¹⁰ of homographs that have different POS (print version of the entry is provided in Figure 3.3).

Such a representation replaces the use of <hom> elements to encode the entry as two constructs with two different POS (such as a verb and as a noun). The use of <entry> again makes it possible to have a unique representation that helps an automatic system to look for entry constructs that might be inconsistently, or just differently, modelled. Some reluctance was, however, manifested within the TEI-Lex-0 group to the adoption of such a modelling for some dictionaries and the use of <sense>¹¹ construct was suggested instead¹². Besides respecting the original modelling in the print version, such a position can also be understood by the fact that some information

⁹<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-hom.html>

¹⁰<https://github.com/DARIAH-ERIC/lexicalresources/issues/48>

¹¹<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-sense.html>

¹²<https://github.com/DARIAH-ERIC/lexicalresources/issues/14>

```

<entry xml:lang="en" xml:id="act">
  <form>act /ɒkt/</form>
  <entry xml:lang="en" xml:id="actNoun">
    <gramGrp>noun</gramGrp>
    <sense>1. something which is done □ He thanked her for the many acts of kindness she
      had shown him over the years</sense>
    <pc>.</pc>
    <sense>2. a part of a play or show □ Act 2 of the play takes place in the gar-
      den</sense>
    <pc>.</pc>
    <sense>3. a short performance □ The show includes acts by several young
      singers</sense>
    <pc>.</pc>
    <sense>4. a law passed by Parliament □ an act to ban the sale of weapons</sense>
  </entry>
<pc>.</pc>
  <entry xml:lang="en" xml:id="actVerb">
    <gramGrp>verb</gramGrp>
    <sense>1. to do something □ You will have to act quickly if you want to stop the
      fire. □ She acted in a very responsible way. □
      <entry type="relatedEntry" xml:lang="en" xml:id="actAsSomeoneRE">
        <form>to act as someone or something</form>
        <sense>to do the work of someone or something □ The thick curtain acts as a
          screen to cut out noise from the street</sense>
      </entry>
    </sense>
    <pc>.</pc>
    <sense>2. to behave in a particular way □ She's been acting very strangely</sense>
  </entry>
  <pc>.</pc>
  <entry type="relatedEntry" xml:lang="en" xml:id="actTogetherRE">
    <form>to get your act together</form>
    <sense>to organise yourself properly □ If they don't get their act together, they'll
      miss their train</sense>
  </entry>
<pc>.</pc>
</entry>

```

FIGURE 6.2: Minimal Encoding of Homographs as Entries in TEI-Lex-0 (Dictionary Articles from (Publishing, 2009))

related to the highest <entry> structure (e.g. etymology, usage, etc) would be falsely included inside the entry “actNoun”. Excluding the <gramGrp>¹³ from that entry would leave an imbalanced representation with regard to entry “actVerb”. The second encoding proposal, using <sense> element, is depicted in Figure 6.3.

This encoding remains faithful to the modelling choices of the lexicographer of the original print dictionary, despite the inconsistency illustrated in Figure 6.3 and its analysis. But the visibility of the entry-like constructs encoded as sense will be sanctioned, as we have explained the advantage for the alternative encoding in Figure 6.2. To maintain analogy and allow automatic comparability between semantic structures, such an encoding requires wrapping all the first four senses of the entry “actNoun” in an additional <sense> block. We can notice in these cases, the influence of the linear aspect of the lexical data in print dictionaries and the dilemma of choosing one option between different highly supported proposals.

Written and Spoken Forms

The differences of opinion to represent written and spoken forms following the TEI-Lex-0 guidelines are relatively less important and a consensus has

¹³<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-gramGrp.html>

```

<entry xml:lang="en" xml:id="act">
  <form>act /ɒkt/</form>
  <gramGrp>noun</gramGrp>
  <sense>
    <sense>1. something which is done ¶ He thanked her for the many acts of kindness she
      had shown him over the years</sense>
    <pc>.</pc>
    <sense>2. a part of a play or show ¶ Act 2 of the play takes place in the gar-
      den</sense>
    <pc>.</pc>
    <sense>3. a short performance ¶ The show includes acts by several young
      singers</sense>
    <pc>.</pc>
    <sense>4. a law passed by Parliament ¶ an act to ban the sale of weapons</sense>
    <pc>¶</pc>
  </sense>
  <sense>
    <gramGrp>verb</gramGrp>
    <sense>1. to do something ¶ You will have to act quickly if you want to stop the
      fire. ¶ She acted in a very responsible way. ¶
      <entry type="relatedEntry" xml:lang="en" xml:id="actAsSomeoneRE">
        <form>to act as someone or something</form>
        <sense>to do the work of someone or something ¶ The thick curtain acts as a
          screen to cut out noise from the street</sense>
      </entry>
    <pc>.</pc>
    <sense>2. to behave in a particular way ¶ She's been acting very strangely</sense>
  </sense>
  <pc>.</pc>
  <entry type="relatedEntry" xml:lang="en" xml:id="actTogetherRE">
    <form>to get your act together</form>
    <sense>to organise yourself properly ¶ If they don't get their act together, they'll
      miss their train</sense>
  </entry>
  <pc>.</pc>
</entry>

```

FIGURE 6.3: Minimal Encoding of Homographs as Senses in TEI-Lex-0 (Dictionary Articles from (Publishing, 2009))

been reached for a unified encoding of morphological and a large part of grammatical information.

The representation of the lemma is recommended to be within a *form@type* "lemma"/orth construct, which should also hold further information specific to the lemma-like pronunciation. Grammatical information, such as POS, number, or tense, related to the entire lexical entry, should be encoded directly under <entry> by means of a <gramGrp> construct, as illustrated in Figure 6.4.

<pre> <entry xml:lang="en" xml:id="act"> <form type="lemma"> <orth>act</orth> <pc>/</pc><pron>ɒkt</pron><pc>/</pc> </form> <gramGrp> <gram type="pos">noun</gram> </gramGrp> ... </entry> </pre>	<pre> <entry xml:lang="en" xml:id="act"> <form type="lemma"> <orth>act</orth> <pc>/</pc><pron>ɒkt</pron><pc>/</pc> </form> <gramGrp> <pos>noun</pos> </gramGrp> ... </entry> </pre>
--	---

FIGURE 6.4: Left: POS Encoding Using <gram> Element. Right: POS Encoding Using <pos> Element

As it can be noticed, two alternatives are provided to encode grammatical

information by means of either a simplified *gramGrp/gram@type* or *<gram-Grp>* and an explicit TEI element. The proposal¹⁴ to choose between one of these two alternatives requires further discussion. These two recommendations are also valid for encoding grammatical information within *<sense>* structures.

For encoding information related to variants or inflected forms, it is recommended to use the *<form>* construct as shown in Figure 6.5.

<p>caa dzinda:¹ tenazas caa dziñe yocanindi: dar de agudo caa dziñendi, futuro cavua: echado estar de lado</p>	<pre> <entry> <form type="lemma"> <orth>caa dzinda</orth> </form><pc>:</pc> <sense> <def>tenazas</def> </sense> </entry> <entry> <form type="lemma"> <orth>caa dziñe yocanindi</orth> </form><pc>:</pc> <sense> <def>dar de agudo</def> </sense> </entry> <entry> <form type="lemma"> <orth>caa dziñendi</orth> </form><pc>,</pc> <form type="inflected"> <gramGrp> <gram type="tns">futuro</gram> </gramGrp> <orth extent="part">cavua</orth> </form><pc>:</pc> <sense> <def>echado estar de lado</def> </sense> </entry> </pre>
---	---

FIGURE 6.5: Left: Extract from Mixtec-Spanish Dictionary (Alvarado, 1593) Containing Inflected Forms. Right: Corresponding Encoding in TEI-Lex-0

In the Mixtec-Spanish example 6.5, we can see how the tense has been encoded within the *<gramGrp>* construct for an inflected form, whose orthography has been specified as partial in the attribute of *<orth>*¹⁵ element.

Further details on more recommendations about modelling morphological and grammatical information have been described by Banski, Bowers, and Erjavec, 2017.

¹⁴<https://github.com/DARIAH-ERIC/lexicalresources/issues/31>

¹⁵<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-orth.html>

Miscellaneous

The TEI-Lex-0 experts could make great progress on the representation of other aspects of the lexical description. The scope of the `<usg>`¹⁶ element, which allows the encoding of usage information, has undergone a drastic restriction of its scope and more control over the classification of its possible types. `usage@type="language"` has been deprecated and such use should be encoded by means of the `<lang>`¹⁷ element. Furthermore, a unified list of possible values of `usage@type` has been compiled by taking into account the existing suggested values in TEI P5 and the classifications defined by Atkins and Rundell, 2008 and Svensén, 2009.

To encode prose information that lacks structure or does not fall within the scope of the possible TEI elements allowed at the encoding level of the parent element, a lexicon designer is allowed, following the TEI-Lex-0 guidelines, to make use of the `<dictScrap>`¹⁸ element. Moreover, and as it has been shown in the encoding examples of this section, the `<pc>`¹⁹ element is recommended to be used to tag the separators between different physical structures, such as dots, commas, parentheses, etc. Besides the need to exclude such separators, as shown for pronunciation in Figure 6.4, such an encoding helps a parser to learn the features marking the transition between two fields. We will develop this idea more in Chapter 7.

The work on the TEI-Lex-0 guidelines is still ongoing and ideas should be further refined, as for the mentioned pending decisions, as well as for other important constructs like etymology and collocations.

6.2.3 Serialisation Model

To summarise the major decisions that have so far been made for the recommendations of TEI-Lex-0, we present a UML diagram that sketches the main TEI components of a lexical entry and their different relationships as they are defined in the actual guidelines.

Entry Class Diagram

Each class in the diagram presented in Figure 6.6 represents a TEI element that holds part of the definition of a lexical entry. Elements from modules other than the TEI dictionaries chapter are not represented in this diagram.

The serialisation model of TEI-Lex-0 represents a rich constellation of TEI elements. `<entry>`, being the central component and allowing recursivity, can be composed of:

- `<gramGrp>`: groups grammatical information, such the grammatical function (i.e. pos), number, gender, etc..

¹⁶<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-usg.html>

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

¹⁸<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-dictScrap.html>

¹⁹<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-pc.html>

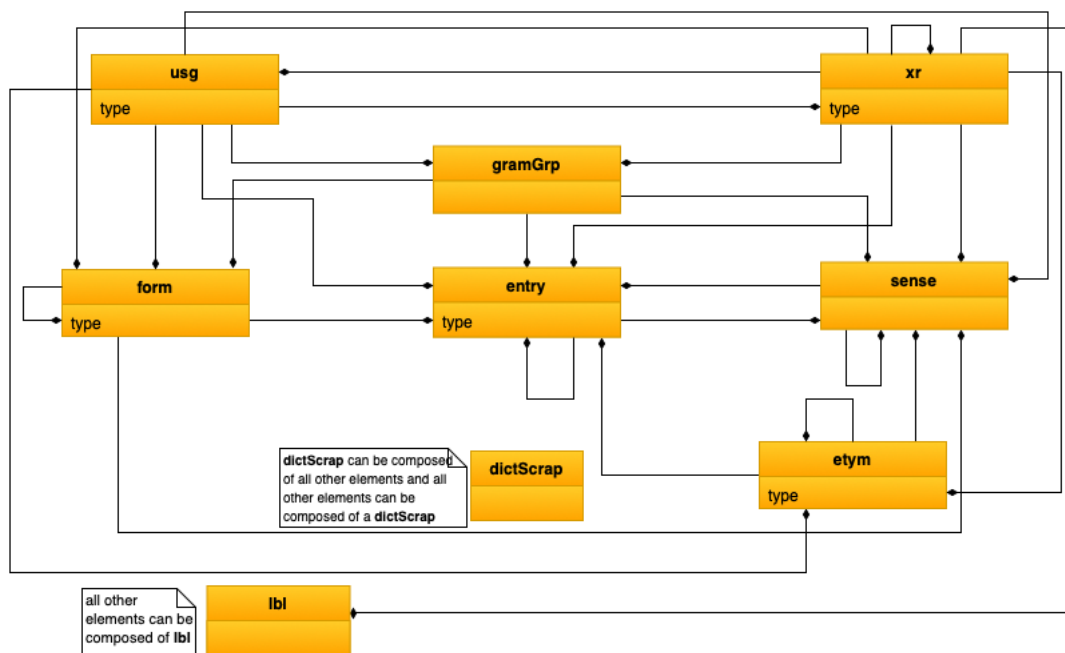


FIGURE 6.6: TEI-Lex-0 Serialisation for <entry> Model

- `<xr>`: enables the encoding of cross-references within all the components of a lexical entry, even itself (recursive). The mapping mechanism is enabled by means of type (i.e. `xr@type`) to express the relationship between the source and a target encoded with `<ref>`²⁰.
- `<usg>`: contains usage information of a higher level component that might be an `<entry>` and all its sub-elements. Typing `<usg>` is recommended to specify the category of the modelled usage.
- `<form>`: holds morphological information of an entry or its components. It can be recursive and might comprise `<gramGrp>`, `<usg>` and `<xr>` constructs.
- `<sense>`: contains the semantics of an entry and can accommodate all its components and even an entire entry structure (e.g. the case of related entries).
- `<etym>`²¹: allows the encoding of etymological and diachronic description. It can be recursive and permits the expression of usage and cross-references by means of respectively `<usg>` and `<xr>` constructs.
- `<dictScrap>`: as has been previously explained, this element could be employed to tag less structured text segments within any element of the model.
- `<lbl>`²²: this element is often used within an `<xr>` structure to encode labels triggering a cross-reference, like "See." in English, "Voir" or "V."

²⁰<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/examples-ref.html>

²¹<https://www.tei-c.org/release/doc/tei-p5-doc/de/html/ref-etym.html>

²²<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-lbl.html>

in French, etc. It can also be used everywhere to designate any text segment that has a label function.

Discussion

The resulting model has managed to reduce alternatives for encoding several categories of lexical information and the scope of TEI key elements of the dictionaries chapter. But, despite the revised scope and new recommendations for a unified use of elements, the model remains complex with many bidirectional and recursive relationships. Such a fact represents an obstacle to finding a unified scheme for heterogeneous lexica, as the joint encoding possibilities can grow exponentially. In other terms, a query system, developed for querying different resources following the TEI-Lex-0 scheme, might miss the extraction of some embedded structures (e.g. <entry>s, <sense>s, etc) as there is no nesting limit. Such an issue can be resolved by defining multiplicities of the relationships which are missing in the guidelines. We do not know, for example, what represents the minimal structure of an entry (e.g. a lemma, a form, a lemma and a sense,..). Such constraints can ensure sanity check and prohibit falsely encoded structures that might be the result of a human manipulation of a wrongly defined transformation scheme from an existing TEI resource.

From our experience in the TEI-Lex-0 group, we have also noticed reservations about making important changes to the model of some problematic existing TEI structures and practices. In addition, the order of the occurrence of lexical information in a print dictionary dictates the constraints for the revised encoding which reduced the scope for changing old practices and left too much choice than if the approach had been based on the logical structure of a lexicon. In Section 6.3, we show how such constraints are relaxed in a different approach, where the NLP requirements have more control on the modelling choices of the lexical structure over the documentation and human readability needs.

6.3 LMF Reloaded

As an ISO standard, LMF is the subject of revision every few years. The process is triggered by a dedicated ballot and the availability of volunteering experts who will lead the revision project. In this section we report the features of the new version of the standard as a result of the recent revision work within the *ISO/TC 37/SC 4/WG 4* committee.

6.3.1 Context

In Section 5.3, we showed how the version of LMF, published 2008, suffered from several shortcomings at the structure and the content levels. These limitations were the motive for the revision work that started in 2015 and which we joined in 2017. It was decided to structure the standard into several parts with more focused scope and total abstraction of the meta-model from its

serialisation. A number of simplifications and improvements have been carried out to reduce the complexity of the modelling. In addition, the connection with leading standards, in particular TEI, has allowed LMF to gain more simplicity and acquire a flexible serialisation model that enables interoperability and exchange with the abundant number of TEI-based resources.

We present below the new features of the standard, with a special focus on the modelling principles and serialisation, as well as the progress made and our criticism of the resulting work.

6.3.2 Modelling Challenges

The LMF version (Francopoulo et al., 2006), the subject of the ongoing revision, tried to bring structure into the meta-model and its description by organising them into different extensions that complement a core model. The definition of such packages and the conformity of the classes they contain lacked precision and the separation among extensions was blurry in some cases.

The experts within the *ISO/TC 37/SC 4/WG 4* committee took up, firstly, the restructuring of the standard. Secondly, they worked on simplifying the existing classes and their relationships. Finally, the meta-model has been enriched with new classes for existing and newly introduced extensions, resulting from the restructuring step.

Our main contribution was focused, at a first stage, on the consistency management of the UML abstract models during the *Restructuring* and *Simplification* processes. In addition, we were involved in finding a suitable TEI serialisation of the resulting models.

Restructuring

The restructuring project was initially based on the already existing extensions that came as a whole in the standard. A user who might be interested in only parts of the standard had to get the entire package and pick up what he/she needs. The organisation of the standard has been thoroughly discussed and revised to create a new structure based on complementary modules. The multi-part standard now has the following structure:

- ISO 24613-1 – LMF Part 1 – Core Model: this represents the minimal structure of a lexicon where the defined classes provide a lexicon designer with the necessary elements to model basic lexical information, like lemma, form and sense.
- ISO 24613-2 – LMF Part 2 – Machine Readable Dictionary (MRD) Model: this part complements the core model by means of classes that bring more precision to the semantic information, like examples and their textual representation, and allows a differentiation between written and spoken forms.

- ISO 24613-3 – LMF Part 3 – Etymological Extension: this is a new extension that carries a whole new range of information describing etymology and diachrony of lexical information.
- ISO 24613-4 – LMF Part 4 – TEI Serialisation: this part defines mappings between the defined classes in the first three parts of the standard and constructs in the different TEI P5 modules, with a focus on the dictionaries chapter.
- ISO 24613-5 – LMF Part 5 – LBX Serialisation: a second serialisation is proposed in this module using the Language Base Exchange (LBX), which had been introduced earlier by George, 2013 as an external application of the standard.
- ISO 24613-6 – LMF Part 6 – Syntax and Semantics Extension: this part has the role of providing deep modelling of semantics and syntax along with their mutual relationships. The work on this extension is mainly focused on remodelling the existing extension by following the new principles defined in Parts 1 and 2.
- ISO 24613-7 – LMF Part 7 – Morphological Extension: this extension has also a great input from what had been already defined for the old version of LMF, touching on the deep modeling of morphology. Enriching the restructured morphological classes and patterns with more precision and new classes is on the agenda of the LMF experts.

Simplification

The emphasis on abstraction and modularisation led to a series of major simplifications affecting several classes of the meta-model. One key feature that has been recently introduced is the *CrossREF* class which is a pointing/mapping mechanism that can be used to model a wide array of lexical features and relationships such as semantic relations, cross references, related entries and others within the meta-model. As a result, some classes (e.g. List of Components and Component) whose features have been taken on, in part, by *CrossREF* have been removed altogether. Figure 6.7 illustrates the simplicity of the new mechanism used to model an MWE previously represented by classes which are now obsolete (see Figure 5.4).

Enrichment

New information has been introduced to describe essential aspects of lexical information such as *Bibliography*. Such information is required to specify references for some usages, definitions, examples, etc. Therefore the new class is kept multi-functional to be used in case of need as determined by the editor of a lexicon. Additionally, the differentiation of *Orthographic Representation* into *Form Representation* and *Text Representation* has been designed to enable more precision in the encoding of written forms touching on respectively *Sense* and *Form* sub-classes.

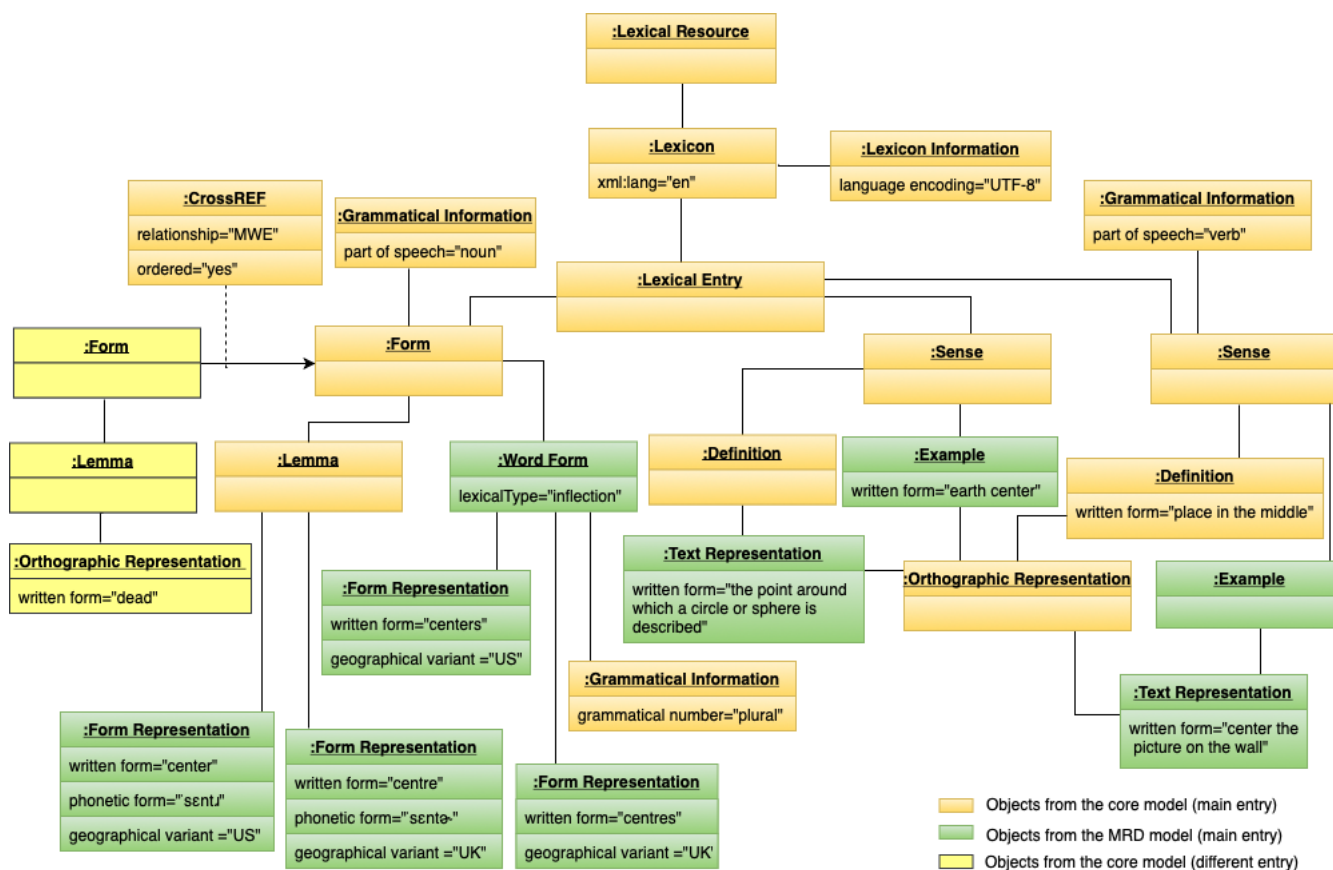


FIGURE 6.7: Example of Modelling the MWE "DEAD CENTRE" and its Components using the New LMF Core and MRD Models

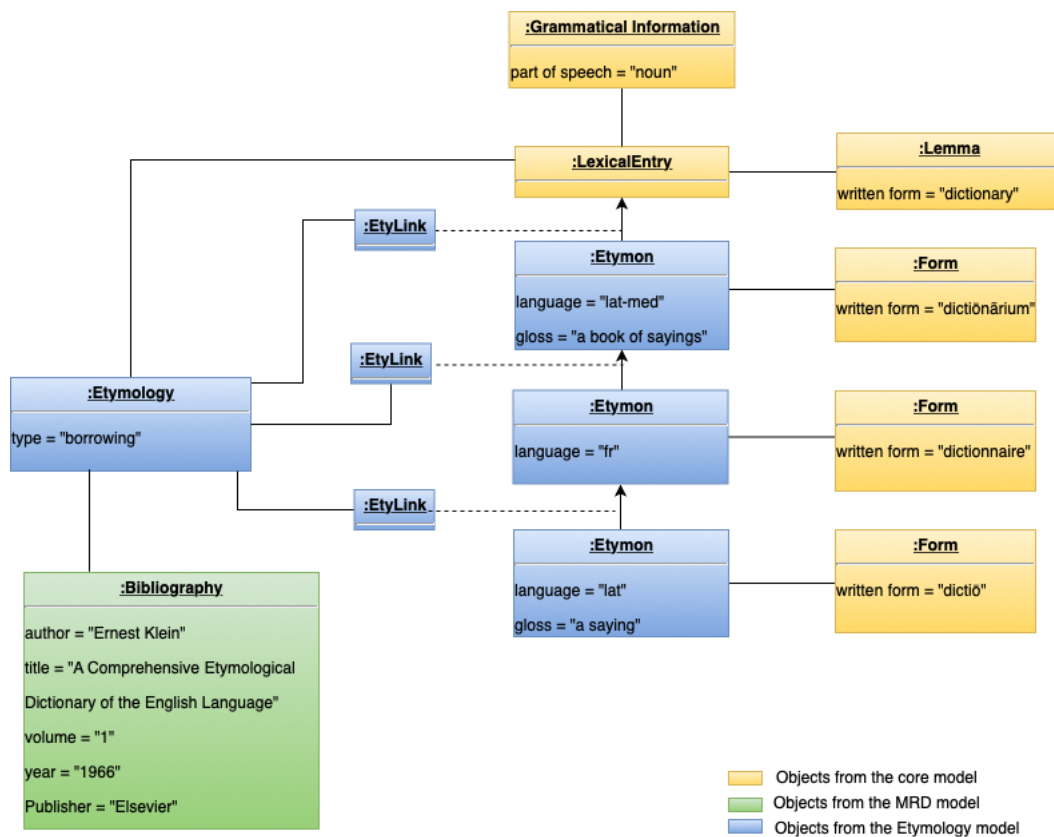
Figure 6.7 illustrates an instance of the new LMF meta-model for Parts 1 and 2. We can notice how the restructuring comes along with a revision of class membership; for example, the Lemma class, which was previously based in the MRD part (see Figure 5.4) is now part of the core model as it is a fundamentally essential part of a lexicon.

As a second enrichment aspect, Etymology and Diachrony come out in the new version of LMF with a whole range of information useful for the study of words and their origins, designed for the NLP usage. It is beyond the scope of this thesis to give an exhaustive description of the decisions made to shape this new extension, but we can give a flavour of the new dimension that such a package has given to the meta-model.

Figure 6.8 shows an instance of the object diagram of the *Etymology* extension in connection with the *Core* and *MRD* packages. We can observe the complementarity of the three extensions for the description of etymological links among different forms. *Etymon* and *Etylink* represent the core mechanisms for expressing an etymological relationship, in this case *borrowing*.

dictionary, n. — ML. *dictiōnārium* (whence also F. *dictionnaire*), prop. ‘a book of sayings’, fr. L. *dictiō*, gen. *-ōnis*, ‘a saying’. See prec. word and subst. suff. **-ary**.

(a) Print Version of the Dictionary Article



(b) Partial UML Object Diagram

FIGURE 6.8: Instance of the New Etymology Extension for the Entry "DICTIONARY" in (Ernest, 1966)

6.3.3 TEI Serialisation Model: ISO 24613-4

The resulting meta-model can be serialised by using any language that suits the needs of the user of the standard, like XML or SQL. Given the fact that TEI benefits from widespread adoption within the lexicographic community, the LMF expert committee has decided to connect with the de facto standard by proposing an XML serialisation based on mapping between the LMF classes and TEI elements.

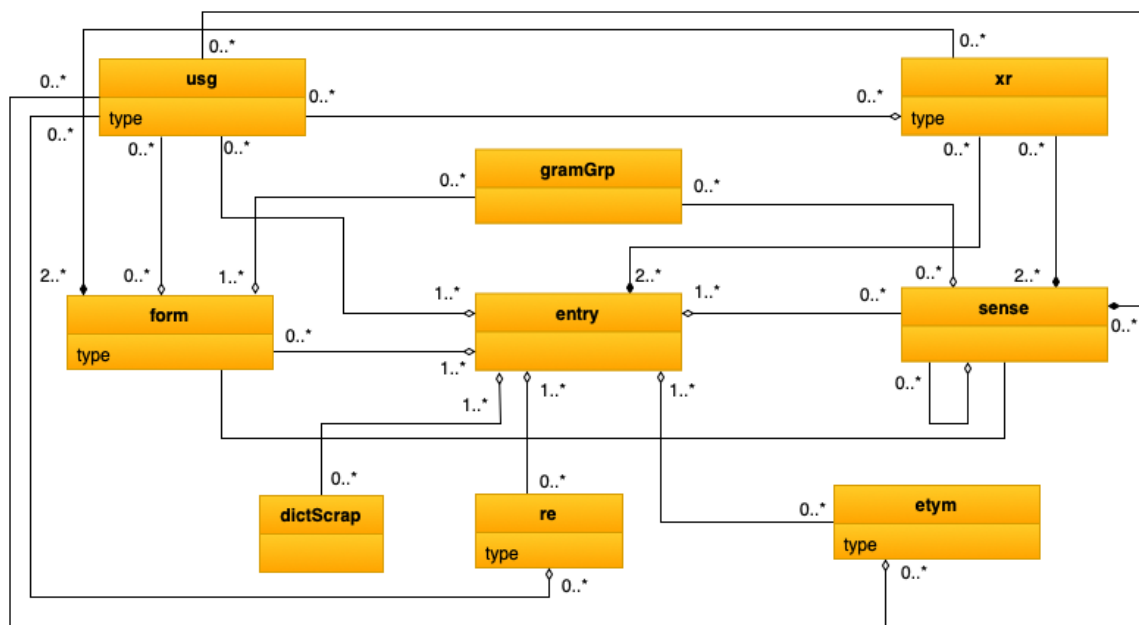


FIGURE 6.9: LMF Serialisation for <entry> Model

In Figure 6.9, we introduce a novel model representing key components of the new TEI serialisation extension. The UML class diagram represents a serialisation of the meta-model of the lexical entry structure by means of TEI elements. For the proposed serialisation, unequivocal mappings between the LMF meta-model classes and TEI P5 elements are kept to a maximum. Where one-to-one alignments are not possible, we opt for the translation of the LMF classes into TEI constructs that can be composed of more than one element or/and element types.

The <entry> element, representing the serialisation of the Lexical Entry class in the meta-model can contain:

- <xr>: to allow a cross-reference to other lexical entries having a relationship with the source entry, such as synonym, antonym or translation equivalent that do not yet have senses to map to. This element also comes to replace old mapping mechanisms within the entire meta-model, as it can be activated to enable the cross-references among nearly all the elements of the serialisation. The mapping mechanism is enabled by means of (i) typing *xr@type*, to specify the relationship, (ii) <ref> element to contain the text of the lemma or phrase representing the target, (iii) and specifying the ID (i.e. *ref@corresp*) to link to an internal

or external target, if it exists, that can be typed (i.e. *ref@type*) with the category of the target (e.g. entry, sense, etc).

- `<usg>`: contains subject field information of an `<entry>` and all its components. The typing of `<usg>` is recommended to specify the category of the modelled usage and the idea is to converge to the same list of categories that has been compiled for TEI-Lex-0.
- `<form>`: holds morphological information of an entry. Typing is the key mechanism to differentiate between the possible forms that might be, in the first place, the lemma, along with other morphological structures like inflected forms, stem, root, etc. It might also contain grammatical information, encoded within `<gramGrp>`, as well as the `<usg>` and `<xr>` constructs.
- `<sense>`: serialises a sense of an entry and allows recursivity to contain sub-senses. It represents the provision of sense information such as `<def>`²³ for sense definition, `<cit>`²⁴ typed with example or translation equivalents or usage information encoded within `<usg>`. It might contain grammatical information encoded within a `<gramGrp>` element and can reference other senses (i.e. synonyms, antonyms, etc) by means of the `<xr>` mechanism. In some cases, when an entry has multiple forms and senses (e.g. the case of homographs) and the grammatical information is specified only for the forms, a link between the `<sense>` and `<form>` constructs needs to be established to differentiate the belonging of the senses (see Figure 6.10). Such a detail is crucial for NLP application where the missing link can be propagated to the syntax of the lexical entry.
- `<etym>`: is the entry point to the etymological description modelled in the third part of the standard. *etym@type* carries the type of etymology description, such as borrowing, inheritance, etc. It may contain the `<usg>` and `<xr>` constructs.
- `<dictScrap>`: contains elements of the lexical descriptions that do not fall into any of the previous categories but is allowed only for the lexical entry. It can be used by a lexical designer to markup information that he/she wants to exclude from indexing.
- `<re>`: has the same logical structure as the `<entry>` element in the actual specifications of the standard. This element is used to encode derived and related forms and can also be used as an alternative mechanism to encode MWEs (see Figure 6.11). There are, however, strong proposals to replace it by a recursivity of the `<entry>` element.

Figure 6.11 shows an example of a serialisation of the MWE “dead center” and its components. We can see how the `<form>` element is enriched by

²³<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-def.html>

²⁴<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-cit.html>

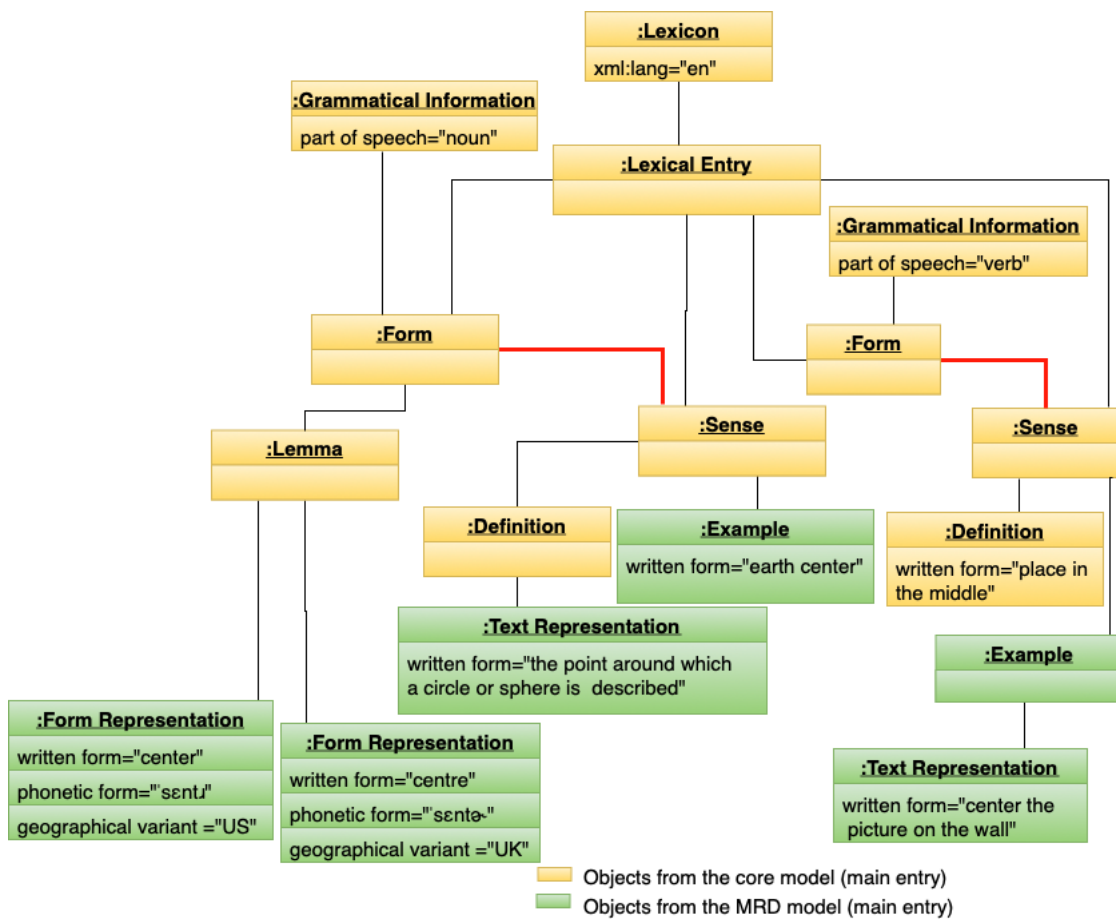


FIGURE 6.10: Example of LMF Modelling of Homographs with two Form Objects


```

<entry xml:lang="en">
  <form type="lemma" xml:id="center_form">
    <orth>center</orth>
    <pron>'sentɹ</pron>
    <gramGrp>
      <pos>noun</pos>
    </gramGrp>
    <usg type="geo">U.S</usg>
    <form type="variant">
      <orth>centre</orth>
      <usg type="geo">U.K</usg>
      <pron>'sentə</pron>
    </form>
  </form>
  <form type="inflected">
    <orth>centers</orth>
    <usg type="geo">U.S</usg>
    <gramGrp>
      <number>plural</number>
    </gramGrp>
  </form>
  <form type="inflected">
    <orth>centres</orth>
    <usg type="geo">U.K</usg>
    <gram type="number">plural</gram>
  </form>
  <sense>
    <def>the point around which a circle or sphere is described</def>
    <cit type="example">
      <quote>earth center</quote>
    </cit>
  </sense>
  <sense>
    <gramGrp>
      <pos>verb</pos>
    </gramGrp>
    <def>place in the middle</def>
    <cit type="example">
      <quote>center the picture on the wall</quote>
    </cit>
  </sense>
  <re type="multiWordExpression">
    <form>
      <seg corresp="#dead_form" n="1">dead</seg>
      <seg corresp="#center_form" n="2">center</seg>
    </form>
  </re>
</entry>

```

FIGURE 6.11: Example of the new LMF Serialisation for the MWE "DEAD CENTER" and its Components

means of fine-grained elements such as `<orth>`²⁵ for lemma, `<pron>`²⁶ for pronunciation and `<seg>`²⁷ for components of the MWE. The latter represents a use of the second mapping mechanism of the proposed standard, which allows a pointing to other forms by means of `@corresp` attribute.

6.3.4 Discussion

The new version of LMF brought new changes to the existing extensions, that helped to improve them, while they persisted to be mutually complementary with the other modules of the standard. The new serialisation answers more to the encoding requirements of print dictionaries, as it has adopted TEI guidelines and practices. In parallel, the new meta-model translated into TEI constructs, remained focused on how the lexical description could be modelled to carry the richest possible different information in the clearest and more simplified way.

The differences in modelling the same lexical structures are less important than those in TEI-Lex-0, as recursivity has been drastically reduced and mutual inclusion between elements has been disabled. However, different alternatives are still possible to model the same information like the case for homographs (see Figure 6.7 and Figure 6.10). In fact, the discussions within the expert committee were mostly focused on stabilising the core and the MRD models. The etymology model and serialisation modules managed to achieve relatively important progress, given the fact they are totally new. The work on the TEI serialisation extension was mostly focused on finding the mapping between the two standards. Consequently, the ongoing and future revisions should build upon the resulting serialisation model and define more restrictions on the scope of each employed TEI element to be fully compliant with the new LMF specifications. The serialisation examples made publically available would also help to obtain feedback from the community and more use cases for future discussions of the ISO committee.

6.4 GROBID-Dictionaries Output Scheme

GROBID-Dictionaries takes into account the recommendations in the TEI-Lex-0 and LMF to organise the recognition of the different lexical structures and their serialisation. But the modelling and the serialisation in the novel system also come up with explorative suggestions that were influenced by the use cases we encountered during the annotation of different samples in the experiments carried out.

In this section, we present main modelling challenges, the resulting serialisation of the CRF models introduced in Section 4.3.1 and their serialisation.

²⁵<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-orth.html>

²⁶<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-pron.html>

²⁷<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-seg.html>

6.4.1 Modelling Challenges

The modelling challenge at this stage lies in the translation of all the recommendations dictated by lexicographic practices - TEI-Lex-0 - and NLP standardisation requirements - LMF - into CRF models able to extract lexical information and serialise it into a TEI scheme that is compliant with the novel standardisation schemes. The workflow first begins by guaranteeing a mapping to TEI compliant output. Then we study the possibility of converging to a scheme that maximally satisfies the requirements of TEI-Lex-0 and LMF.

Cascading Modelling

In section 4.2, we have already explained that the architecture of the cascading CRF models is defined by the ability to match TEI elements in a target scheme with the labels of each model. Avoiding the extraction of constructs that have different granularities is a requirement for the definition of the labels of each model.

We explain this further by illustrating the process of defining the model chain for the extraction of different structures from entries in an etymological dictionary. Figure 5.1 shows an entry from the print version of the dictionary. For this example, we rely on a TEI encoding defined by a lexicographer and neither the TEI-Lex-0, nor the LMF guidelines are followed. An etymology extension is the models chain responsible for parsing diachronic information encapsulated in an <etym> block detected by the **Lexical Entry** model and allows the clustering of specific constructs into TEI sub-blocks. Each sub-block corresponds to a TEI compliant element whose significance in this use case is described in the following table:

TEI Element	Description
<seg>	unclassified parts of an entry
<quote>	a quotation, text quoted from another source (typically from another dictionary)
<def>	a definition, i. e. a semantic description (gloss or paraphrase) associated with a given wordform
<lang>	a language identifier (often abbreviated)
<mentioned>	a mentioned wordform in any language, i. e. a wordform that is considered in the etymological description
<bibl>	a bibliographic citation (typically a scribal abbreviation)

TABLE 6.1: Main labels of the etymology extension

Projected on the print sample, such constructs have a sparse distribution over the etymological description of a lexical entry (see Figure 6.12).

Where <def>, <lang>, <mentioned>, <seg> and <bibl> structures are represented respectively in red, orange, green, black and blue, a <quote> construct has more complex content illustrated by the whole underlined segment in the above entry.

Cabbage 1. *kohl*; *altengl. cabbage*, bei Hal. 226 *cabes*, *cabishes*: *mlat. gabusia*, *fr. cabus*, *it. cappuccio*; vgl. *ndl. cabuis*, *cabuyscoole*, *nhd. kappes*, worüber Weigand 1, 562: „*Im vocab. incip. teut. ante lat. kabbas*, *mhd. der kapa3*, *kape3*, *spätahd. kabu3*, *capu3*. *Aus fr. der cabus*, *it. capuccio*, *welches wie russ. die kapusta kohl*, *aus mlat. caputium kapuze hervorging und der geschlossene kohl schien einer mönchskappe ähnlich*;*“* vgl. Diez 1, 110 und unter den *nhd. kabisz*, *kabis* Grimm 5, 9.

FIGURE 6.12: Etymological Constructs in (Mueller, 1878)

The issue of having scattered information to be recognised in the presented use case has been solved by the effectiveness of the chosen machine learning technique. An experiment on a first etymological CRF model trained with carefully annotated data, gives very accurate results for the extraction of <def>, <lang>, <mentioned> and <bibl> components. However, the recognition of complex structures such as <quote> within the whole etymological description represents a second bottleneck. In the case of Cabbage, extracting a <quote> segment and its encapsulated etymological components requires a cascading processing. Our solution is to start by extracting complex segments by a first model (see Figure 6.13). The resulting <seg> and <quote> blocks will be parsed by a second model to extract the elementary structures (see Figure 6.14).

This use case showcases the need to have 2 models to parse the sparse etymological constructs. We name the first *Etym/Quote*, as it has the goal of isolating the <quote> blocks from the rest of the structures. The second model we name it *Etym*, since it allows the full parsing of the pre-extracted structure by the first etymological model. The Etym model can be recursive to parse the blocks generated by a higher level Etym model. These two models form the etymology extension we define in Figure 6.15.

```
<entry>
  <form>
    <orth>Cabbage</orth><label>1.</label>
  </form>
  <etym>
    <seg>kohl; altengl. cabbage, bei Hal. 226 cabes,cabishes: mlat.
      gabusia, fr. cabus, it. cappuccio; vgl. ndl. cabuis,
      cabuyscoole, nhd. kappes, worüber Weigand 1, 562:
    </seg>
    <quote>„Im vocab. incip. teut. ante lat. kabbas, mhd. der
      kapa3, kape3, spätahd. kabu3, capu3. Aus fr. der cabus, it.
      capuccio, welches wie russ. die kapusta kohl, aus mlat.
      caputium kapuze hervorging und der geschlossene kohl schien
      einer mönchskappe ähnlich;“
    </quote>
    <seg>vgl. Diez 1, 10 und unter den nhd. kabisz, kabis Grimm 5, 9.
    </seg>
  </etym>
</entry>
```

FIGURE 6.13: Minimal TEI Encoding as an Output of the Etym/Quote Model

```

<entry>
  <form>
    <orth>Cabbage</orth><label>1.</label>
  </form>
  <etym>
    <seg><def>kohl</def>;<lang>altengl.</lang>
      <mentioned>cabage</mentioned>, <seg>bei</seg>
      <bibl>Hal. 226</bibl><mentioned>cabes</mentioned>,
      <mentioned>cabishes</mentioned>: <lang>mlat.</lang>
      <mentioned>gabusia</mentioned>, <lang>fr.</lang>
      <mentioned>cabus</mentioned>, <lang>it.</lang>
      <mentioned>cappuccio</mentioned>;
      <seg>vgl.</seg><lang>ndl.</lang>
      <mentioned>cabuis</mentioned>,
      <mentioned>cabuyscoole</mentioned>, <lang>nhd.</lang>
      <mentioned>kappes</mentioned>, <seg>worüber</seg>
      <bibl>Weigand 1, 562</bibl>:
    </seg>
    <quote>„Im <bibl>vocab. incip. teut.</bibl> ante
      <lang>lat.</lang> <mentioned>kabbas</mentioned>,
      <lang>mhd.</lang> der <mentioned>kapa3</mentioned>,
      <mentioned>kape3</mentioned>, <lang>spätahd.</lang>
      <mentioned>kabu3</mentioned>,
      <mentioned>capu3</mentioned>. Aus <lang>fr.</lang>
      der <mentioned>cabus</mentioned>, <lang>it.</lang>
      <mentioned>capúccio</mentioned>, welches wie
      <lang>russ.</lang> die <mentioned>kapusta</mentioned>
      <def>kohl</def>, aus <lang>mlat.</lang>
      <mentioned>caputium</mentioned> <def>kapuze</def>
      hervorging und der geschlossene kohl schien einer
      mönchskappe ähnlich;“
    </quote>
    <seg><seg>vgl.</seg> <bibl>Diez 1, 10</bibl>
      <seg>und unter den</seg> <lang>nhd.</lang>
      <mentioned>kabisz</mentioned>,
      <mentioned>kabis</mentioned> <bibl>Grimm 5, 9</bibl>.
    </seg>
  </etym>
</entry>

```

FIGURE 6.14: Minimal TEI Encoding as an Output of the Etym Model

Satisfying different encodings for the same structures

Our goal, to find a unequivocal mapping between the structures recognised by the CRF models and the TEI-Lex-0 and LMF guidelines, has been challenged by the fact that, in some cases, more than one alternative has been defined in each of these two formats for encoding same structures (e.g., the case of homographs, explained in Section 6.2.2).

Such a need pushed us to study different heuristics to reach a target encoding by allowing required labels for the suitable models in different modelling scenarios. From a user perspective, the activation of such labels and the generation of one of the wanted target encoding is possible by following two steps:

- After choosing the chain of models permitting to reach a structure and the depth of the target TEI output, the user proceeds by annotating the training data for the corresponding models.
- After training each model properly, the user can compose the output he/she wants by selecting in the web application (see Figure 4.8) the models he/she trained according the scenario he/she chose.

Following this setup, we managed to enable two encodings for modelling structures that could be interpreted differently. For instance, we managed to allow the modelling of homographs following two different scenarios. The first scenario, having the output presented in Figure 6.2 as the target, should allow a “Lexical Entry” model to recognise sub-entries within `<entry>` and then parse the sub-entry, serialised as `<entry>` (i.e. the entry having as ID “actVerb”), by using a second entry model. The second scenario, followed to generate the output of Figure 6.3, relies instead on a “Sense” model to parse a `<sense>` block and isolate `<gramGrp>` from senses that will be parsed by a “Sub-Sense” model to extract related entries and other components like definitions and examples (not represented in the figure). The two figures, representing TEI-Lex-0 serialisation instances, recommend the use of typed `<entry>` to represent a related entry. The models of GROBID-Dictionaries were not all implemented in parallel with the discussions of the TEI-Lex-0. As the decision to use `<entry>` for all entry-like constructs is relatively recent, related entries are still serialised as `<re>` the models which remain an LMF compliant choice.

The guidelines for encoding the two formats, TEI-Lex-0 and LMF, are also not fully compatible. For instance, there is a difference between modelling a recognised `<gramGrp>` block related to the entire entry, such as the POS. We can see in Figure 6.3, that TEI-Lex-0 recommends the encoding such a `<gramGrp>` as a direct child of `<entry>`, whereas in LMF, such an encoding is not allowed by the serialisation model (see Figure 6.11) and it is only allowed inside the `<form type="lemma">` element. We chose to allow the first alternative, as it reflects the physical structure of most of the dictionaries we tested and our users give more importance such a representation. Encoding such a logical structure according to the LMF serialisation model remains,

however, easy to implement thanks to a dedicated separation between the recognition and the serialisation stages that we made sure to have.

Further details about the encoding supported by GROBID-Dictionaries are summed up in the serialisation model presented in Figures 6.15 and 6.16.

6.4.2 Serialisation Models

The parsing models within GROBID-Dictionaries generate two XML serialisations: the first is for internal use and the second represents the final output that results from the call of the REST services through the web application facet (see Figure 4.8).

Internal Serialisation

The first serialisation, illustrated in Figure 6.15, uses XML elements that in many cases share the same labels as homologous TEI elements. This serialisation is used to annotate data, train and evaluate the CRF models. The reason behind the use of labels different from those existing in the TEI guidelines is to give more semantics the tagged components. For instance, we make use of `<fromGramGrp>` to denote `<gramGrp>` related to the entire entry and to differentiate such a component from `<senseGramGrp>` which represents a `<gramGrp>` related to a sense. There are also some cases where we use labels for elements, that might be confused with TEI elements, but we use them to denote different structures. For example, we use `<body>` element to mark the body part of page whereas `<body>`²⁸ in the TEI guidelines is usually used to mark the entire body matter of a monograph or a dictionary.

We present below both serialisations of the elements of each model in the architecture. In the remainder of this section, by *serialised in* we mean the element serialised internally and transformed into the second (i.e. final) serialisation as:

- **Dictionary Segmentation:** The first model of the architecture has three main labels:
 - `<headnote>` for header and head-note information, *serialised in* `<fw type="header">`²⁹
 - `<body>` for all the text area containing the lexical entries of a page, *serialised in* `<ab>`³⁰
 - `<footnote>` for footer and footnote information *serialised in* `<fw type="footer">`
- **Dictionary Body Segmentation:** The second model parses a `<body>` block, recognised by the first model, and processes it to recognise `<entry>` constructs that are the same in both serialisations.

²⁸<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-body.html>

²⁹<https://www.tei-c.org/release/doc/tei-p5-doc/de/html/ref-fw.html>

³⁰<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-ab.html>

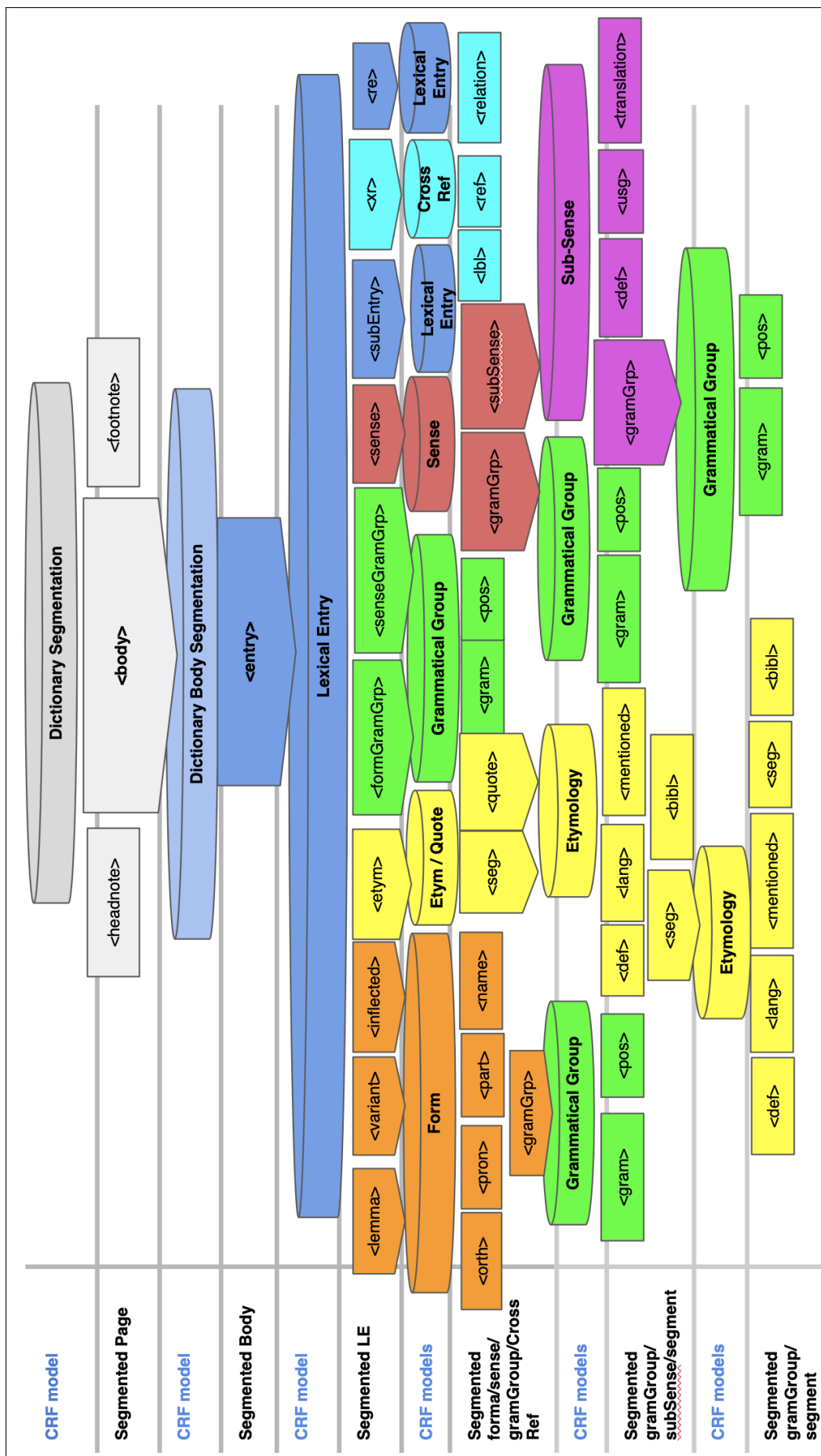


FIGURE 6.15: Internal Serialisation of GROBID-Dictionaries's Models

- **Lexical Entry:** The third model parses each <entry>, to segment it into:
 - <lemma> *serialised in*, <form type="lemma">
 - <variant> *serialised in*, <form type="variant">
 - <inflected> for morphological and grammatical information related to a "Form", *serialised in* <form type="inflected">
 - <senseGramGrp> for grammatical information related to a "Sense", and <formGramGroup> for a general grammatical description, are both *serialised in* <gramGrp>
 - <subEntry> for an embedded entry, *serialised in* <entry>
 - <reference> for a reference to another entry, *serialised in* <xr>
 - <sense> for semantic information, <re> for a related entry, <etym> for etymological description, all have the same labels in both serialisations.
- **Form:** This model analyses blocks representing a "Form" such as <lemma>, <variant>, or <inflected>, generated by the Lexical Entry model, and segments the information it contains. The current list of possible labels for this model contains:
 - <orth> to contain the orthography of a "Lemma" or a "Variant", <pron> for pronunciation, <gramGrp> for grammatical information, such as part of speech, gender, number, etc, <lang> for language information about a "Form", <name> for headwords in encyclopedic dictionaries, <desc> to encode the brief description coming after the headword of an article and <usg> for usage information of the analysed form. All these labels remain the same for both serialisations
 - <part> to contain the extent of an orthography of an "Inflected Form", *serialised in* <orth type="part">
- **Grammatical Group:** This model has the task of parsing a <gramGrp> wherever it appears in the architecture. The actual list of labels contains:
 - <pos> for POS, <gram> for a piece of grammatical information that is going to be typed, <tns> for tense of a verb, <gen> for gender information, <number> for number, <subc> for information about transitivity, countability, etc., There is no difference in labels between the two serialisations for these elements.
- **Cross Ref:** Such a model makes it possible to parse cross references <xr> wherever they appear in the architecture. The main labels predicted by this model are:
 - <ref> for an internal or external reference, and *serialised in* <xr>
 - <relation> to type the reference instance (e.g. synonym, antonym, false friend, etc) *serialised in* the type for the parent <xr> element

- <bibl> for bibliographic information and <lbl> for textual triggers of a reference (e.g. "See", "voir" or "V.", etc), both remain the same in the two serialisations.
- **Sense & Sub-Sense:** These two models orchestrate the decomposition of the hierarchy of senses, if sense nesting occurs. For the Sense model, the supported labels are:
 - <subSense> for an embedded sense, *serialised in* <sense>
 - <gramGroup> for existing grammatical information, *serialised in* <gramGrp>
 - <num> for sense numbering, and note for any prose description related to the upper sense, <def> containing a definition of a sub-sense, <usg> for usage information, <re> for possible embedded related entries, <etym> for diachronic information related to the sense, and <xr> for recognised cross references. These elements have the same labels in the internal and final serialisations.
 - <example> for sense illustration, *serialised in a* <cit type="example"><quote> construct.
 - <translation> for translation equivalents, *serialised in* <cit type="translation"><quote> construct.
- **Etym/Quote & Etymology:** the labels of these models remain identical to what has been already described in Section 6.4.1 for the internal and final serialisation.

Additionally, we use the elements <dictScrap> and <pc> for all models to mark, respectively, junk text resulting from digital conversion (e.g. metadata, conversion errors, etc) or digitisation (i.e. OCR noise), and punctuation. We make use of another label <note>³¹ for related prose descriptions and notes, replacing the use of <dictScrap> in TEI-Lex-0 and LMF, given the fact that their guidelines do not recommend any element to use to encore noisy text in digitised dictionaries. All these elements are serialised using the same labels in the internal and the final serialisations.

Final Serialisation & Discussion

The model of the final TEI serialisation can be partially represented as a UML class diagram, to have an overview of the main serialisation elements that can be compared to the TEI-Lex-0 (Figure 6.6) and LMF (Figure 6.9) schemes.

Figure 6.16 represents the final serialisation model of the <entry> structure, its components and their mutual relationships.

On the one hand, we can notice that GROBID-Dictionaries's entry model contains all the TEI elements used for the serialisations of TEI-Lex-0 and the

³¹<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-note.html>

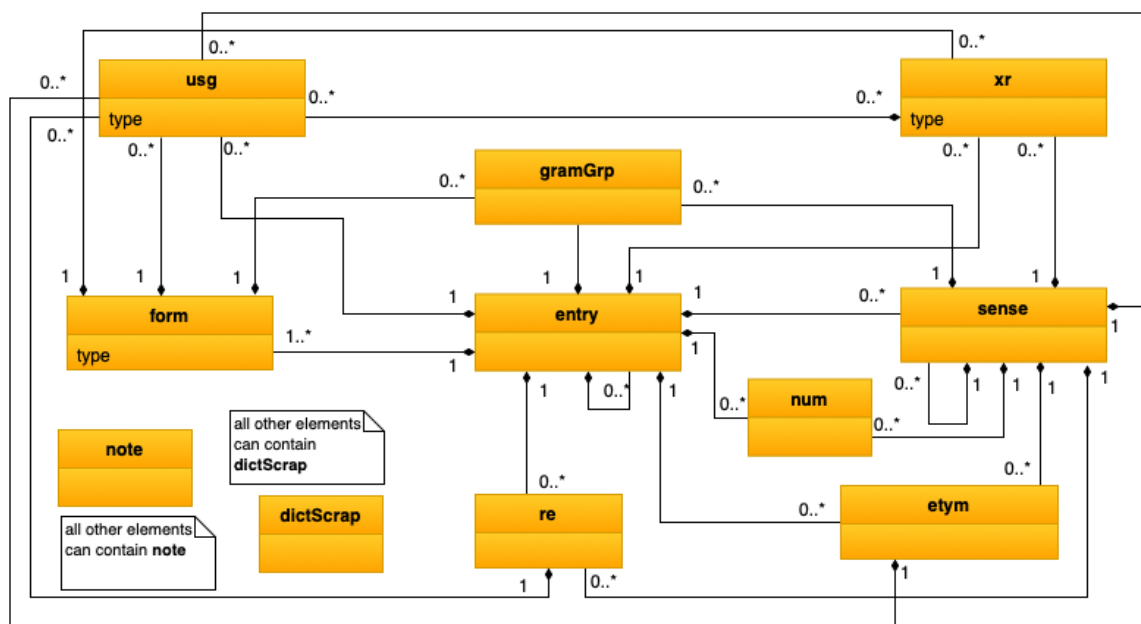


FIGURE 6.16: Final Serialisation of the <entry> Model in GROBID-Dictionaries

new LMF - except <lbl>. The elements for the serialisation of the morphological, grammatical and semantic structures - partially represented in Figure 6.16 - are also included in the GROBID-Dictionaries's serialisation. Thus, using the parsing models of our system enables, in a great number of modelling use cases, the generation of macro TEI serialisation that is compliant to the two new standardised schemes.

On the other hand, GROBID-Dictionaries's final serialisation model represents more constrained relationships than TEI-Lex-0 and gives more encoding alternatives than LMF. The affected relationships, like the mutual inclusions or the recursivity of certain elements in TEI-Lex-0 (i.e. <from> and <etym>) or the cross-reference modelling in LMF (i.e. <xr> being n-ary relationship), are not driven from print dictionaries modelling but rather the needs of NLP databases. These relationships can be added to the output of our system by applying a post-processing, such as enabling the mapping between recognised entries and their cross-references triggered in the source or the target by an <xr> element.

Our modelling anticipates some serialisation choices and provides dedicated serialisation, such as the different use of <note> and <dictScrap> which is driven from practical use cases encountered during the annotation of diverse print dictionaries. The need to encode the entire text of a parsing level forced us to find suitable encoding elements in the TEI guidelines, such as <dictScrap> for OCR noise or <num> for the numbering of entries or senses in some dictionaries.

Finally, the composition associations in this diagram, as in Figure 6.6 and as opposed in Figure 6.9, mean that none of the child elements can exist in the final output without its parent. In fact, this precision reflects the origin of the serialisation, which is not the result of the conversion of a relational

database, but rather an output generated from a cascading approach where a parent triggers and accommodates its child components.

6.4.3 Chapter Summary

This chapter presented two novel standardised schemes for lexical resources, TEI-Lex-0 and LMF, and their inspiration for the serialisation of the parsing models in GROBID-Dictionaries.

TEI-Lex-0, having the goal of defining a baseline encoding for print dictionaries, has managed to reduce TEI modelling alternatives towards more unified encoding guidelines. LMF has undergone an important revision work that has brought more structuring and richer information to the meta-model and its serialisation. The TEI serialisation for LMF managed to propose initial mappings between the two standards. More work is needed to investigate the applicability of the primary serialisation model on complex modelling scenarios and granular lexical information. We have shown the different modelling challenges, as well as the resulting serialisation for both new schemes.

We have also presented the two serialisation facets of GROBID-Dictionaries' models and the overlapping of its final TEI serialisation with TEI-Lex-0 and the new LMF. The compatibility with these two new standardised schemes has been kept maximal, despite the differences between the standardisation approaches. The degree of compliance of our models depends on the evolution of these two novel formats, as they are not yet finalised.

In Chapter 7 we present parsing experiments on different print resources using the described models of GROBID-Dictionaries. We give an exhaustive analysis of the strengths and weaknesses of the models employed and, more importantly, we focus on the possibility of scaling up the parsing while using the presented standardised scheme.

Chapter 7

GROBID-Dictionaries in Action

7.1 Introduction

After having presented the parsing architecture and the corresponding serialisation scheme, we now aim at assessing the performance of each lexical model against different print dictionaries. To this end, we need appropriate annotated data in order to conduct the various machine learning experiments.

In fact, the parsing scenarios we want to experiment require real world data that are able to exhibit different categories of digitisation anomalies (see Section 3.2.1). Consequently, the annotation of the lexical structures should take into consideration the specificities of our new serialisation model, along with these anomalies in raw documents. Besides, given the fact that the existing TEI lexical resources do not cover both specificities, we have carried out the work from scratch and annotated a pool of dictionaries that covered the diversity we needed for our experiments.

This final round of large-scale experiments has allowed us to use, in a uniform manner, the most stable version of the annotation scheme to a representative collection of dictionaries and thus annotate uniformly substantial amounts of data for several cascading models. Through the annotation and the training of the lexical models, we encountered additional challenges related to the quality and the balance of the data in terms of observable features and complexity. Besides the design and the feature selection for the parsing models, such data-related aspects appeared to significantly impact the machine learning experiments.

In this chapter, we present our machine learning setup by explaining our experimental goals and major factors interfering with the introduced sequence labelling task. Then we give an overview of the dictionaries pool and main challenges encountered while applying the annotation scheme to each dictionary. Next, we detail the series of experiments we carried out with various print dictionaries, using a selected set of cascading models of GROBID-Dictionaries' architecture, and show the extent to which our technique and the performed feature engineering can support their parsing. Finally we explore, through another series of experiments, the possibility of scaling up our system either by applying the system on similar structures in print documents or/and by integrating new generation models.

7.2 Machine Learning Experiment Setup

Before presenting the details and the evaluation of our experiments, we define the goals of our investigation and outline the factors interfering with the experimental setup to prepare a solid ground for the analysis of the results.

7.2.1 Experimental goals

Our experiments focus on studying the variation in performance observed for each model in the cascading chain, along with their learning behaviour when exposed to diverse dictionary content. The performance of each lexical model is studied with regard to:

- **Feature engineering:** our aim is to investigate the impact of basic and advanced tweaking of certain features that can improve the performance of essential sequence labelling models in our architecture. For that, we used the combinations of feature templates introduced in Section 4.3.2. A first set of experiments presents the difference in performances resulting from the use of the *Unigram* and *Bigram* templates. Another set of experiments is carried out with the *Engineered templates* to observe the impact of advanced feature engineering.
- **Generalising capacities:** given the best overall performance of a feature combination, we want to explore the ability of each model to generalise over the data that were used for its training. Two main variations are to be investigated: testing the models with data from already seen dictionaries and from unseen samples.
- **Learning curve:** for experiments that have the best performances, we want to study the learning behaviour of the models involved. More precisely, we want to gain insight into the evolution of each model's learning given a number of annotated pages used for its training. Such an aspect is studied with regard to the category and the quality of each dictionary.

To this end, our experiments focus on a selection of models from different levels of lexical parsing. We have chosen seven models of the cascading architecture. Besides the first three models necessary to activate the rest of the architecture explained in Figure 4.6, we selected **Form**, **GramGrp**, **Sense** and **Sub-sense** models. The choice of these models was based on the stability of their TEI encoding, as well as the fact that they are the most commonly used models by the early users of the system.

7.2.2 Interfering Factors

In the early stages of this work, we assumed that the task we are addressing starts with a dictionary file to parse and ends with the corresponding standardised structured output. Although such an assumption sound plausible and even relatively straightforward after gaining hands-on experience, we

discovered that the task is more complex than it might at first appear and starts much earlier. Drawing up an exhaustive list of the obstacles and issues involved is beyond the scope of this thesis. Nevertheless, we shed light on different aspects of sample representativity and the importance of digitisation details explored in different categories of print material. This will help later on in the evaluation of the various experiments that we carried out with different dictionaries.

Sample Selection and Annotation

To train a supervised machine learning model, the annotation of a representative sample is required. In the case of GROBID-Dictionaries, a representative sample can vary from one model to another. For instance, a representative sample for the **Dictionary Body Segmentation** model, which has the task of recognising the boundaries of each dictionary article in a page body, should contain dictionary pages having enough:

- entries of different sizes,
- and different spots of the entries in a page: beginning of page, beginning of a column, end of a page, split over several pages or columns, etc.

On the other hand, a representative sample for the **Lexical Entry** model, which is responsible for parsing the main components of a lexical entry, should contain, for instance:

- entries having a simple structure
- entries having a complex structure. Such entries should comprise:
 - items representing different physical structures (typographic, layout, etc.)
 - items representing different logical structures, especially less frequent components: for instance entries with and without related entries or cross-references that could be less present in dictionary articles than forms or senses
- entries with more textual markup variation, especially when typographic information is poor
- in some cases, entries with and without hyphenation where the latter is frequent and occurring at the beginning or the end of the text sequences, or when the dictionary articles are very short (i.e. few tokens per entry)

The representativity of samples varies from one dictionary to another. And given the differences in the representativity needed to train each model

of our machine learning architecture, selecting pages that contain all the variations required for the convergence of all models is extremely challenging. Selecting different pages for training different models can lead to the propagation of the recognition errors by higher level models in the training data of subsequent models. This would result in an imprecise evaluation as the model being tested may not itself be responsible for any potential recognition errors. In addition, the conventional dataset split ratios ¹, for training, development and testing, can not always be respected.

Model	Training	Evaluation
<i>Dictionary Body Segmentation</i>	572 <entry>	270 <entry>
<i>Lexical Entry</i>	572 <sense>	269 <sense>
	572 <lemma>	270 <lemma>
	28 <inflected>	10 <inflected>
	10 <re>	4 <re>
<i>Sense</i>	856 <subSense>	302 <subSense>
<i>Form</i>	756 <orth>	269 <orth>
	31 <part>	11 <part>
	31 <gramGrp>	11 <gramGrp>
<i>Sub-Sense</i>	905 <def>	319 <def>
	32 <usg>	11 <usg>
	7 <gramGrp>	8 <gramGrp>
	9 <translation>	2 <translation>

TABLE 7.1: Page Sampling Statistics (Bowers, Khemakhem, and Romary, 2019)

Table 7.1 gives an overview of the statistics of a page sampling process for an experiment (Bowers, Khemakhem, and Romary, 2019) with a Mixtec-Spanish dictionary represented by the sample in Figure 7.1.

For training 5 models of the architecture, we selected and annotated 14 pages from different spots in the dictionary: 10 for training and validation, and 4 for testing. We detail the annotated instances for each model, except for the first one dealing with the prediction of main regions of a page. The page sampling shows different ratios for each label of each model. For example, for the *Sub-Sense* model, the ratios for the instances of <def> labels in the training and evaluation datasets are very close to the 2/3, 1/3 ratios, whereas the <gramGrp> testing proportion exceeds the training one. The number of <gramGrp> instances is also small compared to <def>. Annotating more separate instances of <gramGrp> and adding them to the training and testing datasets would be misleading for the model as the context of the sequence must be preserved for the model to learn the right features. The sample in

¹There are two main conventions for dataset splitting ratios. First, 2/3 for training and validation and 1/3 for testing. Second, 80% for training and validation and 20% for testing. The choice of these conventional ratios depends on the amount of data available and the preferences of designer of the machine learning experiment, as the first convention is more challenging for a trained model.

Figure 7.1 shows several hyphenated tokens in small size dictionary articles. The hyphenation occurs in different spots in the entries. Thus, including such a variation in the training dataset is crucial for more than one parsing model.

OCR Impact

The second major factor that impacts on the machine learning experiments that we want to conduct is related to the OCR quality, particularly in the case of digitised material.

To investigate such an impact, we carried out an experiment (Khemakhem et al., 2019) with two versions of a retro-digitised legacy dictionary (Furetière, 1701). We use the term OCRisation to designate the process of using an OCR system, or any character recognition system, to recognise the layout and the text of a document. For this experiment, we carried out the OCRisation by using the Transkribus platform (Kahle et al., 2017) and following the workflow described by Lindemann, Khemakhem, and Romary, 2018. The process consists of using a default OCR model to produce a first layer of OCRs that will be manually corrected and then used to train a Handwritten Text Recognition (HTR) model to produce higher quality digitised text.

The first sample was compiled from a low image quality document that was OCRised with an HTR model trained with 28 pages. The second sample was created using an HTR model trained with 108 pages applied on a high image quality document. We annotated the same 45 pages from each sample of the dictionary. We tried to select the pages that covered the maximum number of variations required for training the first three models of GROBID-Dictionaries. For each parsing level, two instances of the CRF models with the same features were trained and evaluated.

<i>Tag</i>	Sample 1			Sample 2		
	Precision	Recall	F1	Precision	Recall	F1
<body>	75	70	72.41	81.48	73.33	77.19
<footnote>	91.67	73.33	81.48	84.62	91.67	88
<headnote>	88.46	82.14	85.19	100	90	94.74

TABLE 7.2: Field Level Evaluation of the Dictionary Segmentation Model

<i>Tag</i>	Sample 1			Sample 2		
	Precision	Recall	F1	Precision	Recall	F1
<dictScrap>	81.82	85.71	83.72	100	90	94.74
<entry>	85.85	80.53	83.11	89.47	91.07	90.27
<pc>	92.59	96.15	94.34	93.75	97.56	95.62

TABLE 7.3: Field Level Evaluation of the Dictionary Body Segmentation Model

dzavua dzuhua ama ama: así, así (de enojo)	dzavua tnaha huitna: a esta hora
dzavua hooça tucu: sobre todo (precediendo plática)	dzavua tnaha iyo: estado de cada cosa
dzavua huitna: a esta hora	dzavua tnaha iyo sa yonaha sa yonduvui letra: forma de letra
dzavua ino huidzo sindi: obligado ser por profesión el fraile	dzavua tnaha quevui pascua yoquidzandi, yocuvui inindi, yotuvui sindi: hacer cuenta que es pascua
dzavua itnaha iyo: manera o modo	dzavua tnaha sa dzavuandi: correr muy ligeramente
dzavua iyo: estado de cada cosa; forma o manera; inclinación; manera o modo	dzavua tnaha sinindo: de cualquiera manera que lo hagas
dzavua iyo: soler (acostumbrar)	dzavua tnaha tecoo yee yavui: blanca cosa en gran manera
dzavua iyo inindi: certificado estar, por cierto tener	dzavua tnaha tiyee yavui: blanca cosa en gran manera
dzavua iyo ino sindi: obligado ser por profesión el fraile	dzavua tnaha yocachi S.fee [Santa Fe]: acompañar con buenas obras la fe
dzavua iyo ndatu sindi: obligado ser por profesión el fraile	dzavua tnaha yocuvui misa: en tanto, o en tanto que dicen misa
dzavua iyo ndudzu itnayta: comúnmente se dice	dzavua tnaha yuvui cachi: blanca cosa en gran manera
dzavua ndatundi: esta es mi suerte	dzavua tucu: también
dzavua nicaa inindi: aficionado o aplicado naturalmente ser una cosa más que otra	dzavua tucuca: sobre todo (precediendo plática)
dzavua nicacu: condición natural de alguno; inclinación	dzavua yocachita dzavua yosinita: así se dice o se entiende
dzavua nicaye: medio quemado	dzavua yocuvui dzavuayu sa cuvui dzavuayu iyo: acostumbrarse por usarse
dzavua nicoo coo inindi: aficionado o aplicado naturalmente ser una cosa más que otra	dzavua yocuvui nuu yaha: usarse entre algunos o entre todos
dzavua nisa coho naha ñuhu sindi: obligado ser por profesión el fraile	dzavua yocuvui ñuu sa naha: costumbre de pueblo
dzavua nisa coho quevui sindi: obligado ser por profesión el fraile	dzavua yonahata: así se dice o se entiende
dzavua nisa tnaha inindi: aficionado o aplicado naturalmente ser una cosa más que otra	dzavua yonana: semejar o parecerse una cosa a otra
dzavua nisini nitacundi: obligado ser por profesión el fraile	dzavua yoqni inindi: aficionado o aplicado naturalmente ser una cosa más que otra
dzavua ñuhu indaandi: esta es mi suerte	dzavua yosahandita: doblar el precio
dzavua ñuu: a media noche	dzavua yosico inindi: aficionado o aplicado naturalmente ser una cosa más que otra
dzavua quaha inindi: aficionado o aplicado naturalmente ser una cosa más que otra	dzavua yosii tnahata dzo ndehe dzavua yocahata: comúnmente se dice
dzavua quaha ino ndatundi: obligado ser por profesión el fraile	dzavua yotacusita: así se dice o se entiende
dzavua quaha ndudzu: así se usa	dzavua yotusi inindi: aficionado o aplicado naturalmente ser una cosa más que otra
dzavua quidzata sa naha: costumbre de pueblo	dzavua yotuvui: semejar o parecerse una cosa a otra
dzavua saha tniño: acerca de aquel negocio	dzavua yotuvui sindi: parecerme o así me parece
dzavua tachi dzavua tnumi yosino saandi: correr muy ligeramente	dzavua yu: certificado estar, por cierto tener
dzavua tayee dzavua ñaha dzehe: así hombres como mujeres	dzavua yu: usarse entre algunos o entre todos
dzavua tna iyoni: como quiera	
dzavua tnaha: así como; como comparativo; según (<i>adverbio</i>)	

FIGURE 7.1: Excerpt from a Mixtec-Spanish Dictionary (Alvarado, 1593)

Tag	Sample 1			Sample 2		
	Precision	Recall	F1	Precision	Recall	F1
<etym>	87.5	60	71.19	73.68	71.79	72.73
<form>	94.44	92.73	93.58	92.24	96.4	94.27
<pc>	90.91	69.44	78.74	88.97	80.13	84.32
<re>	33.33	9.09	14.29	55.56	22.73	32.26
<sense>	67.65	59.28	63.19	77	76.65	76.84
<xr>	100	80	88.89	100	100	100

TABLE 7.4: Field Level Evaluation of the Lexical Entry Model

The results² of the experiment, represented in the evaluation of the CRF models in Table 7.2, 7.3 and 7.4, show the different measures on the field level. We can notice the differences in the performance of each model to recognise the different labels.

Three main conclusions can be drawn from this experiment: firstly the OCR quality plays an important role in boosting or deteriorating the performance of our machine learning models. Secondly, despite the noisy OCRs, the CRF models can still operate and produce for the recognition of some labels comparable results to high quality OCRised material. Finally, the impact of the OCRs is more visible for the extraction of fine grained information which can be explained by the importance of the textual and markup at such levels.

7.2.3 Dictionary Samples & Annotation

Selecting dictionaries for our experiments was a challenge in itself as several anomalies were often only discovered after many rounds of cascading annotation, especially for OCRised samples. Our final dictionaries pool is composed of 5 dictionaries, each reflecting several aspects of the classification established at the beginning of this thesis (see Figure 2.4): monolingual or bilingual, born-digital or digitised, modern or legacy, and with lexical or/and encyclopaedic content.

The multi-stage annotation of the different dictionaries was carried out manually and semi-automatically by using the two modes introduced in Section 4.4.2 (*raw* and *pre-annotated* training data). One person was in charge of the whole annotation/correction task for all the dictionaries. The raw and annotated data, along with the results of the experiments have been released on a freely available repository³.

In this section we present each dictionary with regard to these criteria as well as the information about its structural complexity. We also highlight main consequential specificities encountered during the annotation of a number of lexical structures.

²In this chapter, the labels presented in the evaluation of the models are of the internal serialisation of the architecture and not the final one

³https://github.com/MedKhem/grobid-dictionaries_data

Dictionnaire de la Langue Française (DLF)

Description: "Dictionnaire de la Langue Française" (Littré, 1873) is a legacy monolingual French dictionary that contains both lexical and encyclopaedic articles. Besides the classic explanation of entries, exceeding 78 000 articles, the lexicographer provides an exhaustive historical and etymological description of lexemes with nearly 300 000 citations. The retro-digitised version⁴ that we used for the purpose of our experiments comprises 4 volumes with over 5 000 article pages each organised in 3 columns (see Figure 7.2). We do not have enough information about the digitisation process but we assess the OCRs quality in the PDF we used to be relatively good.

Dictionary & Annotation Specificities: We consider this dictionary to be the most complex sample we used for our study, as it represents challenges at almost all parsing levels. Lexical markups, namely *parenthesis*, *bold* and *italic*, are omnipresent, especially for distinguishing the structures at the **Lexical Entry** level. Such markers are much less present at the **Sense** and **Sub-Sense** levels where *pipes* and *numbering* are visually the most helpful clues. We can group the main challenges we faced during the annotation of DLF by model:

- **Lexical Entry:** The dictionary articles have a high variation in length, from few tokens up to several columns. *Inflected forms* such as "Etudiant, E" were not marked as inflected forms at the **Lexical Entry** level, as it will break the annotation of grammatical information related to the main form. We also made the same decision for *variants* that occur after lemmas with no geographic or other more elaborate information. Consequently, we postponed the annotation of these structures to the **Form** level, where we annotate them as <part>s.
- **Sub-Sense:** DLF has a very rich semantic representation and it is often a challenge to human experts to differentiate the boundaries of definitions and examples. In fact, most of the examples are citations from classic French literature although some brief usage examples may occur just after a definition. To overcome this issue, we based our approach on the choices made to compile a structured version⁵ of the dictionary and we considered examples to be only citations.

Easier English Basic Dictionary (EEBD)

Description: "Easier English Basic Dictionary" (Publishing, 2009) is a monolingual dictionary for English which contains over 5,000 entries, published in 2009. For our experiments, we used the 370 pages containing the body of the dictionary. The PDF version which we used, is a digitally born one that has two columns per page.

⁴<https://gallica.bnf.fr/ark:/12148/bpt6k5406710m>

⁵<https://www.littre.org>

CAM

† CALYBITE (ka-li-bi-t'), s. m. Nom de solitaires chrétiens qui habitaient dans des huttes.
— ETYM. Καλυβίτης, de καλύβη, hutte.

CALYCANDRIE (ka-li-kan-drie), s. f. Terme de botanique. Classe de plantes dont les étamines sont insérées au calice.
— ETYM. Κάλυξ, calice, et ἀνθή, mâle.

† CALYCANTHÈME (ka-li-kan-tè-m'), adj. Terme de botanique. Dont le calice a l'apparence d'une corolle.
— ETYM. Κάλυξ, calice, et ἀνθήμα, fleur.

† CALYPTÈRES (ka-li-ptè-r'), s. f. plur. Terme de zoologie. Petites plumes qui couvrent le bas de la queue des oiseaux.
— ETYM. Καλυπτήριον, ce qui sert à cacher, de καλύπτειν, cacher.

† CALYPTRE (ka-li-ptr'), s. f. Terme de botanique. Coiffe des mousses.
— ETYM. Καλυπτρα, de καλύπτειν, cacher.

† CALYPTRE, ÈE (ka-li-ptrè, ptrée), adj. Terme de botanique. Qui est croisé d'une coiffe.
— ETYM. Calyptra.

CAMAÏEU (ka-ma-ieu), s. m. || 4° Pierre fine taillée, ayant deux couches de différentes couleurs, dont l'une est devenue la figure en relief, et l'autre fait le fonds. || 2° Genre de peinture où l'on n'emploie qu'une couleur avec des teintes plus sombres et plus claires. Peindre en camaïeu. || Un camaïeu, un tableau peint en camaïeu. || Par dénigrement, un camaïeu, un tableau d'une couleur lourde et monotone. Le coadjuteur a bien ri des camaïeux de peinture, que vous comparez à l'histoire de France en madrigaux, sév. 492. || 3° Gravure qui est une imitation de la manière en lavis.
— HIST. xiv^e s. Beles chambres qui seront d'or et d'argent et de pierres précieuses, c'est à savoir rubis, esmeraudes, saphyrs, cameuz et marguerites, *Livres de la loi au Sarasin*, p. 433. Une autre boiste d'argent où est le tricière au duc, et un camaïeu, *Bibl. des chartes*, 4^e série, t. v, p. 469. Un tableau d'or ouquel il a un grans gamahieu assis sur bois, DE LABOURDE, *Émaux*, p. 488. Un lorain [courtoise garnie de soie] semé de boutons dorés et de camahieux, id. 4b. Le camahieu qui autrement est appelé oncle, id. 4b. Un camahieu, dont le champ est vermeil et a deux figures dessus à une beste assise en une verge toute plaine, id. 4b. || xv^e s. Ung fuzil entaillé en un camaïeu où estoit ses armes, COMM. v, 9. || xvii^e s. Cet anneau avoit pour sa pierre un capidon couronné fort magnonnement, étant entaillé en un camaïeu d'amantite, YVER, 688. Chamahieux, PALSON, p. 402.
— ETYM. Espagn. *camafœu*; bas-lat. *camahotus*, *camahutus*; du bas-lat. *camoules*, sardoine, onyx (voy. CAMÉE).

CAMAIL (ka-mail, il moullées), s. m. || 1° Habille ment du clergé en hiver, couvrant la tête, les épaules, et allant jusqu'à la ceinture. || 2° Petit manteau tombant des épaules à la ceinture, que portent par-dessus le rochet les évêques et autres ecclésiastiques privilégiés. Les évêques étaient en rochet et camail, boss. *Lett. Qué.*, 482. || 3° Terme de blason. Espèce de lambrequin servant à couvrir le casque et l'écu des chevaliers. || Au plur. Des camails.
— HIST. xiv^e s. Et voit ses chevaliers bien armés de camail, *Guescl.* dans RAYNOUARD, *Lesque*. Bertrand tenoit l'espée qui le fer eut tranchant, Au camail lui bouta fèrement en poussant, DU GANGE, *Camelacum*. || xv^e s. Tant s'avance le sire de Langurant [au siège de Duras] que de sa vie il se mit en grand aventure; car ceux de dedans par force lui arracherent le bassin de la teste atout le camail, FROISS. II, II, 44. Ung camail d'argent de l'ordre de monseigneur d'Orléans, pesant sept onces trois gros, DE LABOURDE, *Émaux*, p. 492. Un camail en façon de trelliz, et est leat camail cintré par-dessus de bossètes tant d'or que esmailées de blanc et de rouge cler, m. ib.
— ETYM. Provenç. *capmail*, *capmail*, *capmail*, *capmail*; ital. *camaglio*; de *cap*, tête (voy. CHEF), et *mail*, armure (voy. MAILLE); proprement une armure de tête, puis un vêtement de tête.

CAMALDULÈ (ka-mal-du-lè), s. m. || 4° Religieux d'un ordre monastique fondé, à la fin du x^e siècle, par saint Romuald; l'habit est blanc; la règle est celle de saint Bernard. Ragotzi s'était retiré aux camaldules de Grosbois, sr-sim. 469, 203. || Il y avait aussi des religieux camaldules. || 2° S. f. Une camaldule, un couvent de camaldules.
— ETYM. Camaldoli, localité de la Toscane où l'ordre fut d'abord établi.

† CAMANIOC (ka-ma-ni-ok), s. m. Espèce de manioc qu'on cultive à Cayenne et dans les Antilles, dont on peut manger la racine cuite, sans prépa-

ration préalable, comme les pommes de terre, tandis que les racines de manioc contiennent un suc vénéneux, qu'il faut d'abord extraire.

CAMARADE (ka-ma-ra-d'), s. m. || 4° Nom que se donnent entre eux les militaires. Des camarades de régiment. En avant! partons, camarades, L'arme au bras, le fusil chargé, BÉRANG. *Vieux cap.* || 2° Par extension, substantif des deux genres, celui, celle qui a même vie, mêmes habitudes, mêmes occupations que plusieurs autres personnes. Camarades d'école, de collège, de chambre. Des camarades d'enfance, des camarades de bureau. C'est une mauvaise camarade. La taille du maréchal duc de Noailles est assez grande, mais épaisse; sa démarche lourde et forte; son vêtement, uni ou tout au plus d'officier, voudrait montrer la simplicité la plus naturelle; il la soutient avec le gros de ce que, faute de meilleure expression, on entend par apparence de sans façon et de camarade, sr-sim. 317, 438. Eh, mon Dieu! s'écria-t-il, je crois que c'est là Jeannot; le petit homme rebondi ne fait qu'un saut et court embrasser son ancien camarade, VOLT. *Jeannot et Colin*. || Camarade de lit, celui qui couche dans le même lit qu'un autre. Deux soldats qui couchaient dans le même lit étaient camarades de lit. || Fig. Que le bon soit toujours camarade du beau, Dès demain je chercherai femme, LA FONT. *Fab.* VII, 2. || 3° Populairement, ami. Ils se sont remis camarades. Mon camarade, Tiens, bois rasade, BÉRANGER, *Troub.* || 4° Se dit de ceux qui courent même fortune. Nous avons été camarades d'aventures, d'infortune. Dans ce désappointement il eut bien des camarades. || 5° Familièrement, en s'adressant à des inférieurs, même inconnus. Mon camarade, enseignez-moi, je vous prie, le chemin de...
— SYN. CAMARADE, COMPAGNON. Camarade est d'origine un terme militaire, et signifie de la même chambre; de là, figurément, il exprime celui qui a avec d'autres même genre d'occupations ou d'habitudes. Compagnon, qui veut dire d'origine celui qui mange le même pain, n'a point cette particularité de sens; il n'implique pas qu'on soit de même occupation; il implique qu'on accompagne. Ainsi on dit: des camarades de lit, des compagnons de voyage. Vivre d'un même genre de vie pour camarades, s'accompagner pour compagnons, voilà la nuance de sens essentielle entre ces deux mots. Nous disons camarades de collège et non compagnons de collège; mais au féminin compagnes de pension, de couvent; cette déviation tient à ce que l'oreille a désiré marquer le féminin que la désinence ne signale pas dans camarade.
— HIST. xvii^e s. Ordinairement un capitaine [d'infanterie espagnole] en aura cinq ou six [soldats choisis] qu'il appelle ses camarades, LANOUE, 296. M. de Langy, au lieu qu'il a écrit de la discipline militaire, parle des camarades, qu'il appelle en nostre langue française chambre, et les fait de dix soldats, baillant à l'un d'eux quelque proeminence sur les autres, et le nomme chef de chambre, id. 294. Comba fut pris en la maison d'une vieille qui blanchissoit le linge de sa camarade, qu'il nommoit ainsi à l'espagnol, CARLOIX, VI, 46. Comme estant d'une camarade, et participants à toutes ses entreprises, id. X, 44.
— ETYM. Espagn. *camarada*, s. m.; ital. *camerata*, s. m. de l'espagnol *camara*, ital. *camera*, chambre (voy. CHAMBRE); proprement chambre, puis, au masculin, celui qui demeure dans la même chambre, camarade. Dans les exemples cités à l'historique, *camarade* signifie chambre, et, par extension, homme de chambre. *Camarade* est d'origine un terme militaire.

CAMARADERIE (ka-ma-ra-de-rie), s. f. || 4° La familiarité qui existe entre camarades. Cette camaraderie de vous et de Mlle Duplessis, sév. 70. La plupart des liaisons de société, la camaraderie... tout cela est à l'amitié ce que le sigisbéisme est à l'amour, CHAMPFLOIT, dans le *Dict. de dochez*. || 2° Disposition d'esprit qui fait que des écrivains, des artistes qui ont des liaisons entre eux se soutiennent et se prouvent mutuellement. Son succès n'est pas de bon aloi, il est dû à la camaraderie. Le premier emploi de camaraderie en ce sens est attribué à H. Delatouche, par Chastes, *J. des Débats*, 15 juillet 1860.
— ETYM. *Camarade*.

CAMARD, ARDE (ka-mar, mar-d'; le d ne se lie pas: ka-mar et bossu; au pluriel, semblablement, kamars et bossus; d'autres prononcent ka-mar-z et bossus). || 4° S. m. et f. Qui a le nez plat et écrasé. Un camard. Une camarde. || 2° Adj. Un nez camard. C'é-

tait une grosse fille écrasée, brune, laide, camarde, avec de l'esprit, sr-sim. 24, 47. L'Égypte... Dans sa robe de sable enfonce enveloppés Ses colosses camards, à la face frappés Par le pied brutal de Cambyse... v. HUGO, *Voix*, 4. || Dans le style burlesque, la camarde, la mort. Il fut complimenté d'abord Par le Sonneil et par la Mort; Pour lui faire honneur, la camarde, Contre son humeur, fut gaillarde, SCARRON, *Énéide*, VI.
— HIST. xvii^e s. Mais d'où vient cet orgueil? on ne voit par la ville Un plus rogue vilain, qui contreface mieux Depuis un peu de temps le brave et glorieux Que ce petit camard... TABOURET DES ACCORDS, *Bigarr. Descriptions pathétiques*.
— ETYM. Même radical que *camus*.

† CAMARE (ka-ma-r'), s. m. Terme de botanique. Fruit aplati et membraneux composé de deux valves soudées.
— ETYM. Καμάρα, voûte.

† CAMARILLA (ka-ma-ri-la), s. f. Coterie de personnes qui approchent du prince le plus près.
— ETYM. Diminutif de *camara*, chambre en espagnol (voy. CHAMBRE).

† CAMARIN (ka-ma-rin), s. m. Espèce de plongeon.
— HIST. xviii^e s. Banquier qui se livre aux opérations de change. || Vieilli. On dit aujourd'hui agent de change.
— ETYM. Ital. *cambio*, change (voy. CHANGE).

† CAMBIUM (kan-bi-om'), s. m. || 4° Terme de botanique. Suc nutritif, élaboré, destiné à fournir les matériaux de l'accroissement des plantes. || 2° Terme de jardinage. Nom donné aux tissus en voie de formation et étant encore moles et gélatineux.
— HIST. xvii^e s. La troisième humeur de nourrissement s'appelle cambium, qui est à changé et agglutiné et peu s'en faut à tourné en nourrissement, PARÉ, *Introd.* 6.
— ETYM. Bas-lat. *cambium*, mot qui se trouve dans Arnaut de Villeneuve, xiv^e siècle (*cambium*, humiditas manifeste alterata membris continentis complexionis); de *cambire*, changer (voy. CHANGER).

CAMBOUIS (kan-bou-i), s. m. Vieux oing qui, employé pour adoucir les frotements d'une roue sur l'essieu, d'une machine, prend le nom de cambouis quand il a été noirci par le frotement et le mélange des parties métalliques. L'huile qu'on met aux roues des voitures devient aussi du cambouis par le frotement. || Terme de vétérinaire. Matière sébacée qui s'accumule souvent en quantité considérable à l'intérieur du fourreau du cheval.
— HIST. xvii^e s. Prenez cambouis, c'est le limon noir qui est aux deux bouts de l'essieu de la charrue, MÉNAGIER, II, 6. || xvii^e s. Ah très ordie vieille truaude! Vous me baillez du cambouis [vous me dupez], *Farce du meunier de qui le diable emporte l'âme en enfer*, Paris, 1834, p. 49.
— ETYM. Raynouard le tire du provençal *camois*, boue, souillure.

† CAMBOUISÉ, ÈE (kan-bou-i-zé, zée), adj. Crasseux, en parlant des pièces de la batterie d'un fusil.

† CAMBRAI (kan-brè), s. m. Sorte de toile de lin très-claire. || Aujourd'hui, dentelle faite à la mécanique et non aux fuseaux; l'imitation, la fausse dentelle.

— ETYM. *Cambrai*, ville où ce tissu se fabrique.

† CAMBRE (kan-br'), s. f. *Cambure*.

CAMBRE, ÈE (kan-brè, brée), part. passé. Une taille cambrée, taille qui présente une concavité en arrière. Jambes cambrées, celles dont la courbe naturelle est exagérée, de sorte que les genoux sont distants l'un de l'autre quand les talons se touchent. On dit dans le même sens qu'un homme est cambré.

† CAMBREMENT (kan-brè-man), s. m. Action de cambrer.

CAMBRER (kan-brè), v. a. || 4° Arquer légèrement. Cambrer une pièce de bois. || 2° Se cambrer, v. réfl. Devenir cambré. Cette poutre commence à se cambrer.
— HIST. xvii^e s. Des astelles cambrées, pour mieux se coucher autour de la jambe, PARÉ, XIII, 23. Par une violence les os des jeunes enfans se courbent et cambrent, id. XV, 4. Elle se cambre, en marchant, très-fort, PALSON, p. 461. Vous allez en cambrant comme se ce fust un qui eust les rayns rompus, id. p. 673.
— ETYM. *Camberare*, voûter, de *camera*, voûte.

† CAMBRÉSINE (kan-brè-zi-n'), s. f. Toile de lin claire et fine qui se fabriquait à Cambrai.
— ETYM. *Cambrai*, ville où se fabriquait ce tissu.

FIGURE 7.2: Excerpt from the Dictionnaire de la Langue Française Dictionary (Littré, 1873)

<p>b /bi:/, B <i>noun</i> the second letter of the alphabet, between A and C</p> <p>baby /'beɪbi/ <i>noun</i> 1. a very young child ○ <i>Most babies start to walk when they are about a year old.</i> ○ <i>I've known him since he was a baby.</i> 2. a very young animal ○ a <i>baby rabbit</i> (NOTE: The plural is babies. If you do not know if a baby is a boy or a girl, you can refer to it as it: <i>The baby was sucking its thumb.</i>)</p> <p>back /bæk/ <i>noun</i> 1. the part of the body which is behind you, between the neck and top of the legs ○ <i>She went to sleep lying on her back.</i> ○ <i>He carried his son on his back.</i> ○ <i>Don't lift that heavy box, you may hurt your back.</i> 2. the opposite part to the front of something ○ <i>He wrote his address on the back of the envelope.</i> ○ <i>She sat in the back of the bus and went to sleep.</i> ○ <i>The dining room is at the back of the house.</i> ■ <i>adjective</i> 1. on the opposite side to the front ○ <i>He knocked at the back door of the house.</i> ○ <i>The back tyre of my bicycle is flat.</i> 2. (of money) owed from an earlier date ○ <i>back pay</i> ■ <i>adverb</i> 1. towards the back of something ○ <i>She looked back and waved at me as she left.</i> 2. in the past ○ <i>back in the 1950s</i> 3. in the state that something was previously ○ <i>Put the telephone back on the table.</i> ○ <i>She watched him drive away and then went back into the house.</i> ○ <i>She gave me back the money she had borrowed.</i> ○ <i>I'll phone you when I am back in the office.</i> (NOTE: Back is often used after verbs: to give back, to go back, to pay back, etc.) ■ <i>verb</i> 1. to go backwards, or make something go backwards ○ <i>He backed or backed his car out of the garage.</i> 2. to encourage and support a person, organisation, opinion or activity, sometimes by giving money ○ <i>Her colleagues were willing to back the proposal.</i> ○ to put someone's back up to annoy someone</p> <p>back up <i>phrasal verb</i> 1. to help or support someone ○ <i>Nobody would back her up when she complained about the service.</i> ○ <i>Will you back me up in the vote?</i> 2. to make a car go backwards ○ <i>Can you back up, please – I want to get out of the parking space.</i></p> <p>background /'bækgraʊnd/ <i>noun</i> 1. the part of a picture or view which is behind all the other things that can be seen ○ <i>The photograph is of a house with mountains in the background.</i> ○ <i>His white shirt stands out against the dark background.</i> Compare foreground □ In the background while other more obvious or important things are happening 2. the experiences, including education and family life, which someone has had ○ <i>He comes from a working class background.</i> ○ <i>Her background is in the restaurant business.</i> 3. information about a situation ○ <i>What is the background to the complaint?</i></p> <p>backward /'bækwəd/ <i>adverb</i> US same as backwards</p> <p>backwards /'bækwɔːd/ <i>adverb</i> from the front towards the back ○ <i>Don't step backwards.</i> ○ <i>'Tab' is 'bat' spelt backwards.</i> □ backwards and forwards in one direction, then in the opposite direction ○ <i>The policeman was walking backwards and forwards in front of the bank.</i></p> <p>bacon /'beɪkən/ <i>noun</i> meat from a pig which has been treated with salt or smoke, usually cut into thin pieces</p> <p>bacteria /'bæktɪəriəl/ <i>plural noun</i> very small living things, some of which can cause disease (NOTE: The singular is bacterium.)</p> <p>bacterial /'bæktɪəriəl/ <i>adjective</i> caused by bacteria ○ a <i>bacterial infection</i></p> <p>bad /bæd/ <i>adjective</i> 1. causing problems, or likely to cause problems ○ <i>Eating too much fat is bad for your health.</i> ○ <i>We</i></p>	<p>badge /bædʒ/ <i>noun</i> a small sign attached to someone's clothes to show something such as who someone is or what company they belong to</p> <p>badly /'bædli/ <i>adverb</i> 1. not well or successfully ○ <i>She did badly in her driving test.</i> 2. seriously ○ <i>He was badly injured in the motorway accident.</i> 3. very much ○ <i>His hair badly needs cutting.</i> (NOTE: badly – worse /wɔːs/ – worst /wɔːst/)</p> <p>bag /bæg/ <i>noun</i> 1. a soft container made of plastic, cloth or paper and used for carrying things ○ a <i>bag of sweets</i> ○ <i>He put the apples in a paper bag.</i> 2. same as handbag ○ <i>My keys are in my bag.</i> 3. a suitcase or other container used for clothes and other possessions when travelling ○ <i>Have you packed your bags yet?</i></p> <p>baggage /'bæɡɪdʒ/ <i>noun</i> cases and bags which you take with you when travelling</p> <p>bake /beɪk/ <i>verb</i> to cook food such as bread or cakes in an oven ○ <i>Mum's baking a cake for my birthday.</i> ○ <i>Bake the pizza for 35 minutes.</i></p> <p>baker /'beɪkə/ <i>noun</i> a person whose job is to make bread and cakes □ the baker's a shop that sells bread and cakes ○ <i>Can you go to the baker's and get a loaf of brown bread?</i></p> <p>balance /'bæləns/ <i>noun</i> 1. the quality of staying steady ○ <i>The cat needs a good sense of balance to walk along the top of a fence.</i> □ to keep your balance not to fall over □ to lose your balance to fall down ○ <i>As he was crossing the river on the tightrope he lost his balance and fell.</i> 2. an amount of money remaining in an account ○ <i>I have a balance of £25 in my bank account.</i> 3. an amount of money still to be paid from a larger sum owed ○ <i>You can pay £100 now and the</i></p> <p>balance in three instalments. ○ <i>The balance outstanding is now £5000.</i> ■ verb 1. to stay or stand in position without falling ○ <i>The cat balanced on the top of the fence.</i> 2. to make something stay in position without falling ○ <i>The waiter balanced a pile of dirty plates on his arm.</i></p> <p>balcony /'bælkəni/ <i>noun</i> 1. a small flat area that sticks out from an upper level of a building protected by a low wall or by posts ○ <i>The flat has a balcony overlooking the harbour.</i> ○ <i>Breakfast is served on the balcony.</i> 2. the upper rows of seats in a theatre or cinema ○ <i>We booked seats at the front of the balcony.</i> (NOTE: The plural is balconies.)</p> <p>bald /bɔːld/ <i>adjective</i> having no hair where there used to be hair, especially on the head ○ <i>His grandfather is quite bald.</i> ○ <i>He is beginning to go bald.</i></p> <p>ball /bɔːl/ <i>noun</i> 1. a round object used in playing games, for throwing, kicking or hitting ○ <i>They played in the garden with an old tennis ball.</i> ○ <i>He kicked the ball into the goal.</i> 2. any round object ○ a <i>ball of wool</i> ○ <i>He crumpled the paper up into a ball.</i> 3. a formal dance ○ <i>We've got tickets for the summer ball.</i> ○ to start the ball rolling to start something happening ○ <i>I'll start the ball rolling by introducing the visitors, then you can introduce yourselves.</i> ○ to play ball to work well with someone to achieve something ○ <i>I asked them for a little more time but they won't play ball.</i> ○ to have a ball to enjoy yourself a lot ○ <i>You can see from the photos we were having a ball.</i></p> <p>ballet /'bæleɪ/ <i>noun</i> 1. a type of dance, given as a public entertainment, where dancers perform a story to music 2. a performance of this type of dance ○ <i>We went to the ballet last night.</i></p> <p>balloon /bə'luːn/ <i>noun</i> 1. a large ball which is blown up with air or gas 2. a very large balloon which rises as the air inside it is heated, sometimes with a container attached for people to travel in ■ <i>verb</i> to increase quickly in size or amount</p> <p>ban /bæn/ <i>noun</i> an official statement which says that people must not do</p>
--	--

FIGURE 7.3: Excerpt from the Easier English Basic Dictionary (Publishing, 2009)

Dictionary & Annotation Specificities: As Figure 7.3 illustrates, the dictionary has a very modern and clear typography and its markup system. Pronunciation, sense definition, examples and related entries are clearly marked up. We consider such a dictionary to have medium complexity level as it represents a few parsing challenges:

- On the structural level, the major challenge is the representation of related entries and notes which occur at the Lexical Entry and Sub-Sense levels. We annotate the notes only at the Sub-Sense level. We are aware that a post-processing is necessary to reassign some notes which are related to the whole entry. Related entries marked with diamonds are annotated within the lexical entry, the rest are annotated within Sub-Sense
- Other issues present in the versions of the dictionary that we had to deal with are related to the character encoding, especially for pronunciation constructs, which is different from the one supported by GROBID core. Lemmas are represented physically by redundant strings which we annotated as <dictScrap>s. Finally, the PDF we used originated from different steps, starting from the first download till the sampling process. We suspect that one of the PDF engines we used resulted in the introduction of some meta-data at the end of almost all the pages. We used that version for the tutorial⁶ series we organised for early system users. This anomaly surprisingly disappeared when we restarted the workflow from scratch to generate data for our experiments.

Mixtec-Spanish Dictionary (MxSp)

Description: We refer by Mixtec-Spanish⁷ dictionary to a PDF document that was compiled and published in 2009 by Jansen and Perez Jiménez from “Voces del Dzaha Dzahui”, a historical lexical resource published by the Dominican fray Francisco Alvarado in the year 1593. This bilingual dictionary, documenting a legacy lexicon, has over 370 pages with modern layout and short entries arranged in two columns (see Figure 7.1).

Dictionary & Annotation Specificities: We consider this dictionary to be the easiest to parse, given the length of the entries and the clear and clean typography. The parsing of this dictionary was the fruit of a collaboration with a linguist working on endogenous languages. The most challenging part was the sampling process, since the annotator did not speak both languages of the dictionary and had to collaborate with the linguist for the sampling process to cover the required logical and physical representativity explained in Section 7.2.2

⁶https://github.com/MedKhem/grobid-dictionaries/wiki/Docker_Instructions

⁷Mixtepec-Mixtec is an Otomonguean language spoken by roughly 9,000 – 10,000 people, and in addition to the native communities in Mexico, it is also spoken by small communities of people spread over several cities in the United States. The Spanish variation of this dictionary is Castilian.

Fang-French (FangFr) & French-Fang (FrFang) Dictionary

Description: The Fang-French⁸ & French-Fang dictionary (Galley, 1964) is a bilingual dictionary gathered in one volume that has over 500 pages of lexical entries split into two parts. The dictionary was published in 1964 but later digitised and the PDF version we used has a medium OCR quality. For our experiments, we consider the first part **FangFr**, containing over 390 pages, to be a separate dictionary from the second part **FrFang**, with over 140 pages.

Dictionary & Annotation Specificities: As Figures 7.4 and 7.5 show, the markup system is based on typographic change along with textual clues to markup field transition. For the page sampling milestone, we detected different anomalies in the OCRs such as messy text order of footnotes and some important text blocks (e.g. section titles, page numbers, parts of entries, etc.) disappearing in the extracted text. We suspect the origin of such anomalies to be the recognition of text regions as images by the OCR system. We encountered several challenges during the annotation process and the decisions we made could be gathered by parsing level for each part of the dictionary:

- **FangFr:** Two models represented difficulties for their annotation
 - **Sense:** Finding the boundaries of non-numbered senses was not obvious
 - **Sub-sense:** We used <note> for any prose description. However, it was hard to distinguish the different fields, especially <note> from <def>. We chose to annotate the first sentence or any short description (gloss) as <def>. Any that comes after, which is neither an example nor a cross reference, is considered a <note>
- **FrFang:**
 - **Lexical Entry:** We used <sense> to annotate
 - * Definitions followed by a set of translation equivalents. For instance:
 - **ANCÊTRE** aïeul, grand-père, grand-mère, mvam (h)
 - * A definition and one translation or more. For example:
 - **AMULETTE** fétiche, grigri, ñgîr (b)
 - * Simple translation equivalents and their pronunciation. For instance:
 - **ANANAS** aloès, ñkuba (h), ñkôkh ô sekh (hm)

We used <re> to annotate related entries, which often represents the rest of the lexical description of a dictionary article

⁸Fang is a language spoken by around 1 million people in several central African countries

<p>AYO — 58 —</p> <p>nuit à tel endroit. <i>Ba bukh ayôa</i> va, nous couchons ici. Syn.: <i>azakh (Aké)</i>.</p> <p>AYÔE (h) n.4, pl. <i>meÿôe</i> (vb <i>yôe</i> h). <i>AYôe</i> méria, action de faire chauffer de l'eau. Syn.: <i>anôghé méria, ayôgha mézim</i>.</p> <p>AYÔL (b) n.4, ss pl. (vb <i>yôl</i> b). Amer, mauvais, amertume physique ou morale. <i>Ebwsma ayôl</i>, fruit amer. <i>Mesô m'ayôl</i>, paroles amères. Contr.: <i>anôgha, nseghba</i>. <i>AYôl</i> est aussi une odeur, l'odeur de quelque chose qui est amer (voir <i>nyumayôle</i>).</p> <p>AYOM (m) n.4, ss pl. (vb <i>yômbé</i> b). Vieillesse. <i>AYôm e to nye e nyôl</i>, il se fait vieux. Syn.: <i>ayômbé</i>.</p> <p>AYOM (h) n.4, pl. <i>meÿôm</i>. De même tribu. <i>Ble nye bi ne ayôm</i>, lui et moi sommes de la même tribu. Un homme du clan des <i>Esemekôhâ</i> et un homme du clan des <i>Esisam</i> sont tous les deux de la tribu des <i>Esiandukh</i>. Si l'un va chez l'autre, il dit: <i>Ma ke ayôm dam</i>.</p> <p>AYOMBE (bm) n.4, pl. <i>meÿômbé</i> (vb <i>yômbé</i> b). Vieillesse. Syn.: <i>ayôm</i>.</p> <p>AYÔMLE (h) n.4, pl. <i>meÿômlé</i> (vb <i>yômlé</i> h). Bénédiction féliche, parole qui porte bonheur. Voir <i>seseghe nit e nit</i>.</p> <p>AYOR (h) n.4, ss pl. (vb <i>yôh</i> h). 1. Chaud, chaleureux. <i>Mesim me ne ayôh</i>, l'eau est chaude. Mais on dit très fréquemment: <i>mesim meÿôh</i>, eau chaude (et <i>mesim meÿôl</i>, eau froide). Voir <i>meÿôh</i>. — 2. Zèle, force, vivacité, tempérament bouillant. <i>Zal e ne ayôh</i>, le village est plein d'animation. Voir <i>alugha</i> (b). <i>Ntabga ayôh</i>, soyons zélés. Contr.: <i>awôl, Adôkh é e ne ayôh</i>, cette danse est très entraînante.</p> <p>AYVI (m) n.4, pl. <i>meÿvi</i>. Arbre à bois très dur qui sert à faire des béches en bois pour creuser des trous (éran), ou des manches de haches. Autre bois pour les mêmes usages: <i>ébam</i>.</p> <p>AYVIA (bm) n.4, pl. <i>meÿvia</i> (vb <i>yvia</i> b). Mécontentement, fait d'être fâché. Syn.: <i>éghé</i>.</p> <p>AZA (h) n.4, pl. <i>meza</i> (vb <i>za</i> h). Destruction. Syn.: <i>azôh</i>.</p> <p>AZAKH (h) n.4, pl. <i>mezahk</i>. 1. Endroit arrangé par le chimpanzé ou le gorille pour y dormir, ce qui lui tient lieu de maison. C'est assez près du sol. <i>Azahk e segha, azahk e éghé</i>. — 2. Campement d'homme, étape pour la nuit (<i>Aké</i>). Syn.: <i>ayôa</i>.</p> <p>AZAMÉ (h) n.4, pl. <i>mezamé</i> (vb <i>zámé</i> h). Action de pardonner, de laisser. Syn.: <i>bizamé</i>.</p>	<p>AZI — 59 —</p> <p>AZAN! (h) n.4, et interj. (<i>Atsé</i>) (vb <i>zâh</i> h). Imprécation pour le serment. <i>Azâh bôr!</i> <i>E bô be éga man-e-zâh me</i>, tous les miens qui sont morts, je jure par eux. Autre phrase analogue: <i>ma bele, me ta mimbin</i>, je l'affirme, je vois les morts.</p> <p>AZANÉ (h) n.4, pl. <i>mezané</i> (vb <i>zâh</i> h). Destruction, fait d'être détruit, de mourir en grand nombre. Syn.: <i>aza</i>.</p> <p>AZAP (b) n.4, pl. <i>mezap</i>. Nom d'arbre. Syn.: <i>azo</i>.</p> <p>AZEBE (bm) n.4, pl. <i>mezebe</i> (vb <i>zebe</i> b). Enterrement, funérailles.</p> <p>AZÉE (bm) n.4, pl. <i>mezéé</i> (vb <i>zéé</i> b). Action d'ensevelir. Action d'écarter les bûches du foyer pour étendre le feu. <i>Azéé mbâm, Azéé bishik, zi</i>.</p> <p>AZEGHA (bm) n.4, pl. <i>mezegha</i> (vb <i>zegha</i> b). Dernier soupir, fait d'expirer. Syn.: <i>ayé</i>.</p> <p>AZEM (b) n.4, pl. <i>mezem</i>. 1. Paquet de feuilles de manioc pilées avec sel, piment, viande ou poisson. — 2. Petite pile de au bord des ruisseaux.</p> <p>AZI (h) n.4, ss pl. (vb <i>zi</i> h). Aliment, chose qui se mange. Syn.: <i>bisi, nzi</i>.</p> <p>AZIE (h) n.1, pl. <i>bazie</i>. Boue que les femmes mettent sur les barrages de rivières (<i>myekh</i>) pour les rendre étanches.</p> <p>AZIE (bm) n.4, pl. <i>mezié</i> (vb <i>zi</i> b). Action d'enfoncer une pointe. <i>Azié aloh</i>.</p> <p>AZIE (h) n.4, pl. <i>mezié</i> (vb <i>zi</i> h). Action de manger. <i>Be vagha zi azié awôl</i>, ils ont mangé une fois. <i>Mesim mezéé</i>, ils ont mangé deux fois.</p> <p>AZIGHA (b) n.4, pl. <i>mezigha</i> (vb <i>zigha</i> b). <i>Azigha mam</i>, inventaire. Action de compter des choses.</p> <p>AZIGHÉ (h) n.4, pl. <i>mezighe</i> (vb <i>zighe</i> h). Action de brûler quelque chose. <i>Azighe tsé</i>, action de brûler un débroussement pour plantation.</p> <p>AZIGHÉ (h) n.4, pl. <i>mezighe</i> (vb <i>zighe</i> h). Incendie, fait de brûler soi-même.</p> <p>AZIKH (h) n.4, pl. <i>mezikh</i> (vb <i>zikh</i> h). Flot de paroles dans une palabre pour en finir vite.</p> <p>AZIMÉ (h) n.4, pl. <i>mezimé</i>. 1. Faute, tort, fait de se tromper ou de se perdre (vb <i>zimé</i> h). — 2. <i>Azimé égon</i>, pl. <i>mezimé me égon</i>, fin de lunaison, nouvelle lune (voir <i>atôé égon</i>), ou encore coucher de lune. <i>Azimé sé</i>, coucher du soleil (vb <i>zim</i> h).</p> <p>AZIN (b) n.4, pl. <i>meziñ</i> (vbs <i>ziñ</i> h et <i>ziñâ</i> h). 1. Variété de palmier rotin épineux et grimpatif qu'on voit surtout dans l'Abanga et dans la Lolo. Il ressemble au <i>akôn</i>, et sa base s'appelle aussi <i>akôn</i>. On emploie la base qui est grosse comme le bras comme râpe en enlevant seulement les pointes des épines. <i>Aziñ</i> servait autrefois à faire des flûtes appelées <i>niñ</i>. — 2. Nom de toute espèce de râpe. Voir <i>akastigha</i>.</p> <p>AZIR (m) n.4, ss pl. Lourd, poids, pesanteur (vb <i>zir</i> b). <i>Akôh azir</i>, pierre lourde. <i>Mô azir</i>, tête dure. <i>Oson wa ce nit azir</i>, la honte alourdit la tête. Syn.: <i>anémé</i>.</p> <p>AZO (b) n.4, pl. <i>mezo</i>. 1. Un des plus beaux arbres de la forêt équatoriale qui peut atteindre 40 mètres. Arbre à beurre. Nom commercial: moabi. (Mimusops djavé). Le tronc est très droit et les branches horizontales. Son bois est très apprécié pour la belle menuiserie. Le fruit (<i>ébona</i>) est gros et comestible, et les noyaux contiennent une bonne huile (<i>ézo</i>). Une légende ancienne veut que toutes les tribus des <i>Faâ</i> dans leurs migrations du nord au sud aient passé par une certaine cavité pratiquée entre un <i>azo</i> et un <i>ébon</i> qui se touchaient. Il n'y avait pas d'autre issue possible, et l'ouverture que l'on fit entre ces deux arbres s'appelle <i>azo mbôgha</i>. <i>Faâ bese be éga lôr azo mbôgha</i>. <i>Mbôgha</i> veut dire entaillé (vb <i>bôh</i> b). Syn. de <i>azo</i>: <i>azop</i>. Voir <i>ôndôh azo, byézo</i> (faux <i>azo</i>). — 2. <i>Azo éwé</i> (bb) n.4, pl. <i>mezo m'éwé</i>. Variété d'igname. Syn.: <i>abulé</i>.</p> <p>AZOKH (h) n.4, ss pl. (vb <i>zokh</i> h). Fait de nager, usage, natation. <i>A vagha ke ôsvi ayar y'azokh</i>, il a traversé la rivière à la nage. Syn.: <i>neoghga</i> (h).</p> <p>AZOM (m) n.4, pl. <i>mezom</i>. Amome, espèce de roseau à grande palme, très résineux, qui pousse dans les anciens</p>
--	---

FIGURE 7.4: Excerpt from the Fang-French & French-Fang Dictionary (Galley, 1964) (FangFr)

– **Sub-Sense**: The definition and translation equivalents are differently annotated from **MxSp**

- * For the **MxSp** dictionary, the differentiation between translations and definitions in the target language is too fuzzy. The recommendation made by the linguist was to annotate information that does not represent grammatical or usage information, as one or many definitions.

- * For the **FrFang** dictionary, it is straightforward to recognise the French definitions followed by translation equivalents in Fang

7.3 Experiment Series 1: Training with One Dictionary

Having introduced the general setup of the experiments we want to conduct, in this section we present a first series of experiments. Each of the experiments uses one dictionary to train all the selected models of the architecture. First, we report and discuss the evaluation of each model using the three feature templates introduced in Section 4.3.2. Then, we present the results of our experiments revealing further the behaviour of the best models for each dictionary, in particular the learning curve.

AIN	— 416 —	AMA	AMA	— 417 —	ANN	
ancien, <i>nyamère</i> (bh). Aîné et puîné, <i>nyamère ye nasimé, ngól ye nasimé, nidi ye nasimé</i> .						
AINSI comme cela, <i>ana</i> (h), <i>anena</i> (h), <i>nale</i> (h).		<i>ñke mbókòh</i> (bh). Aller et venir pour porter des choses, <i>iera</i> (h). Je suis allé porter une charge une seule fois, <i>me sagha ke lere mevgehe</i> . Aller et venir sans s'éloigner beaucoup, <i>lòlila</i> (b). Aller plus loin, <i>tór ósu</i> (bh). Aller et venir plusieurs fois, <i>tóra</i> (b). Aller à la selle, faire ses besoins, <i>soñy</i> (b), <i>ke meseniy</i> (bb), <i>nyakh mébí</i> (bh). Action d'aller à la selle, <i>me-nyakh</i> (b). Aller plus loin qu'on avait dit, <i>tóla</i> (b). S'en aller sans rien dire, comme en cachette, <i>monga</i> (b), <i>wonga</i> (b), <i>dorga</i> (b). Aller et venir <i>ndendeñ</i> (h), <i>tsitaim</i> (h). Allées et venues, va-et-vient, <i>ndendéñ</i> (h), <i>ndendéñ</i> (h), <i>tsitaim</i> (h), <i>eyeyá</i> (h). Aller voir une fille pour l'épouser, <i>sia ngón</i> (bh). Voir <i>saña</i> (b). Aller et retour, <i>meke ye meso</i> (bb).		<i>ndokh</i> , de <i>kómi</i> qu'on a sortis des coquilles, <i>bifé bí ndokh, bifé bí kómi</i> . Motte de <i>ndokh</i> , <i>ditéré ndokh</i> (voir <i>dítama</i>). Amande du fruit de l'arbre <i>afó, fo</i> (m). Fruit de <i>fofo, atóra fo</i> . Amande, noyau, <i>mboñ</i> (m). Amande sortie de sa coque, <i>usé</i> (h). Amande de palme, palmiste, <i>usé alen</i> (h).		travail, cloque d'eau après brûlure, <i>éyèñ</i> (h). Ampoules d'orties, <i>diyèñ bí sar</i> (hh). Eau qui se trouve dans les ampoules de brûlures, srosités des plaies, <i>bistaim</i> (h). Ampoules sur les doigts de pieds quand on a marché dans la boue, <i>sive</i> (bm), pl. <i>bestre</i> .
AISSELLE <i>fefé</i> (h), <i>mevgha</i> (bm).		AMANT amante, <i>ebon</i> (b). Mon ami, mon mari, ma femme, <i>ébo zam</i> .		AMUSEMENT jeu, <i>bivé</i> (h).	AMULETTE fétiche, grigris, <i>ngír</i> (b).	
AJOUTER augmenter, accroître. donner en plus, <i>kóghla</i> (b), <i>beré</i> (b). Ajouter un cadeau, <i>beré aber</i> (bb). Ajouter par-dessus le marché, <i>tséré</i> (b), <i>beré</i> (b). Ajouter des mailles à chaque tour en faisant un filet, <i>vam tan</i> (hh), <i>vam mevgha</i> (bh).		AMAS d'eau immobilisée par un barrage, <i>kwame</i> (h), <i>kwame a mezin</i> , <i>mekòh</i> (b).		AMUSER (s'), danser, chanter, <i>wakh</i> (b), <i>vé</i> (b), <i>tór</i> (h). S'amuser entre hommes et femmes, <i>nyókh</i> (b).	AMYGDALÉ abès des genévies, cloque, ampoule à la main ou au pied, <i>atsigbé</i> (h), <i>éyèñ</i> (h).	
AKÉLÉ tribu ou peuple du Gabon. Un Akélé, <i>mon Éngom</i> (b). Les Akéles, <i>Bi-Éngom</i> (b). Eux-mêmes s'appellent <i>Óngom</i> . Akélé est le nom que leur donnent les Galva.		AMÉ esprit, <i>nsisim</i> (h), <i>kón</i> (h).		ANANAS aloès, <i>ñkuba</i> (h), <i>ñkòkh ó sokh</i> (hm).	ANCÊTRE aïeul, grand-père, grand-mère, <i>mwam</i> (h).	
ALBINOS méi (h). Demi albinos, albinos foncé, <i>ewón é méi</i> (hh).		AMÉ amertume, mauvais goût, <i>ayól</i> (b). Fruit amer, <i>édmuma ayól</i> (hb). Paroles amères, <i>masé w'ayól</i> (hb). Serpent noir à l'odeur amère, <i>nyumayóle</i> (bb). Être amer, devenir amer, <i>yól</i> (b). Être amer, <i>yóle</i> (b). Un tel est devenu méchant (amer), <i>kale a mama yól</i> .		ANCIEN vieux, <i>ntól</i> (h).	ANCIEN <i>nyen</i> (h).	
ALCOOL eau-de-vie distillée, <i>meyókh</i> (b).		AMI amitié, fidélité, <i>añgom</i> (m), <i>mef</i> (h), <i>amú</i> (h), <i>mwie</i> (h), <i>ñgwa</i> (b). Echange de femmes par amitié, <i>amú</i> (h). Mon ami, <i>mie</i> (h), <i>é mie wam, é mie wam, añgom dam</i> . Mes amis, <i>benie</i> . Ton ami, tes amis, <i>nyubele, benyubele</i> . Son ami, ses amis, <i>ébele, bébele</i> . Notre ami, <i>añgom da</i> . Votre ami, <i>ébele éwa</i> . Leur ami, <i>ébele wo</i> . Vocatif : mon ami, <i>a mi, a mwi</i> (h). Être amis, se lier d'amitié, <i>ñgoma</i> (b), <i>lar añgom, yañ añgom</i> . Celui qui se lie facilement d'amitié, <i>ñgom-ñgoma</i> (bh). Amitié très proche, <i>añgom ne-baghdaa</i> (mh).		ANÉANTIR détruire, <i>yéi ntukh</i> (hb).	ANESTHÉSIE faire la narcoze, ôter la sensibilité, <i>yólé</i> (h).	
ALIGNÉ <i>éyarsyar</i> (bbm). Être aligné, <i>yarbe</i> (b), <i>yaré</i> (bm).		AMOMÉ roseau à grande palme, <i>azom</i> (m). Son fruit rouge, <i>ésón</i> (h). Endroit où il y a beaucoup d'amomes, <i>ésom</i> (m). Tiges sèches de l'amome qui sont résineuses et servent de flambeau, <i>kómé</i> (bm), <i>ñgwa</i> (b), <i>ñkua</i> (b), <i>ñkèkh</i> (h). Variété d'amome (remède comme le pigment), <i>lum dule</i> (hh). Il a de l'allure, <i>a yem-e-wule</i> (bb).		ANGE <i>añgal</i> (h). Vient du grec.	ANGLE droit, équerre, <i>akan</i> (m). Les quatre angles d'une caisse, <i>mekan m'évora mené</i> . Angle de maison, <i>akan e ndo</i> (mh). Angle de deux surfaces, arête, <i>ñkòá ó ñgbe</i> (bh).	
ALIGNEMENT d'hommes debout côte-à-côte pour la chasse ou la danse, alignés de front, <i>mbakh</i> (b), <i>ñka</i> (h). S'ils sont l'un derrière l'autre, c'est <i>nosma</i> (b). Alignement d'hommes ou d'arbres, <i>nyoñ ó bór</i> (bh), <i>nyoñ bíi</i> (bh). Syn. : <i>nsama</i> (h), <i>ndóh</i> (b). Alignement tout droit par rang de taille, rangée en ligne droite, <i>ñgelga</i> (b). Alignement de maisons qui se touchent, <i>ñkagha zai</i> (hm).		ALLUMER du feu, <i>kóba si</i> (bh), <i>kóba ndóba</i> (bh), <i>lara si</i> (hh), <i>kóba mbèkh</i> (bh). Allumer un grand feu, <i>bekh ndóba</i> (hh).		ANGLOIS <i>ñges</i> (m), <i>ñges</i> (m). L'Angleterre, <i>afan ñges</i> . Un Anglais, <i>moné ñges</i> . Les Anglais sont arrivés, <i>ñges de sóa</i> .	ANGOISSE souci, crainte, inquiétude, <i>bivekh</i> (h), <i>bivekh</i> (h), <i>akelé nlem</i> (bm).	
ALIGNER les autres, <i>yaré</i> (b). S'aligner en une rangée, être en ligne, <i>yaré</i> (bm), <i>yarbe mbakh</i> (bh). S'aligner en deux rangées, <i>yarbe mimbakh midé</i> . Alignez-vous bien, <i>yarba ne-nyoñ</i> (h). Marcher de front, <i>yarbe éfakh</i> (b).		ALLONS ! <i>néa-ñkèña</i> ! (hh), <i>mnagha bí é</i> !		ANGUILLE intermédiaire entre poisson et serpent, un mâtre long. Elle a des dents, <i>énavolé</i> (bm).	ANIMAL bête, viande, <i>tsir</i> (h). Animal féroce qui tue les hommes, <i>ébbí</i> (b). Petit d'animal, <i>éwakh</i> (h), <i>éyel</i> (b). Animal domestique (chèvre, mouton, cochon, chien, chat, poule, canard), <i>éyena</i> (h). Animal vivant fétiche préparé par des drogues (<i>mebyéñ</i>), <i>éyena é byañ</i> (hh), <i>ñkèkh ó byañ</i> (bb), <i>é tsir de mana yéñ, éyeyéña</i> (b). Serpent fétiche, <i>éyeyéña</i> (h). Animal qui se couche, <i>mbakhé</i> (b), pl. <i>mimbókhé betsir</i> . Petit animal, <i>étsisair</i> (h), <i>moné tsir</i> .	
ALIMENT nourriture, ce qui se mange, repas, <i>azi</i> (h), <i>bici</i> (h), <i>nsia</i> (h).		ALLOUS <i>néa-ñkèña</i> ! (hh), <i>mnagha bí é</i> !		ANNEAU baguette, <i>akana</i> (h). Anneau de cuivre, <i>akana ñgò</i> (hm). Anneau d'or, <i>akana kòñ</i> (hb). Anneau de cuivre à la		
ALLAITER l'enfant, <i>nya mon abí</i> (hh).		ALORS (dans le récit), <i>ane</i> (bm). Alors ils vinrent, <i>ane be ñga so</i> . Alors, en ce temps-là, <i>añgè éto</i> (mm), <i>vale</i> (h). Alors (début d'un discours), <i>wena</i> (h).		ANNEAU baguette, <i>akana</i> (h). Anneau de cuivre, <i>akana ñgò</i> (hm). Anneau d'or, <i>akana kòñ</i> (hb). Anneau de cuivre à la		
ALLEMAND les Allemands, <i>Kofini</i> (bb).		ALORS (dans le récit), <i>ane</i> (bm). Alors ils vinrent, <i>ane be ñga so</i> . Alors, en ce temps-là, <i>añgè éto</i> (mm), <i>vale</i> (h). Alors (début d'un discours), <i>wena</i> (h).		ANNEAU baguette, <i>akana</i> (h). Anneau de cuivre, <i>akana ñgò</i> (hm). Anneau d'or, <i>akana kòñ</i> (hb). Anneau de cuivre à la		
ALLER le (b), <i>nú</i> (b). Va, <i>keñ</i> (h), <i>keñé</i> (h). Va-t'en, <i>kórgo</i> (h). Je m'en vais, <i>ma kóre</i> (h). Aller bien ensemble, <i>borra</i> (h), <i>gia</i> (b), <i>yala</i> (b), <i>kòkh</i> (b), <i>wéha</i> (h). Contr. : <i>eyéle</i> (b). Aller à la dérive, <i>fep</i> (h). Ce qui descend au fil de l'eau, <i>éfeféba</i> (h). S'en aller avec colère, <i>kéga</i> (h). Aller chercher quelque chose ou quelqu'un, <i>teghé</i> (h). Va chercher ton père, <i>keñé teghé ésoé</i> . Celui qui va en voyage, <i>ñke nzen</i> . Celui qui va en prison,		ALORS (dans le récit), <i>ane</i> (bm). Alors ils vinrent, <i>ane be ñga so</i> . Alors, en ce temps-là, <i>añgè éto</i> (mm), <i>vale</i> (h). Alors (début d'un discours), <i>wena</i> (h).		ANNEAU baguette, <i>akana</i> (h). Anneau de cuivre, <i>akana ñgò</i> (hm). Anneau d'or, <i>akana kòñ</i> (hb). Anneau de cuivre à la		
		ALORS (dans le récit), <i>ane</i> (bm). Alors ils vinrent, <i>ane be ñga so</i> . Alors, en ce temps-là, <i>añgè éto</i> (mm), <i>vale</i> (h). Alors (début d'un discours), <i>wena</i> (h).		ANNEAU baguette, <i>akana</i> (h). Anneau de cuivre, <i>akana ñgò</i> (hm). Anneau d'or, <i>akana kòñ</i> (hb). Anneau de cuivre à la		
		ALORS (dans le récit), <i>ane</i> (bm). Alors ils vinrent, <i>ane be ñga so</i> . Alors, en ce temps-là, <i>añgè éto</i> (mm), <i>vale</i> (h). Alors (début d'un discours), <i>wena</i> (h).		ANNEAU baguette, <i>akana</i> (h). Anneau de cuivre, <i>akana ñgò</i> (hm). Anneau d'or, <i>akana kòñ</i> (hb). Anneau de cuivre à la		

FIGURE 7.5: Excerpt from the Fang-French & French-Fang Dictionary (Galley, 1964) (FrFang)

7.3.1 Feature Engineering Experiments

The goal of these experiments is to understand the impact of feature engineering on the outcome of each model based on common characteristics of the dictionary samples at each parsing level. For training⁹ and testing each model, we selected annotated data taken from different parts of a dictionary. We tried to respect as much as possible $\frac{3}{4}$ $\frac{1}{4}$ ratios for splitting the training/testing data, but it was not always possible given the reasons explained in Section 7.2.2. For **MxSp**, we wanted to challenge the models by trying $\frac{2}{3}$ $\frac{1}{3}$ ratios.

Experiments

In the following, we present, model by model, the evaluation¹⁰ of seven parsing levels for each sample of the five selected dictionaries. For each model, except **Dictionary Segmentation**, we experimented with the three categories of templates: Unigram, Bigram and Engineered. The number of pages used for training/testing is different from one dictionary to another, depending on the complexity of the sample. More information about the number of training pages is provided in Section 7.3.2.

We report the *macro-average* F1-score¹¹ of each label recognised in every sample, as well as of the *macro-average* F1-score of *field level* evaluation for all the labels of a model. We chose macro-average measurement because of the imbalanced number of instances of the classes of each model. The field level choice made the feature engineering more challenging, as we have often noticed an improvement at the token level but one single wrongly recognised token, quite often a full stop or an OCR garbage, sanctions the performance on a whole text block. For labels not encountered in a dictionary sample, their scores are rendered as -1. One dictionary, **FrFang**, did not have any grammatical information to be parsed and consequently the **GramGrp** model was not activated, which explains why its scores are rendered as -1.

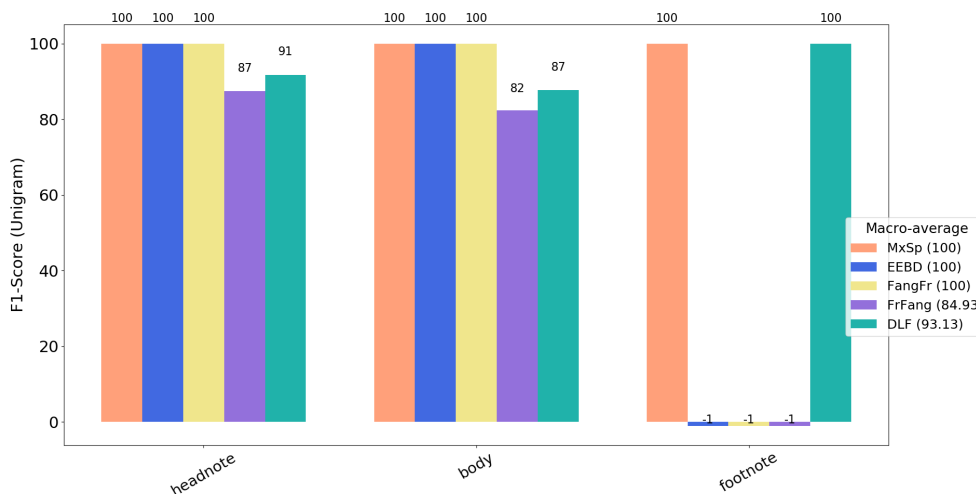
Dictionary Segmentation Figure 7.6 shows an identical performance of the two categories of templates with perfect recognition for 3 dictionaries and lower recognition error by the bigram variation (less than 4%) for DLF, despite the sample's relatively complex layout. The lowest score of both templates observed for FrFang reflects the struggle of the model to differentiate

⁹Training data in this context also includes the development set. The GROBID training module automatically manages the split between the two datasets

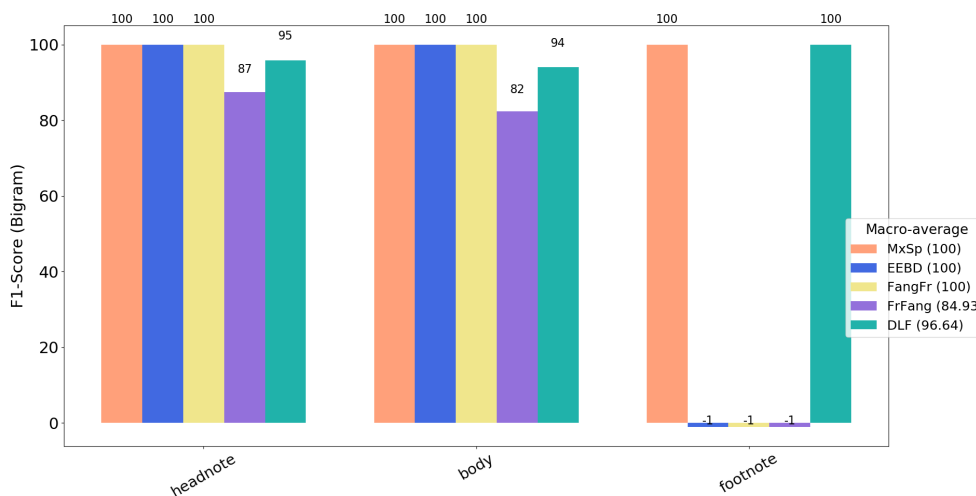
¹⁰all evaluation figures in this chapter are generated using <https://github.com/MedKhem/grobid-benchmark>

¹¹We have noticed that some experiments do not have always the exact same results, when they are run on different occasions. The difference is estimated to be around 1.5% in the F1-Score. As we discovered such a minor anomaly at the very end of this thesis, we did not have enough time to find the exact origin of such a behaviour in the WAPITI implementation we used. But we suspect the dynamic allocation of threads, used to perform the training, to be one main source of this inaccuracy. Therefore, our comparison for the outcome of the different templates takes this into consideration by tolerating such differences in the selection of the best template category

headnotes from the bodies of pages. This is mainly due to the fact that some headnotes were not recognised as text regions by the OCR system and it consequently affects the learning of the model when they disappear from the text sequences.



(a) Unigram Templates



(b) Bigram Templates

FIGURE 7.6: Mono-sample Evaluation of the Dictionary Segmentation Model Using two Classes of Templates

Dictionary Body Segmentation Figure 7.7 shows no difficulties to recognise the boundaries of entries in digital-born samples which have short and mid-size dictionary articles along with consistent clear layout and typography. Difficulties appear for identifying bigger entries in digitised samples. The fluctuant score for the <dictScrap> label, which we used to tag dictionary sections, is related to its limited occurrences in the training data (a ratio

of less than 2 per 400 instances). In general, engineered templates seem to combine the best of the bigram and unigram features. They have slightly better results, especially for recognising less frequent instances, with Macro-average F-1 scores exceeding 95% for 4 dictionaries.

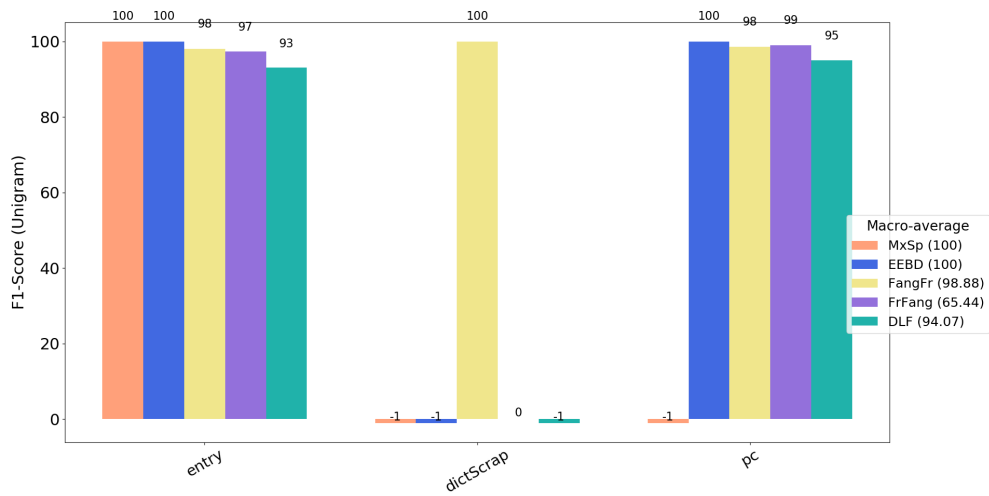
Lexical Entry For this model, the comparison of performances represented in Figure 7.8 is reduced between the bigram and the engineered templates with a mutual competition on certain samples. The boosted general scores for **EEBD** and **DLF** illustrate the feature engineering impact that we aimed at by tweaking the bigram templates. The difficulty for both templates concerns the recognition of less frequent constructs: *notes*, *related entries*, *usage*, *cross references* and OCR garbage, tagged as <dictScrap>. Nevertheless, longer range features again prove to be useful for recognizing less frequent labels in relatively long sequences. The parsing performance for *inflected forms* and *etymology* is insensitive to the addition of templates to the bigram combination. More information about neighbouring tokens seem to slightly harm the identification of lemma blocks coming always at the beginning of a main sequence, where the recognition of semantic blocks presents a small variation between the two best templates that depends on the sample.

Form As shown in Figure 7.9, the three classes of templates have very similar performances for all the dictionaries except for **MxSp**, where the bigram templates win. For the latter, the overall recognition is good with individual F1-score for labels exceeding 95%, except for <part>, <lbl>, <pc>, and <usg>. The difficulties in recognising these constructs mainly stems from their low number in the training data. For <part> the model with bigram templates manages to reach a score of 85%, despite the occurrence limitation. This shows the positive impact of the previous label and the negative impact of long range features on the predictions of the model.

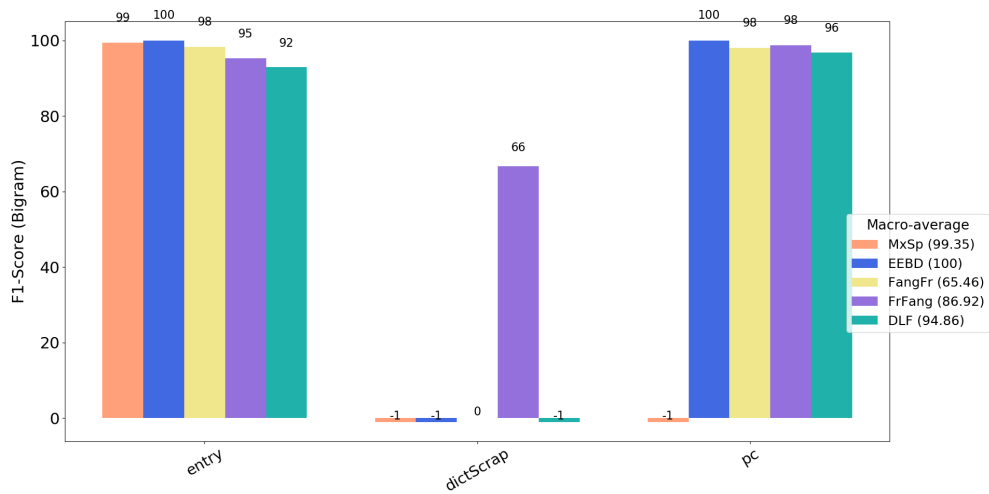
GramGrp Unigram templates beat the rest of the templates with almost perfect recognition, where the worst score exceeds 97% for the **FangFr** sample (see Figure 7.10). For this fine grained lexical information, the sequence labelling model gives its best when the clues used are more focused on the current token.

Sense The sense model has the smallest number of labels to be predicted on relatively long text sequences. Figure 7.11 shows a stable performance of the bigram templates over all samples. But all the templates still have comparably good, often perfect, results.

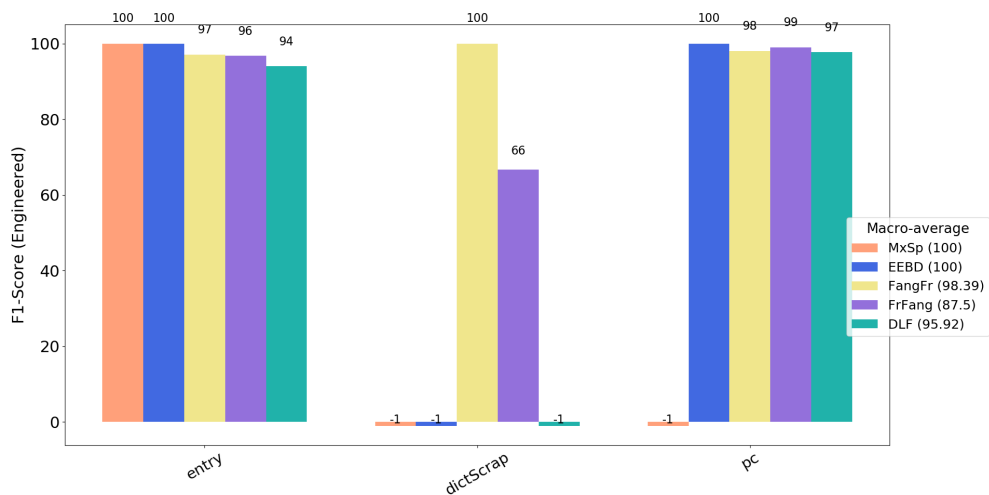
Sub-Sense This is the model with the biggest number of classes to be differentiated with over 10 labels. From the results reported in Figure 7.12, we notice that the longer the text of the sense is, the worse the Unigram templates perform. Except for the less frequent labels, the other two templates



(a) Unigram Templates

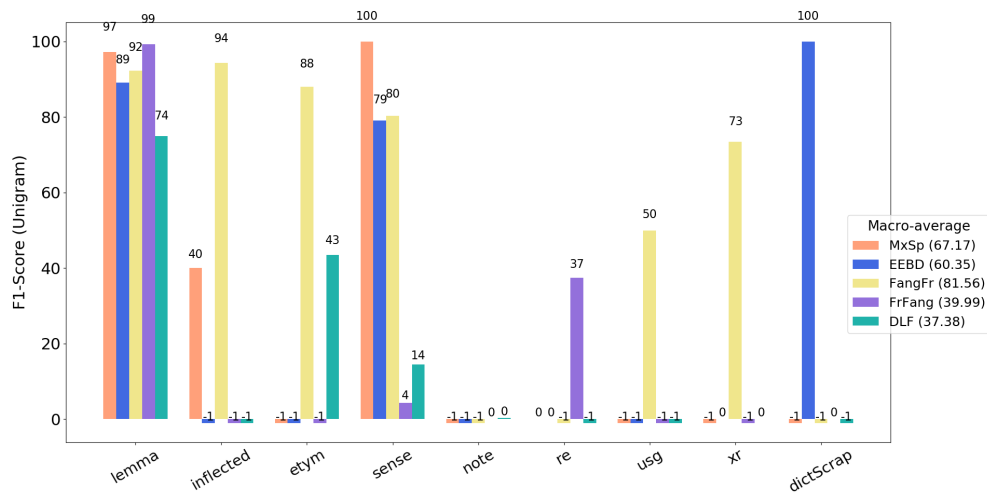


(b) Bigram Templates

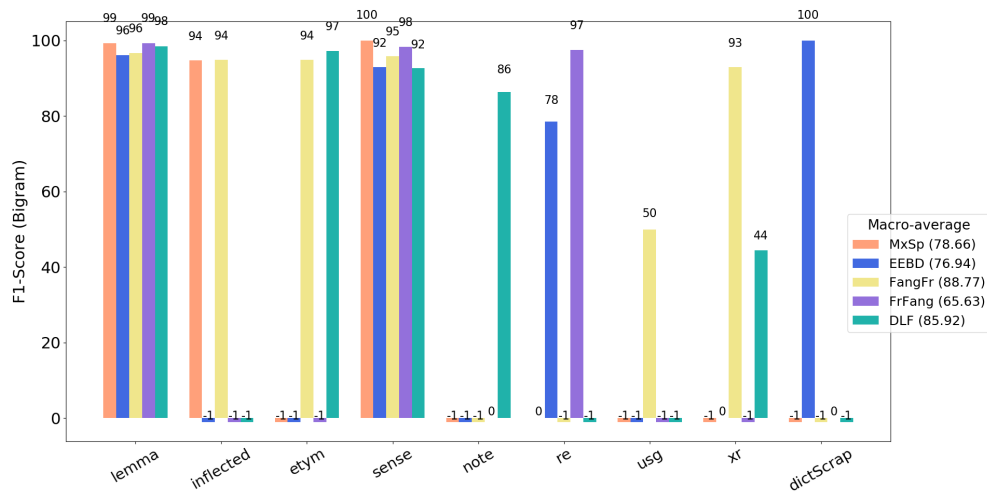


(c) Engineered Templates

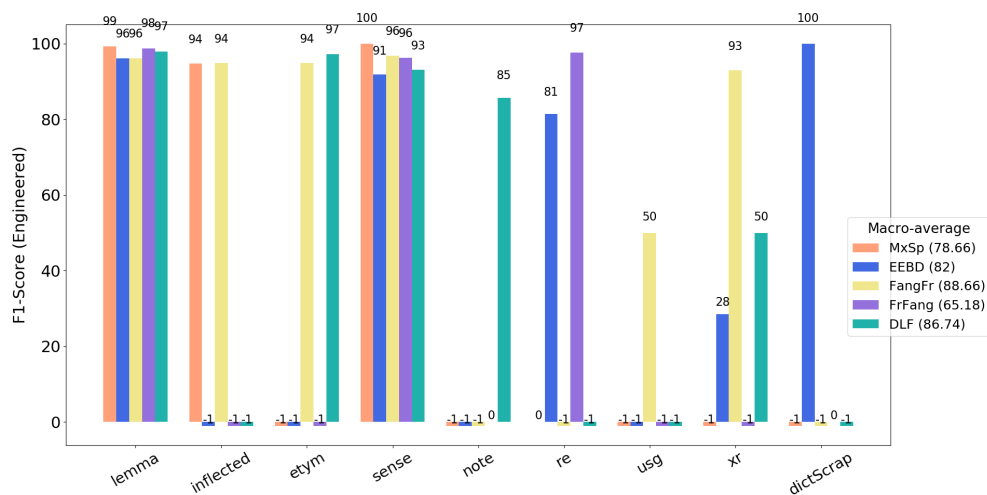
FIGURE 7.7: Mono-sample Evaluation of the Dictionary Body Segmentation Model Using three Classes of Templates



(a) Unigram Templates

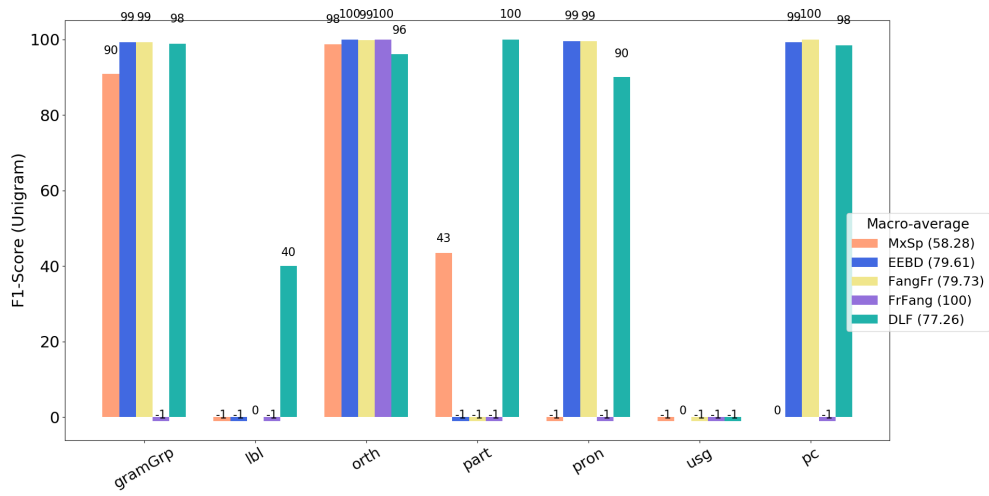


(b) Bigram Templates

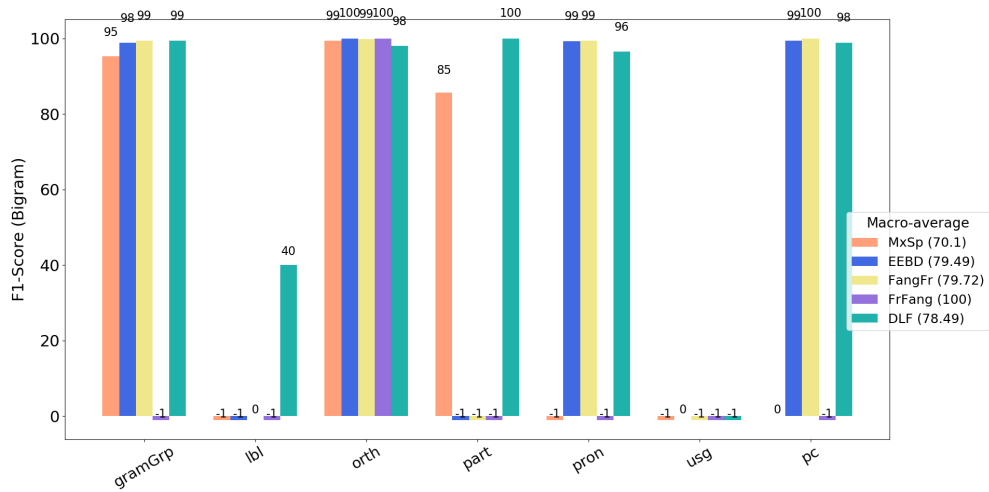


(c) Engineered Templates

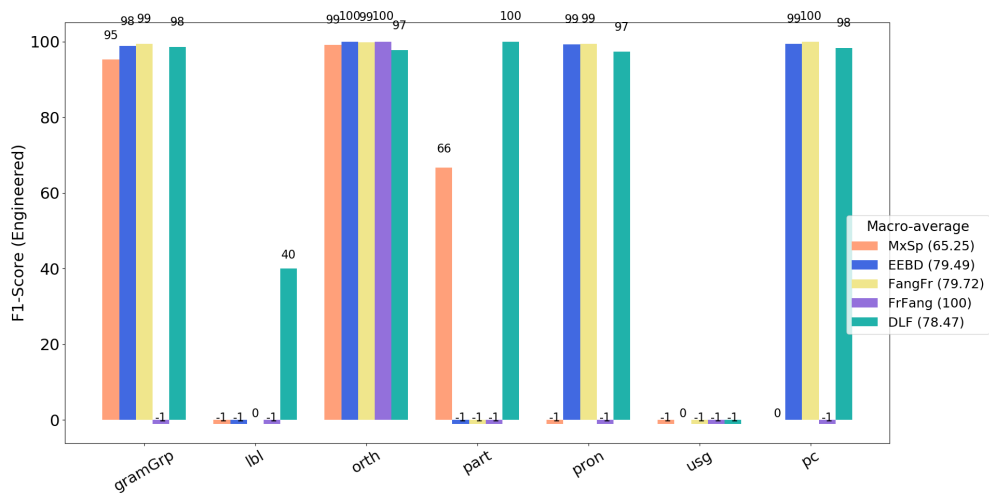
FIGURE 7.8: Mono-sample Evaluation of the Lexical Entry Model Using three Classes of Templates



(a) Unigram Templates

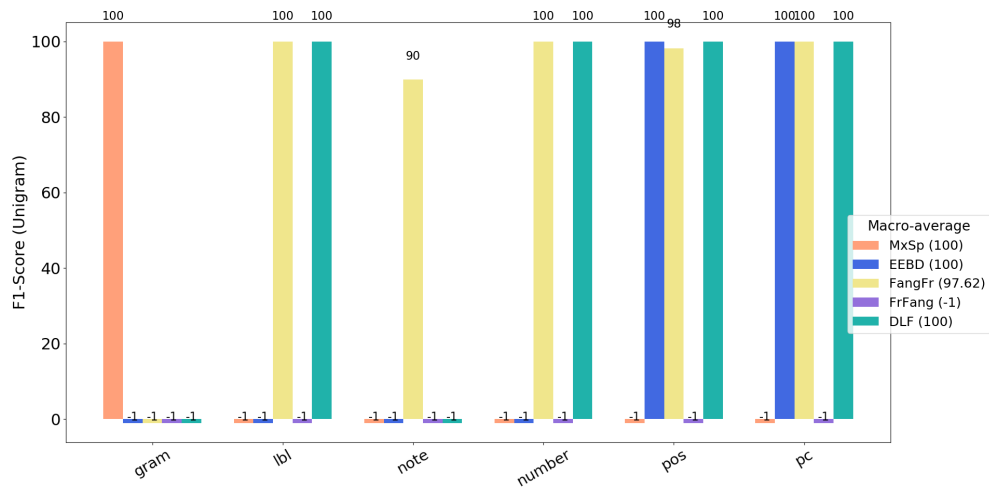


(b) Bigram Templates

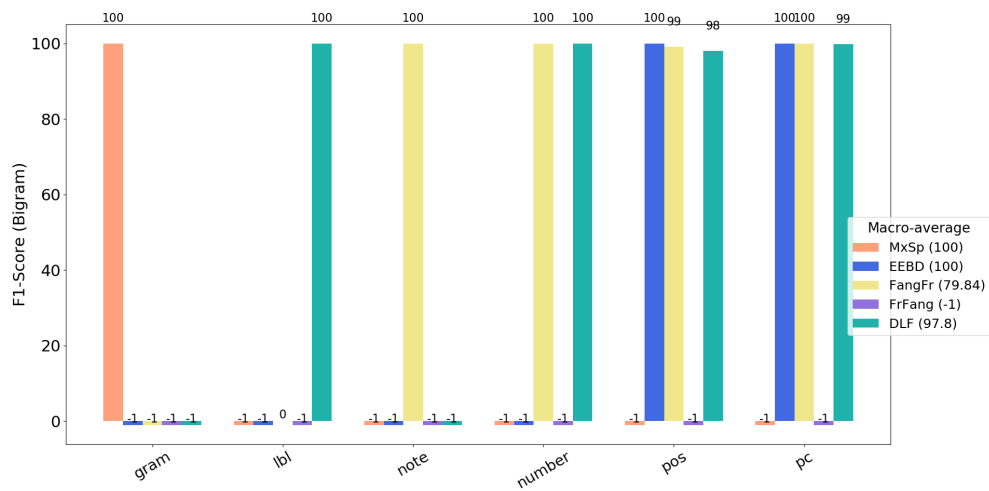


(c) Engineered Templates

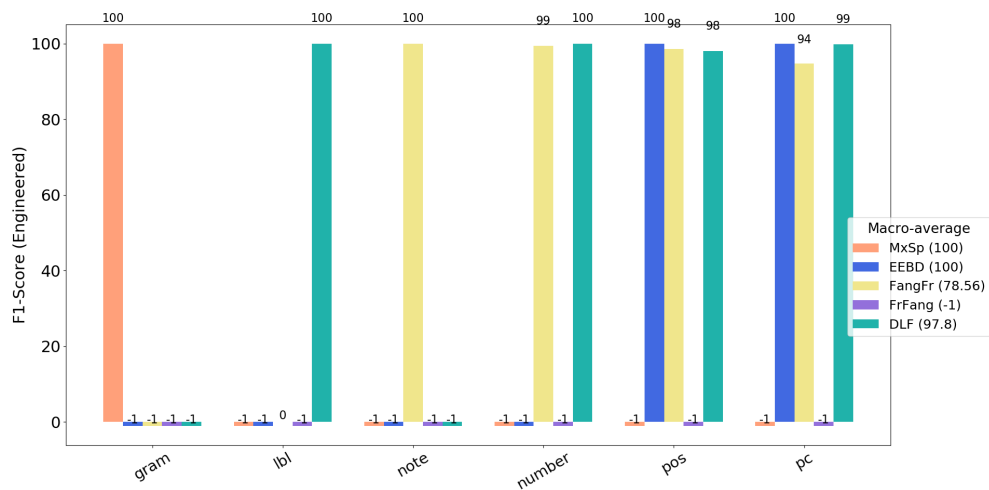
FIGURE 7.9: Mono-sample Evaluation of the Form Model Using three Classes of Templates



(a) Unigram Templates

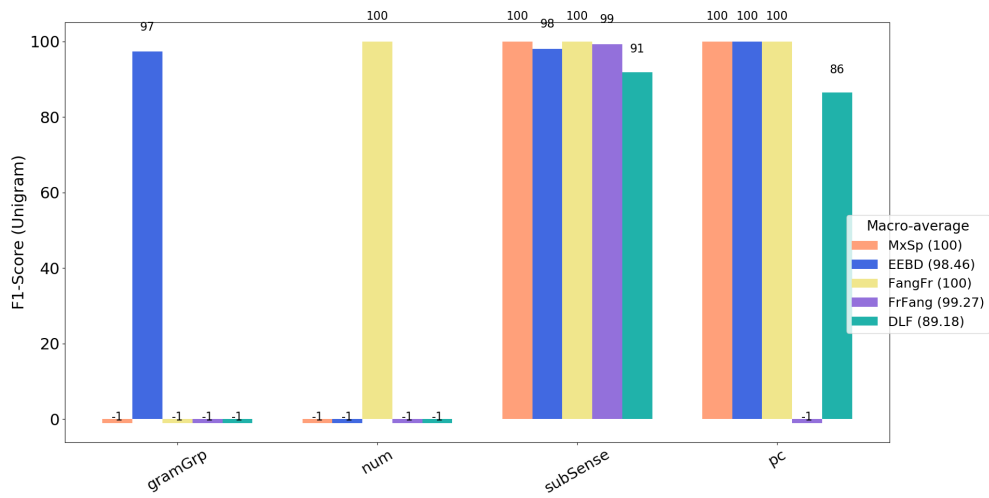


(b) Bigram Templates

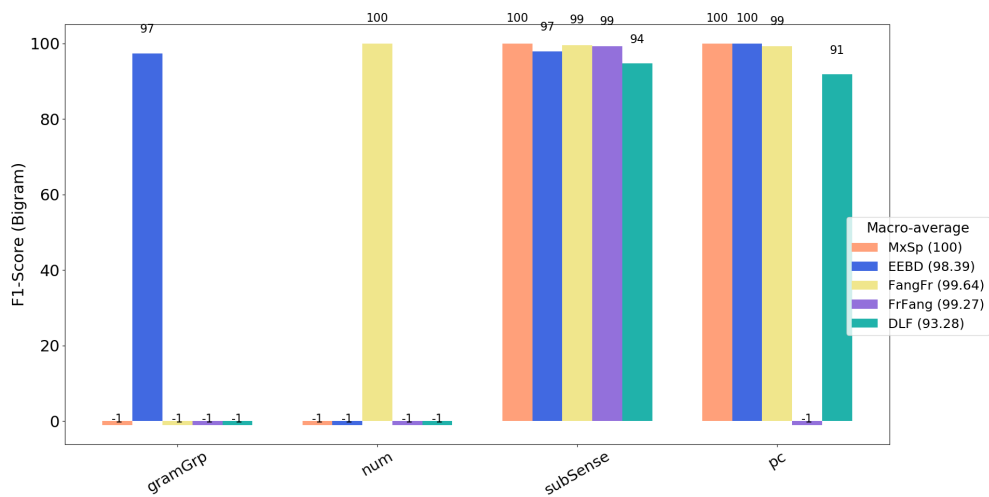


(c) Engineered Templates

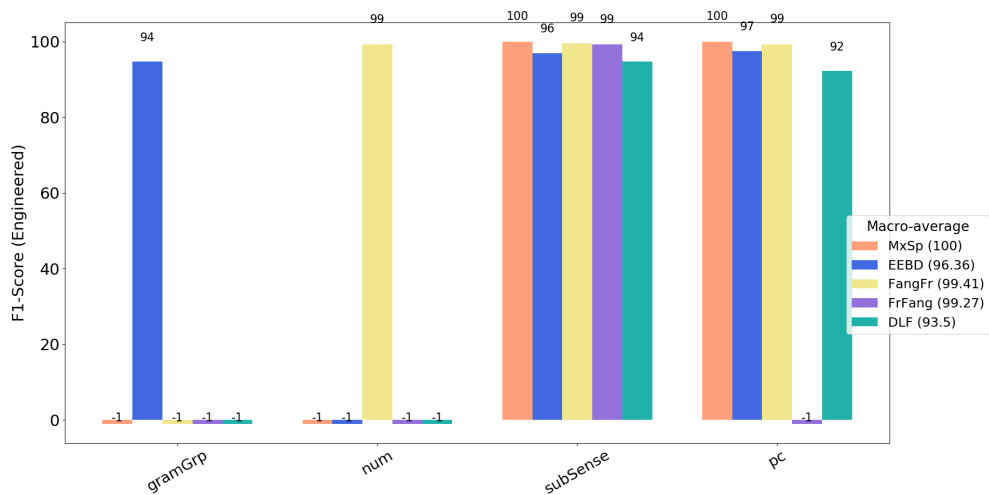
FIGURE 7.10: Mono-sample Evaluation of the GramGrp Model Using three Classes of Templates



(a) Unigram Templates



(b) Bigram Templates



(c) Engineered Templates

FIGURE 7.11: Mono-sample Evaluation of the Sense Model Using three Classes of Templates

give very accurate results with a slight extra-strength for the bigram templates.

Discussion

The previous experiments can be summarised by Table 7.5, where **U** stands for Unigram templates, **BB** for Basic Bigram templates, **EB** for Engineered Bigram templates and **A** for any category of templates.

From these experiments, we can draw the following conclusions:

- For parsing short text sequences, as in the case of the GramGrp model, simple Unigram features are more efficient.
- For the rest of the models, where text sequences are relatively longer, the label of the previous token is a very important clue.
- Engineered templates, with long-range features, are generally the best choice for labelling lexical constructs in longer text sequences
- The Bigram templates beat in many cases the Engineered templates, despite the previous conclusions made by (Lavergne, Cappé, and Yvon, 2010), favouring larger numbers of features per model.

Model	MxSp	EEDB	FangFr	FrFang	DLF	Best
<i>Dictionary Segmentation</i>	A	A	A	A	BB	BB
<i>Dictionary Body Segmentation</i>	A	A	U EB	EB BB	EB	EB
<i>Lexical Entry</i>	BB EB	EB	BB EB	BB EB	EB BB	EB
<i>Form</i>	BB	A	A	A	A	BB
<i>GramGrp</i>	A	A	U	-	U	U
<i>Sense</i>	A	U BB	A	A	EB BB	BB
<i>Sub-Sense</i>	U BB	BB	EB BB	BB EB	EB BB	BB

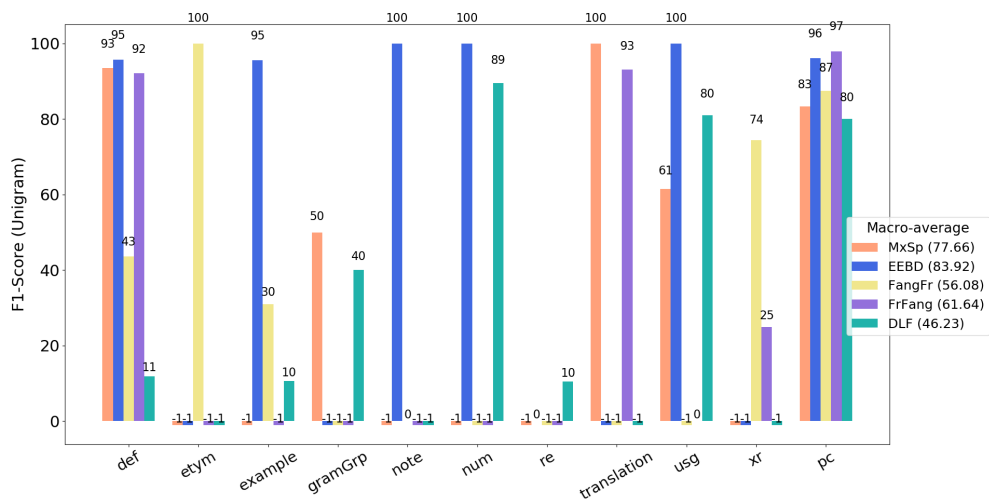
TABLE 7.5: Summary of the First Series of Experiments

7.3.2 Learning Curve Experiments

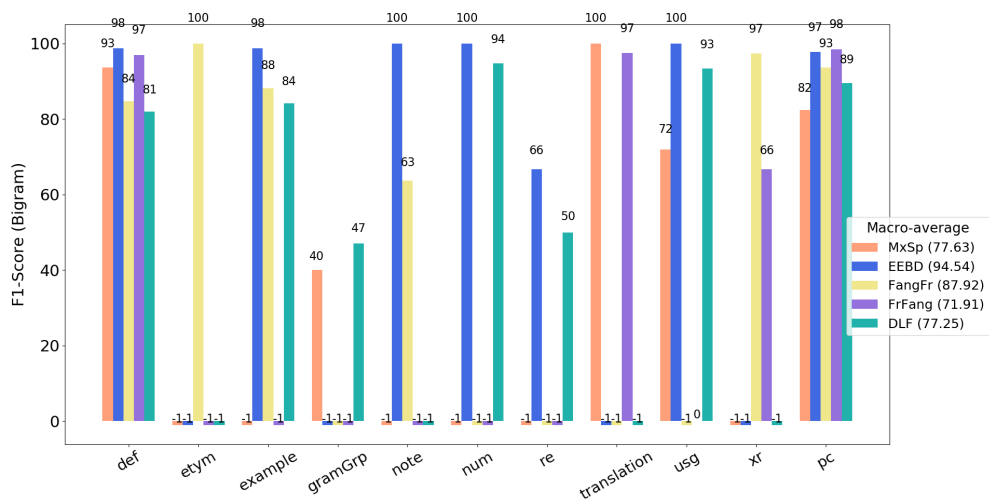
A second behaviour of the models we want to know more about, is actually the learning curve of the best models selected in the previous experiments.

Experiments

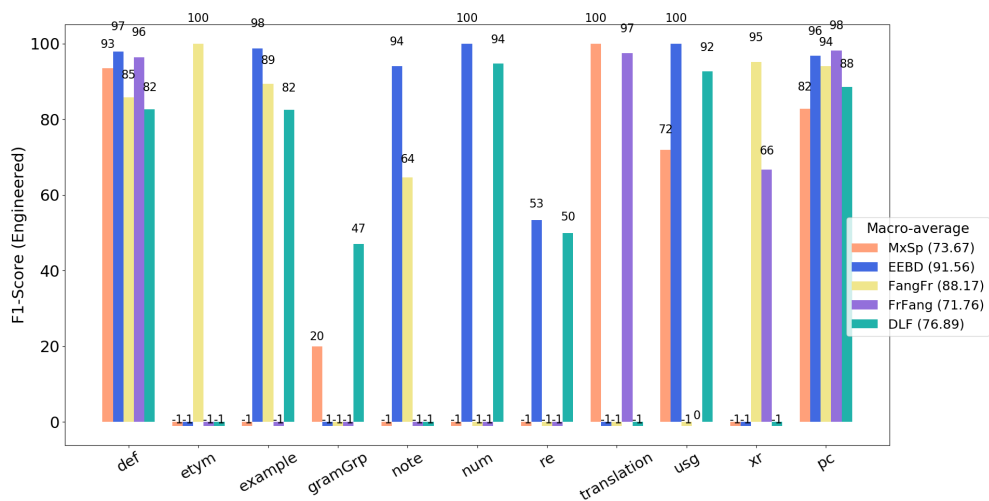
To generate such a learning curve, we designed our experiments by having for each model to be trained on one dictionary sample, 4 batches that do not necessarily contain the same number of annotated pages. The pages in each batch are chosen from different parts of the dictionary. For each dictionary we report the number of annotated pages for each model and the



(a) Unigram Templates



(b) Bigram Templates



(c) Engineered Templates

FIGURE 7.12: Mono-sample Evaluation of the Sub-Sense Model Using three Classes of Templates

performance of each model after adding a new batch of annotated pages. As pointed out earlier, the number of pages annotated for each dictionary varies according to the complexity of the sample. In what follows, we present the learning curve of all models observed for each sample in our dictionary pool.

DLF The evolution of the learning of the different models illustrated in Figure 7.13 shows a convergence and a stability for a number of pages between 35 and 40 for 6 models. The seventh model, Dictionary Segmentation, reaches such convergence after approximately twice the number of pages. This is due to the fact that the scope of such a model requires more pages to cover most of the variations in the structure and layout of the pages that have a dense text and slightly changing clues.

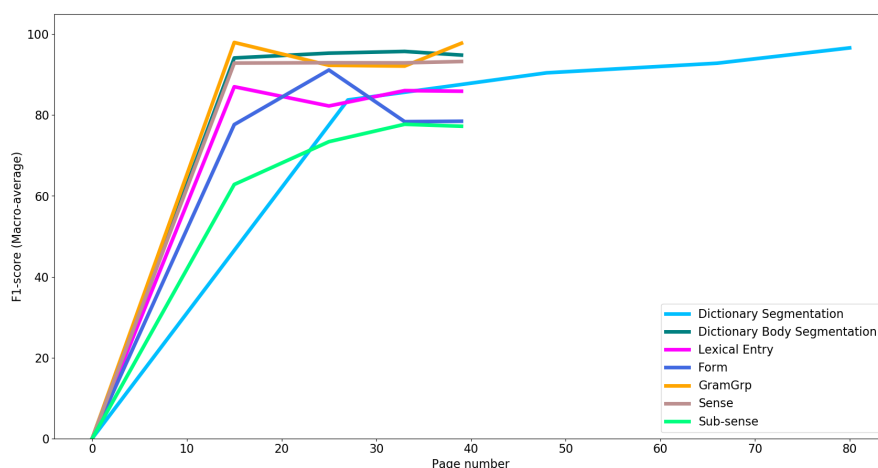


FIGURE 7.13: Learning Curve of the Different Models Given the Number of Training Pages from DLF

EEBD The curve reported in Figure 7.14 shows that fewer pages are required for this less complicated sample to reach the best performance and stability of the different models. Moreover, 10 pages were enough to achieve a high labelling accuracy. The additional pages were useful to stabilize the recognition of certain labels and further strengthen the quickly achieved accuracy.

MxSp Fewer than 10 pages were enough for the second digital-born bilingual dictionary to reach perfect performance for 4 models. Except for the Form model, the other the models seem to be reaching a plateau by the 8th page. It is worth remembering, that the low F1 score in such a curve (see Figure 7.15) is actually due to difficulties recognising less frequent labels.

FangFr For 5 models, 10 pages enabled the labellers to reach a plateau of their best performance (see Figure 7.16). The Sub-Sense model gives better

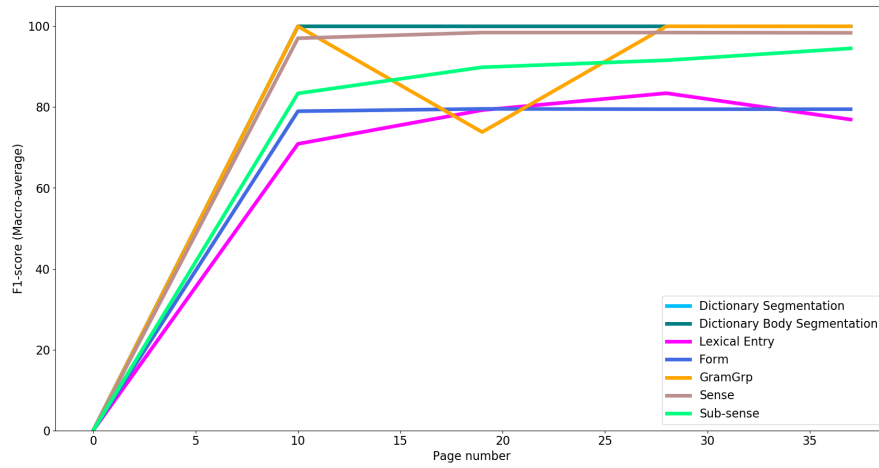


FIGURE 7.14: Learning Curve of the Different Models Given the Number of Training Pages from EEBD

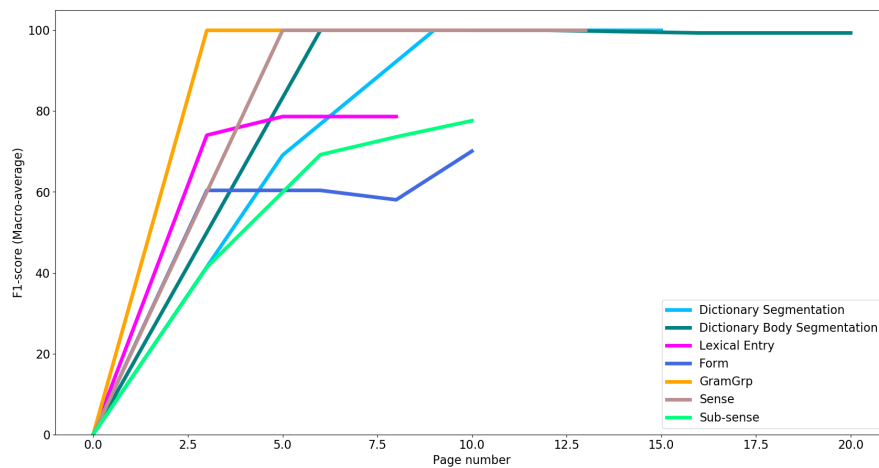


FIGURE 7.15: Learning Curve of the Different Models Given the Number of Training Pages from MxSp

results much earlier, after only 6 pages, but its performance stabilizes after 20 pages. The Form model is as quick as the first 5 models, but `<lbl>`, a rarely occurring label, causes a fluctuant learning behaviour due to the uncertainty that seems to grow with more added batches.

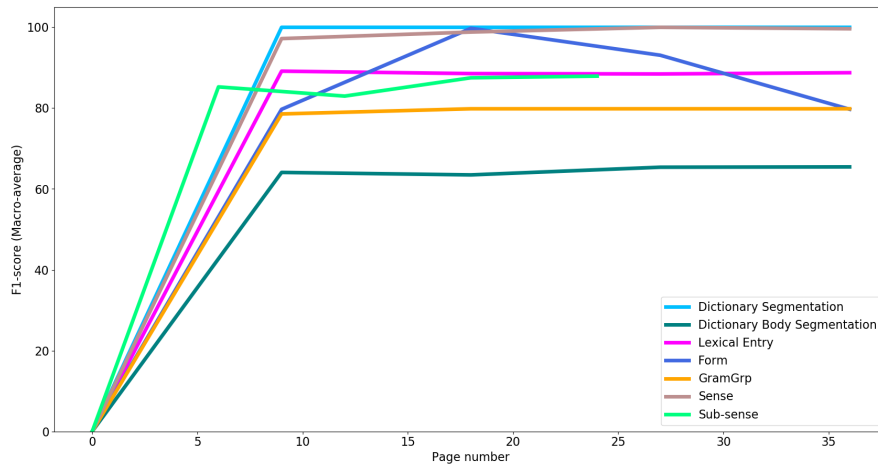


FIGURE 7.16: Learning Curve of the Different Models Given the Number of Training Pages from FangFr

FrFang The threshold of 10 pages also remains enough for 4 models, applied to the FrFang sample, to reach the plateau (see Figure 7.17). Sub-Sense and Dictionary Body Segmentation needed respectively 10 and 15 pages more before their learning curve stabilised.

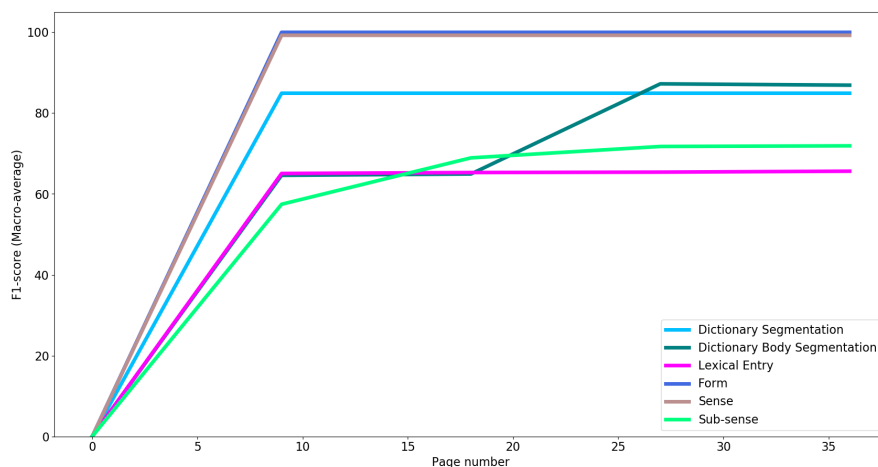


FIGURE 7.17: Learning Curve of the Different Models Given the Number of Training Pages from FrFang

Discussion

Through these experiments we can notice a rapid convergence of the different models, where a plateau could often be reached after annotating 10 pages. For models operating on longer text sequences, such as the case for the Dictionary Segmentation model, more pages are required. In general, the difference in the rapidity and the evolution of the learning are related to a number of factors. Digitisation origin, the quality of the OCRs, the representativity of the sample and the presence of less frequent labels are among these determinant factors.

7.4 Experiment Series 2: Training with Multiple Dictionaries

The models, upon selecting their best feature templates, often yield a very accurate recognition of the mono-sample lexical structures they are exposed to. We aim at further investigating the behaviour of these models, tuned with the same feature templates, by experimenting their training with multi-samples at once. The goal is to study the generalisation capacities of each model and the impact of the combination of samples on the learning curves. To do so, we selected 5 sample combinations:

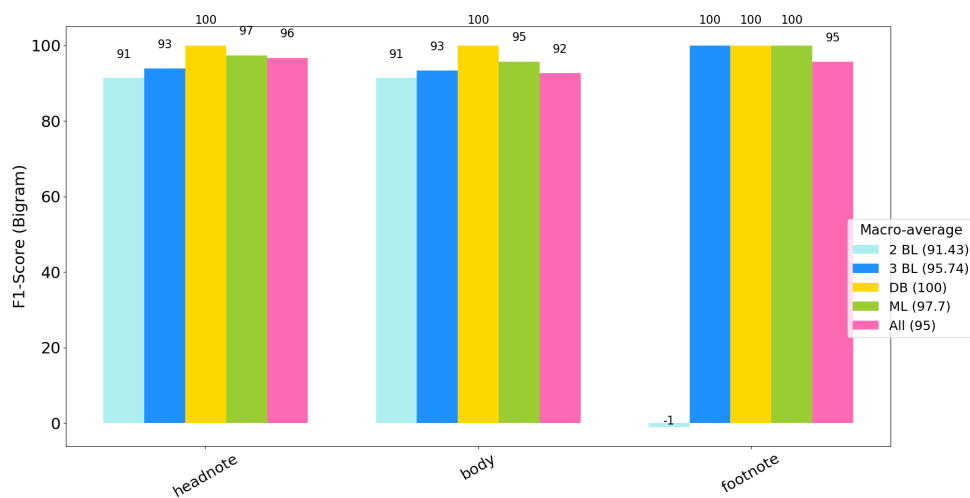
- **2 BL:** stands for 2 Bilingual dictionaries that have the same pair of languages and are most likely designed and compiled by the same lexicographer(s). From our pool, this represents the case of the two parts of the Fang-French and French-Fang dictionary
- **3 BL:** stands for 3 Bilingual dictionaries where the language pairs are different and the design and the compilation of the lexicons are carried out by different lexicographers. For that, we added the Mixtec-Spanish sample to the previous combination.
- **DB:** stands for Digital-Born samples, which are represented in our dictionary pool by the English and Mixtec-Spanish dictionaries. Such samples have a very little amount of noisy text as the main source, the OCRs, are present in this combination.
- **ML:** stands for Monolingual samples where the lexical description in the dictionary articles is in the same language and have, on average, longer text sequences than those in the bilingual samples. The modern English and the legacy French dictionaries are the constituents of this combination.
- **ALL:** The final combination gathers all the samples we collected and annotated, where the multi-linguality is mixed with the different digitisation and aging aspects.

7.4.1 Feature engineering Experiments

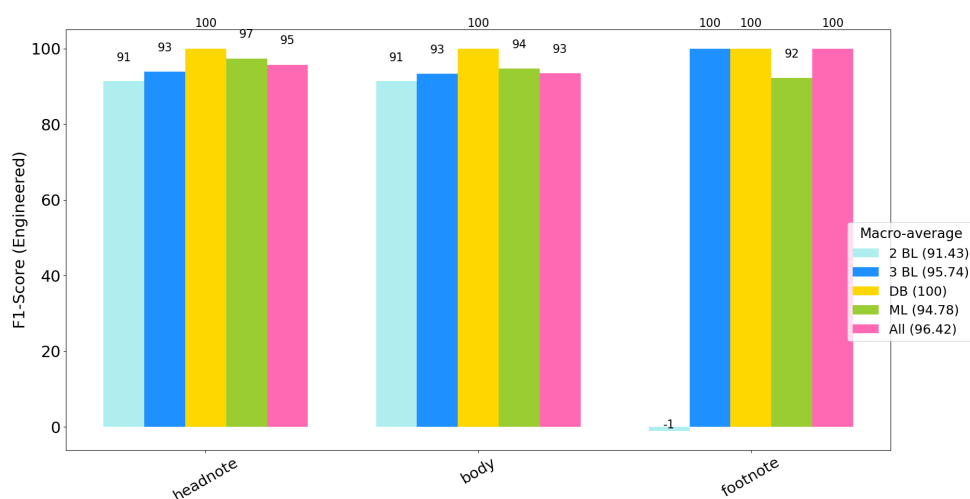
The first set of experiments focuses on investigating the stability of the performance of the templates experimented with separate samples, by training the labelling models on different dictionaries at once.

Experiments

Dictionary Segmentation As illustrated in Figure 7.18, the Bigram templates remain the best variation of this labelling level, despite the relatively good performances of the Unigram templates. None of the combinations of samples seems to harm the performance of the models.



(b) Bigram Templates



(c) Engineered Templates

FIGURE 7.18: Multi-sample Evaluation of the Dictionary Segmentation Model Using two Classes of Templates

Dictionary Body Segmentation Figure 7.19 shows the Bigram templates taking the lead from the Engineered ones experimented with the mono-sample training. Average score is reached when the two samples containing the <dictScrap> label (i.e. 2 BL) are mixed. The model a little confused when only one sample is added to the training (i.e. 3 BL). However, trained with more samples (i.e. ALL) enables good performances to be achieved again.

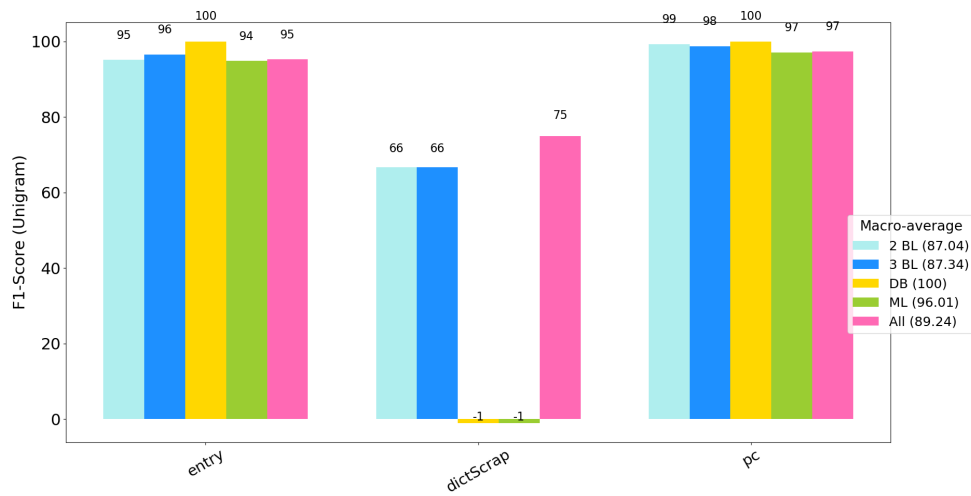
Lexical Entry As for the mono-sample experiments, this model, with text sequences from different sizes and multiple classes to predict, gives its best performances with the Bigram and Engineered templates. The latter outperform the former on the 2 BL, 3 BL and DB combinations. The generalization of the model over all samples is identical for both categories of templates. The most interesting aspect at this lexical parsing level, observed with the Engineered Templates, is that the F1-score obtained with all samples is identical to, and often better than individual scores obtained with the best model from mono-sample training.

Form On the partial combinations of samples, the Bigram and Engineered templates enable the model to reach its best performances with the best results on the ML sample achieved by the Bigram variation. Nevertheless, the evaluation in Figure 7.21 shows identical performances of all templates when training with a combination of all samples. For the best templates, mixing samples boosts the macro average performance reported for single sample training. Such a behaviour is observed for DB and 3 BL. However, that is not always the case as we see a little deterioration in the overall performance for All.

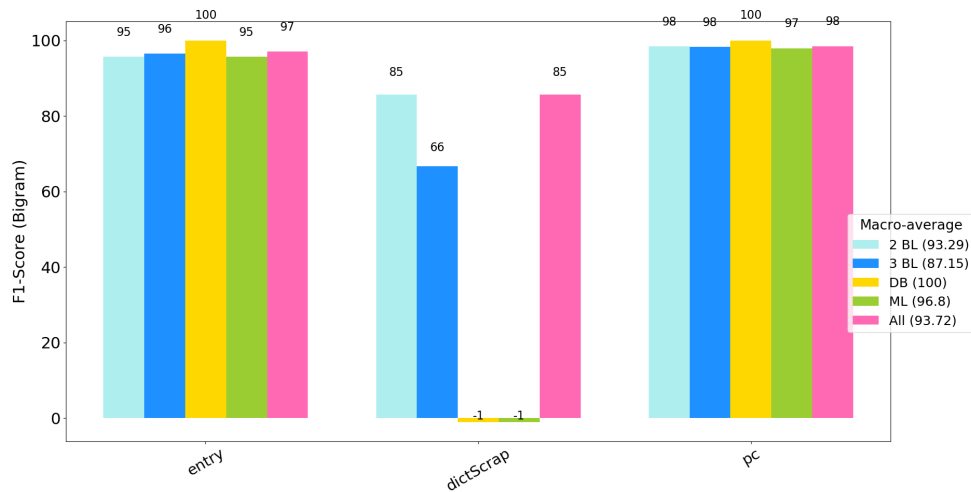
GramGrp As for the mono-sample experiments, the Unigram templates achieve the best scores with almost perfect recognition for all combinations. And compared to the performance on individual samples, the multi-sample training shows only a slight regression. The suitability of simple features for this model is also further confirmed.

Sense Another model, with the limited number of labels but operating on longer text sequences, that shows its preference for simple features, as illustrated in Figure 7.23. The overall performance is also very comparable to the ones achieved by models trained on one sample.

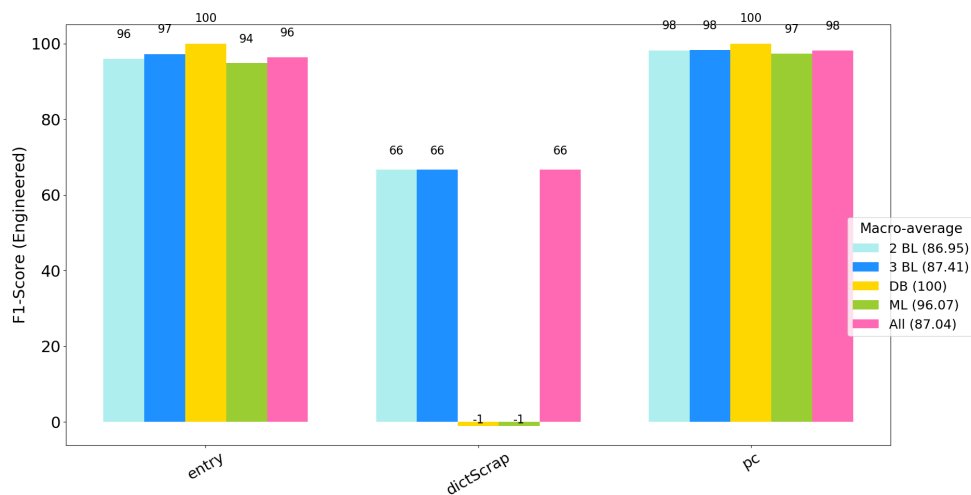
Sub-Sense Macro-average scores reported in Figure 7.24 show a better generalisation capacity for the Engineered templates when the model is trained with all the samples of the pool. Good results are also achieved on other combinations but the Bigram variation has better results. We also notice no deterioration in the performances of the models as a result of combining the training data.



(a) Unigram Templates

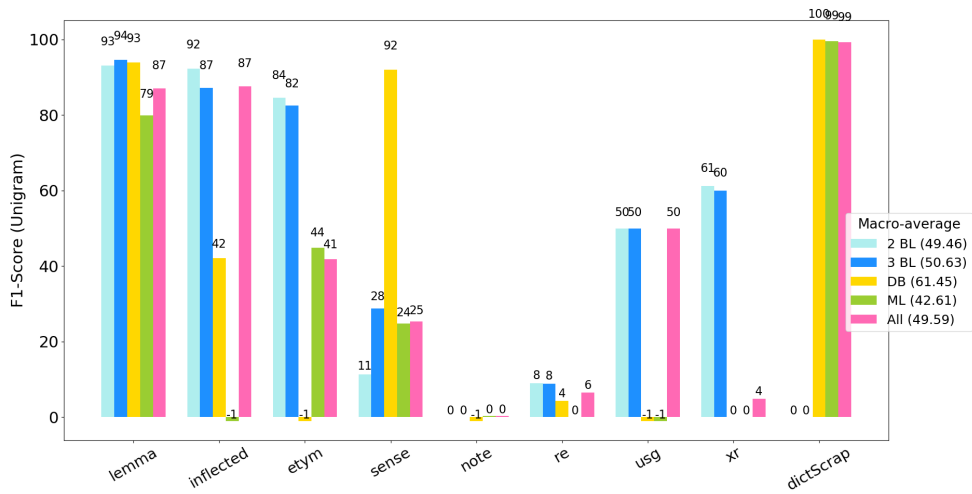


(b) Bigram Templates

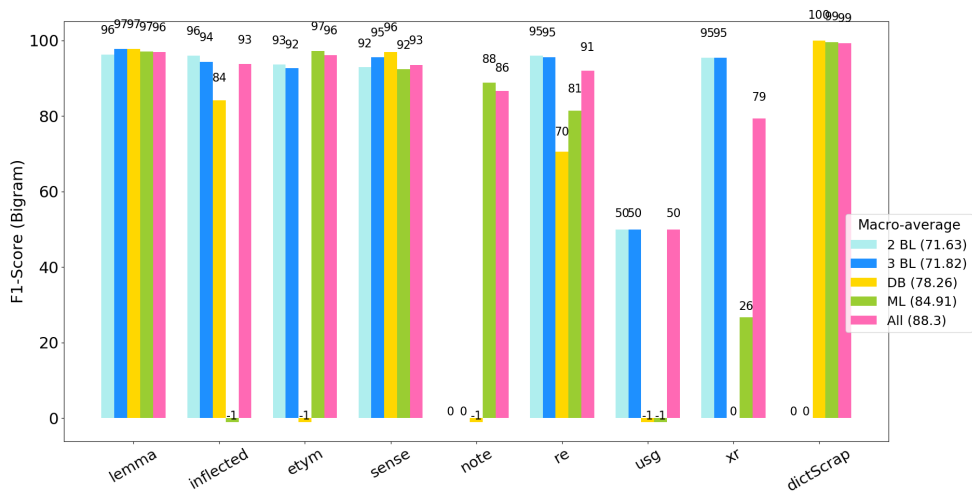


(c) Engineered Templates

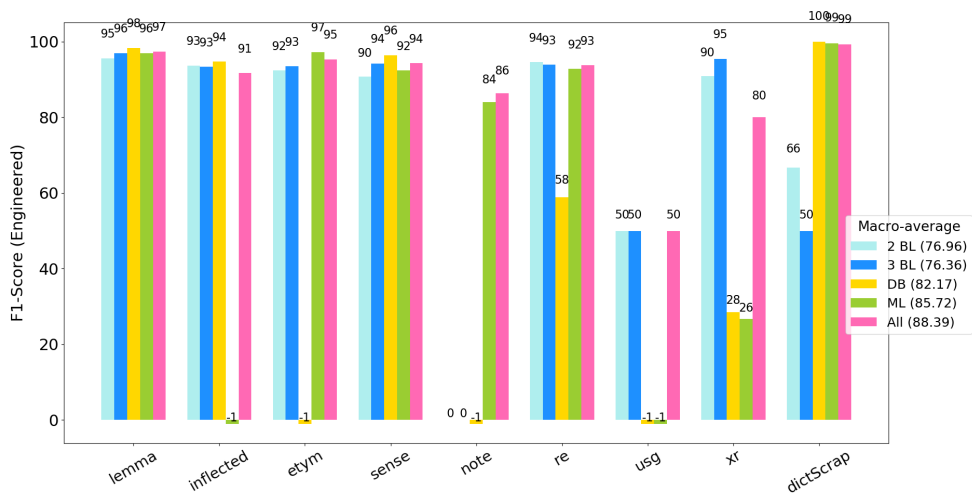
FIGURE 7.19: Multi-sample Evaluation of the Dictionary Body Segmentation Model Using three Classes of Templates



(a) Unigram Templates

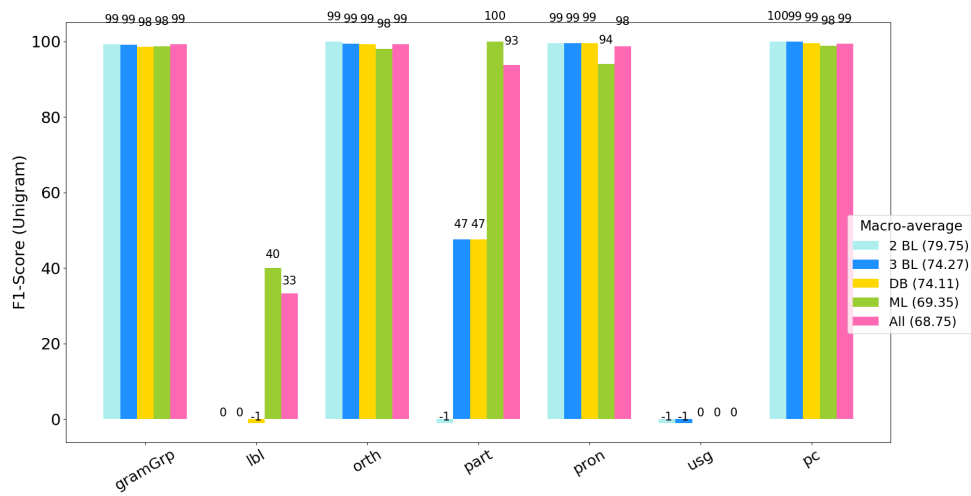


(b) Bigram Templates

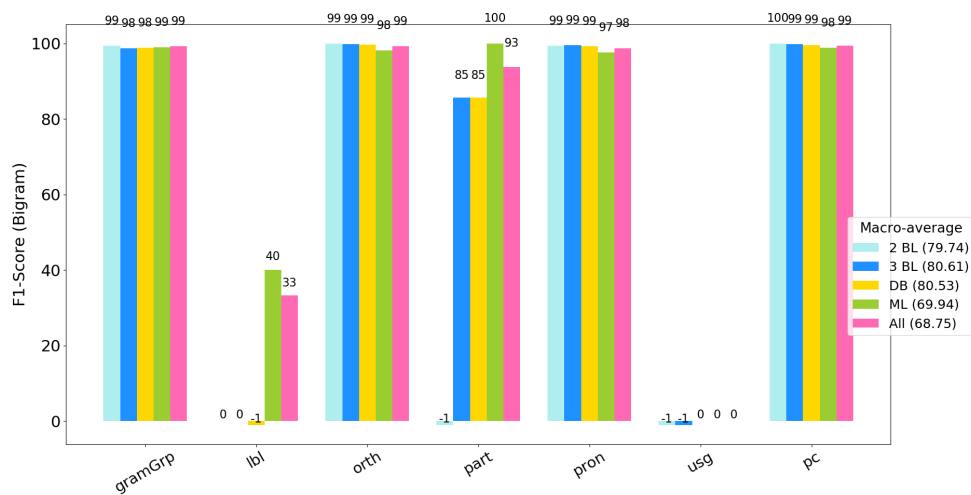


(c) Engineered Templates

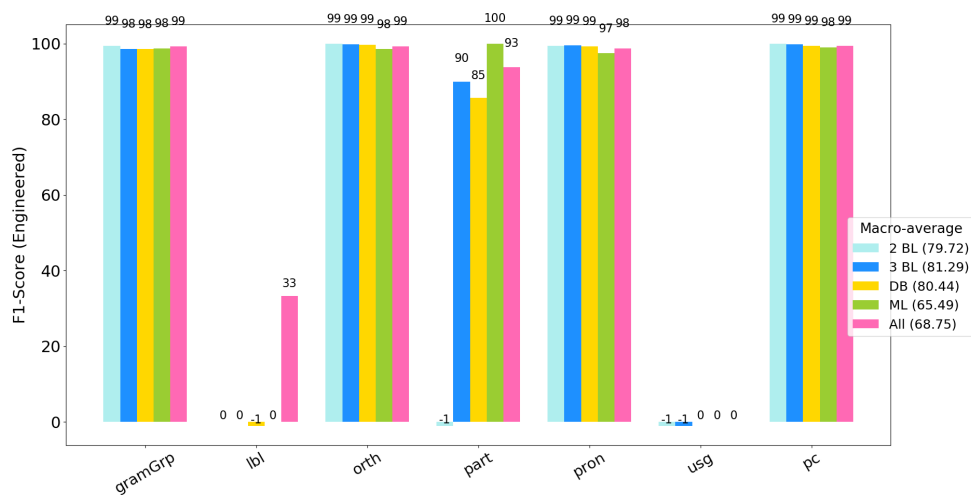
FIGURE 7.20: Multi-sample Evaluation of the Lexical Entry Model Using three Classes of Templates



(a) Unigram Templates

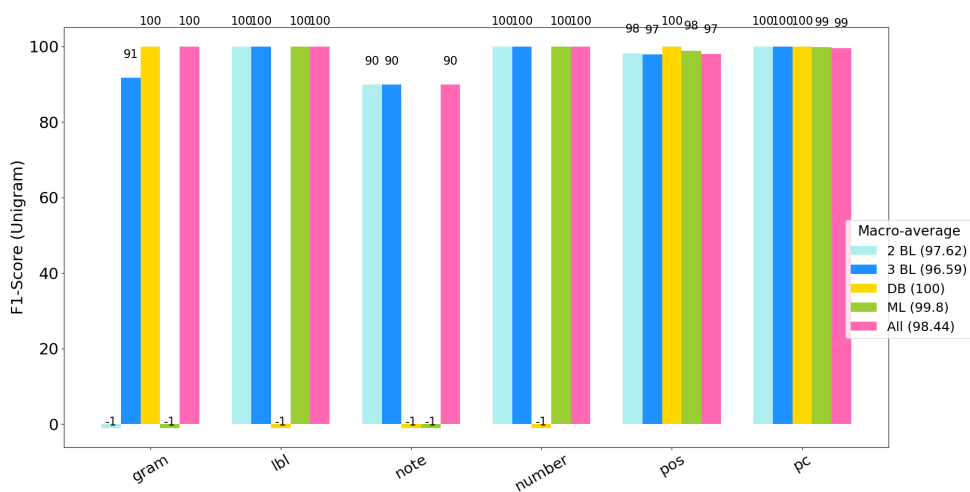


(b) Bigram Templates

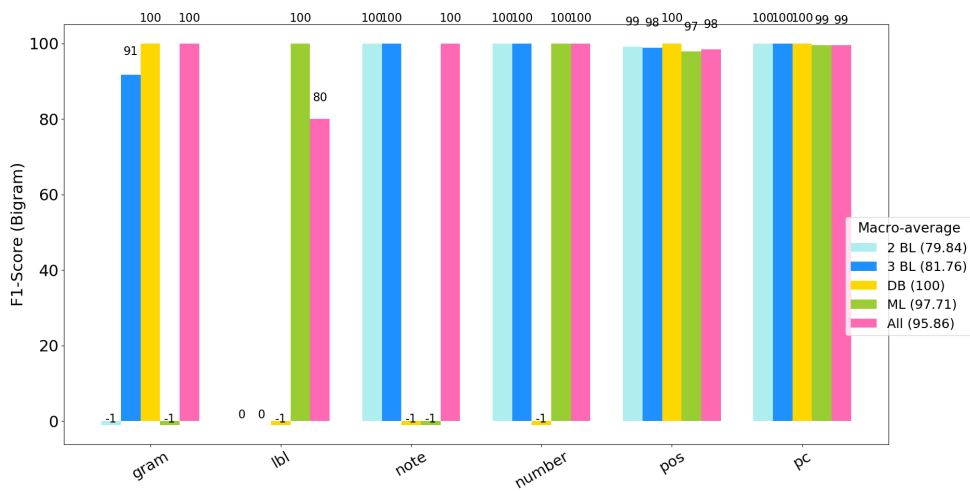


(c) Engineered Templates

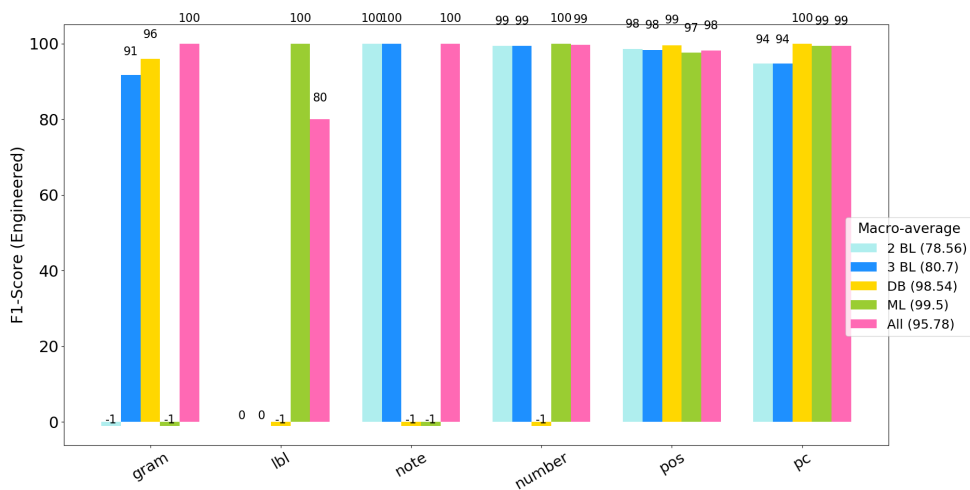
FIGURE 7.21: Multi-sample Evaluation of the Form Model Using three Classes of Templates



(a) Unigram Templates

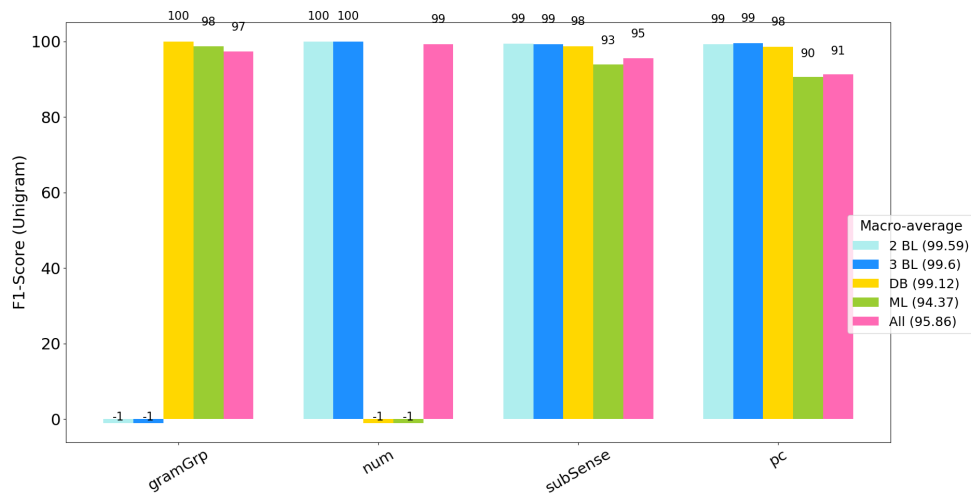


(b) Bigram Templates

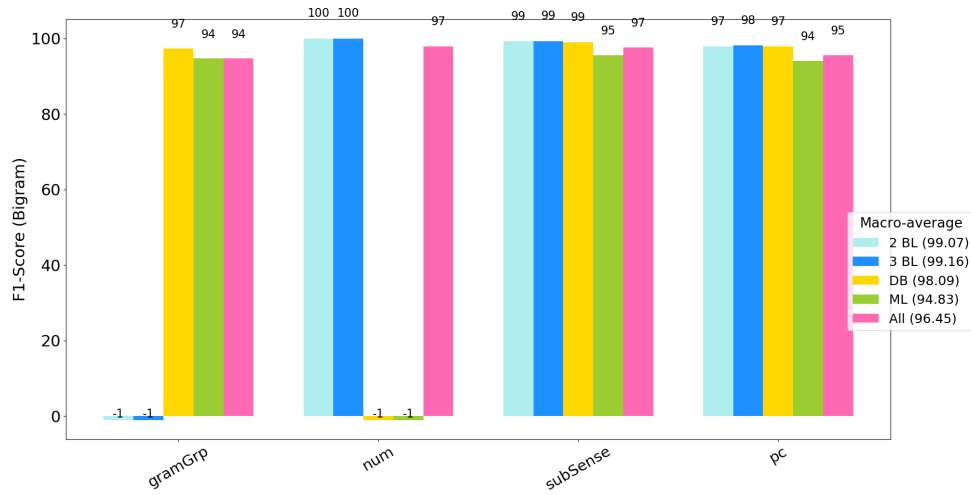


(c) Engineered Templates

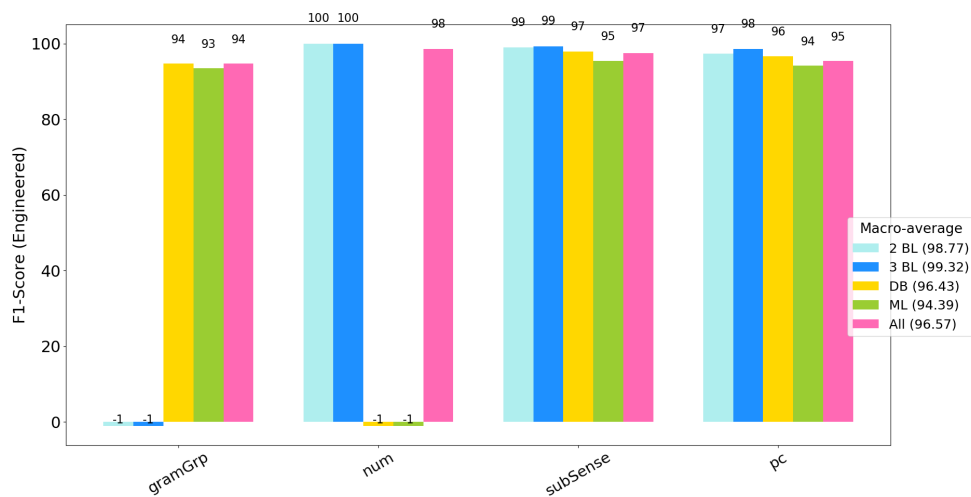
FIGURE 7.22: Multi-sample Evaluation of the GramGrp Model Using three Classes of Templates



(a) Unigram Templates

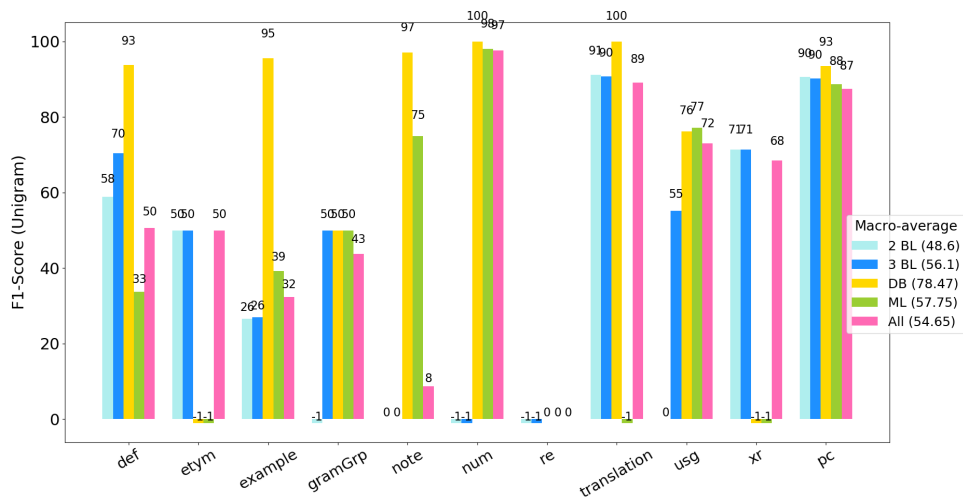


(b) Bigram Templates

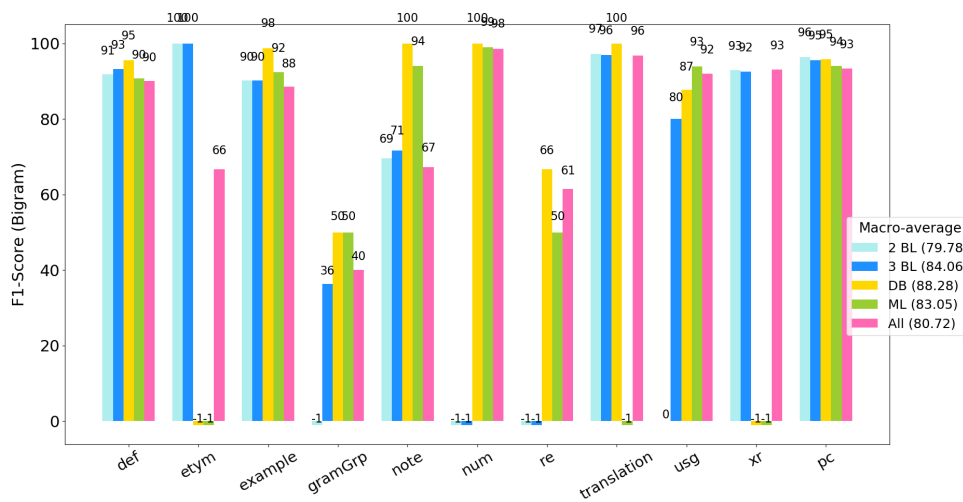


(c) Engineered Templates

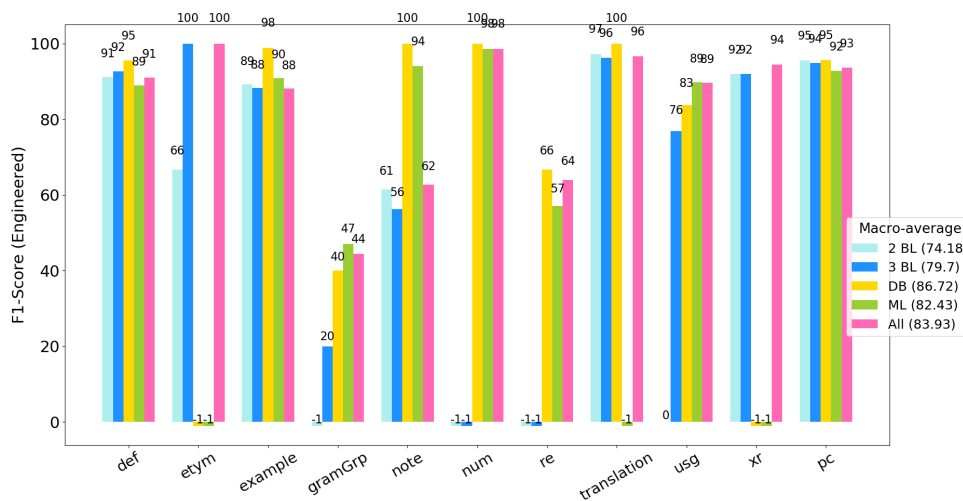
FIGURE 7.23: Multi-sample Evaluation of the Sense Model Using three Classes of Templates



(a) Unigram Templates



(b) Bigram Templates



(c) Engineered Templates

FIGURE 7.24: Multi-sample Evaluation of the Sub-Sense Model Using three Classes of Templates

Discussion

We sum up the results of this series of experiments in Table 7.6.

Model	2 BL	3 BL	DB	ML	All	Best
<i>Dictionary Segmentation</i>	BB	BB	A	BB	BB	BB
<i>Dictionary Body Segmentation</i>	BB	A	A	A	BB	BB
<i>Lexical Entry</i>	EB	EB	EB	BB EB	EB BB	EB
<i>Form</i>	A	BB EB	BB EB	BB U	A	BB
<i>GramGrp</i>	U	U	U BB	U EB	A	U
<i>Sense</i>	A	A	U	A	A	U
<i>Sub-Sense</i>	BB	BB	BB EB	BB EB	EB	B

TABLE 7.6: Summary of the Second Series of Experiments

The main conclusion to be drawn from these experiments is that, in most cases, mixing samples does not harm the learning of a model and can often boost it. Such a combination seems to be healthy for the lexical models to overcome potential over-fitting to data coming from one source. Compared with the analogous table for mono-sample experiments, in Table 7.6 we notice changes of the best templates for each model. The Unigram templates appear to be the most suitable for parsing lexical structures in relatively short text sequences with models having few classes to predict. Engineered templates are the best for the Lexical Entry model over most of the combined samples and remain in tight competition with the Bigram variation. The latter dominates the competition with other templates for the rest of the parsing levels.

7.4.2 Learning Curve

After investigating the impact of mixing samples on the feature engineering choices, we aim at studying the impact on the learning curve of each model for the same combinations used in the previous experiments.

Experiments

The same design of the experiments with mono-sample training was followed to generate the learning curve for the different combinations of these experiments. We simply summed the page numbers of analogous batches and selected the best models observed in the previous series of experiments. The learning curves per combination of samples are presented in Figures 7.25, 7.26, 7.27, 7.28 and 7.29.

Discussion

Given the limited number of samples we used for this experiment, we can not be sure about the origin of some behaviours when certain categories of

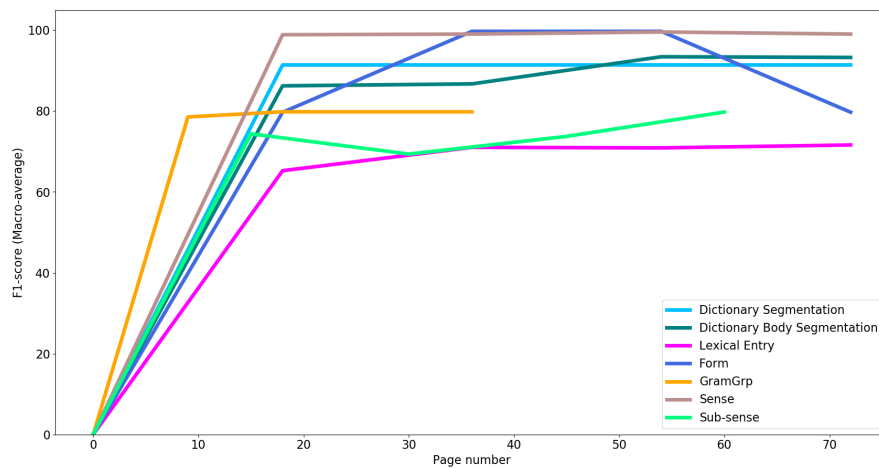


FIGURE 7.25: Learning Curve of the Different Models Given the Number of Training Pages from **2 BL**

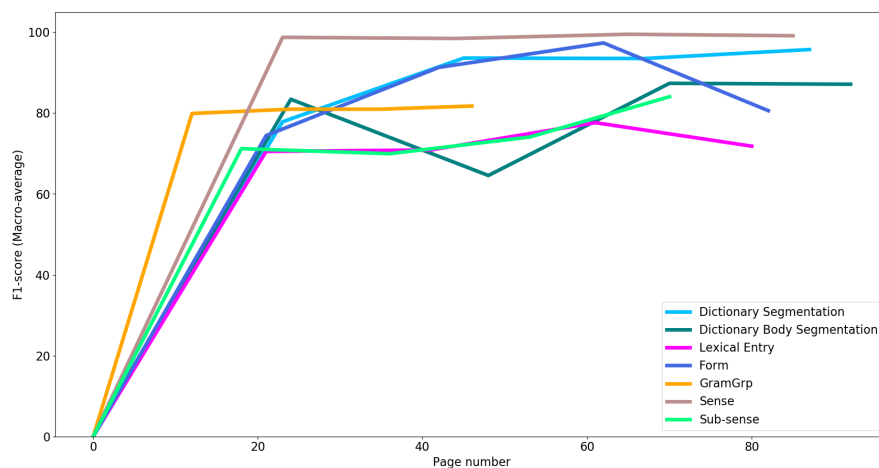


FIGURE 7.26: Learning Curve of the Different Models Given the Number of Training Pages from **3 BL**

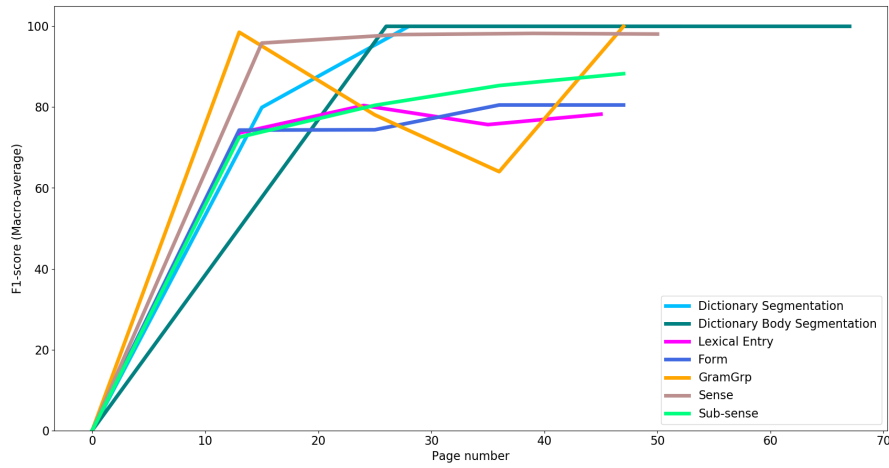


FIGURE 7.27: Learning Curve of the Different Models Given the Number of Training Pages from **DB**

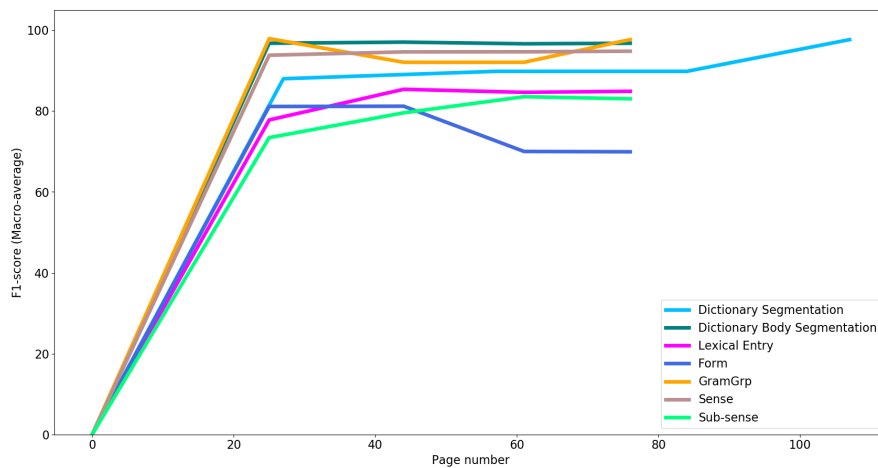


FIGURE 7.28: Learning Curve of the Different Models Given the Number of Training Pages from **ML**

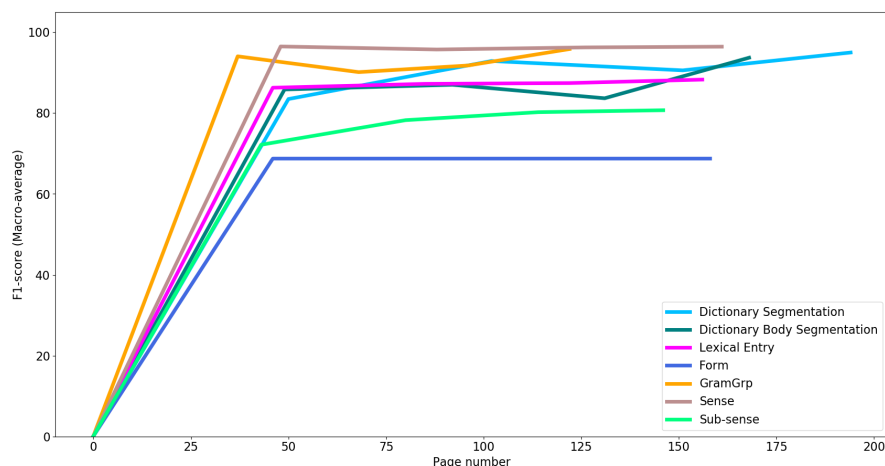


FIGURE 7.29: Learning Curve of the Different Models Given the Number of Training Pages from All

samples are mixed. But we can observe more harmony in the learning curves when:

- a relatively small number of mixed samples comes from the same compilation process, such as the case of **2 BL** compared to **3 BL** or **DB**
- a bigger number of samples are mixed, as observed for **All** compared to **3 BL** and **ML**. Such a combination seems to help the models to have a more abstract prediction over the variety of the training samples.

We can also observe, as for models trained on mono-samples (see Section 7.3.2), that a model has a more stable learning curve when it is operating on long text sequences.

Finally, the convergence of models remains quick as a proportion of pages between 1/4 and 1/3 of the total number of annotated pages allows most of the models to reach performances close to the plateau.

7.5 Experiment Series 3: Testing with Unseen Dictionaries

After studying the behaviour of our models on seen samples, we want to get more insight into the performance of the best models trained with one or more dictionaries and tested with a different sample. In fact, this section sets out to answer a question that we not only asked ourselves, but one that we have been asked several times by the community: “Could the models of GROBID-Dictionaries parse dictionaries that the tool was not initially trained on?”

To answer this question, we carried out the following experiments.

Experiments

We train each model using for each sample the templates that give the best performance, as observed in Tables 7.6 and 7.5, and we change the test dataset to have a sample that the model did not see during the training. We designed 5 experiments where we used samples from previous experiments. Therefore, the previous abbreviations for naming the sample remain valid and we add a dash to differentiate respectively the training from the testing datasets. The training and testing pairs are the following:

- **EEBD-DLF & DLF-EEBD**: the training and the testing is performed with one monolingual dictionary in each dataset then we permute the sample, train and test again.
- **2BL-MxSp & MxSp-2BL**: the same approach applies for this experiment with bilingual dictionaries.
- **ML-BL**: for this experiment, we want to further push the models to their limits by training on monolingual samples and testing on bilingual ones.

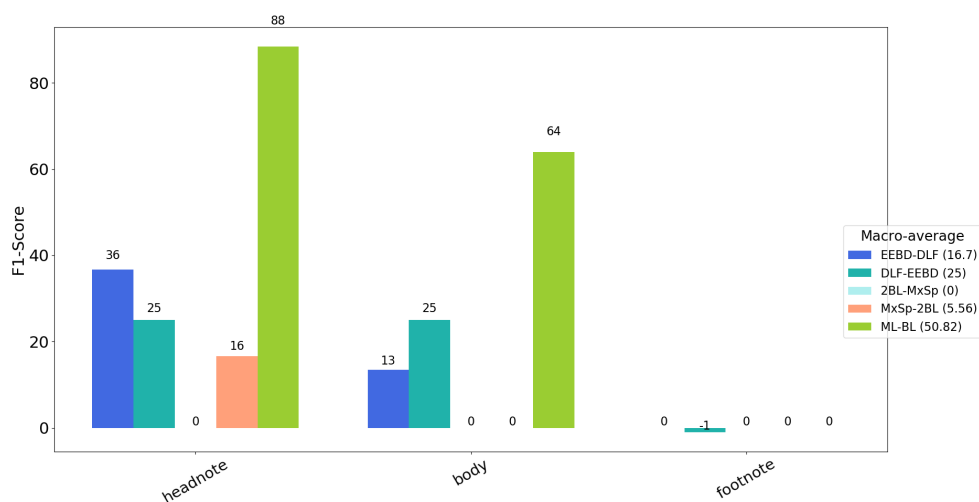


FIGURE 7.30: Evaluation of the Dictionary Segmentation Model Using the Best Templates and Testing with Unseen Sample

Discussion

Figures 7.30, 7.31, 7.32, 7.33, 7.34, 7.35 and 7.36 show very deteriorated performance of the models at different levels of the lexical parsing. The best average performances, around 50% F1-score, are achieved by the pairs **ML-BL** for the **Dictionary Segmentation** model and **DLF-EEBD** for the **GramGrp** model. Some individual good predictions for some labels are also visible such as for <orth> or <subSense> but not for all pairs of samples.

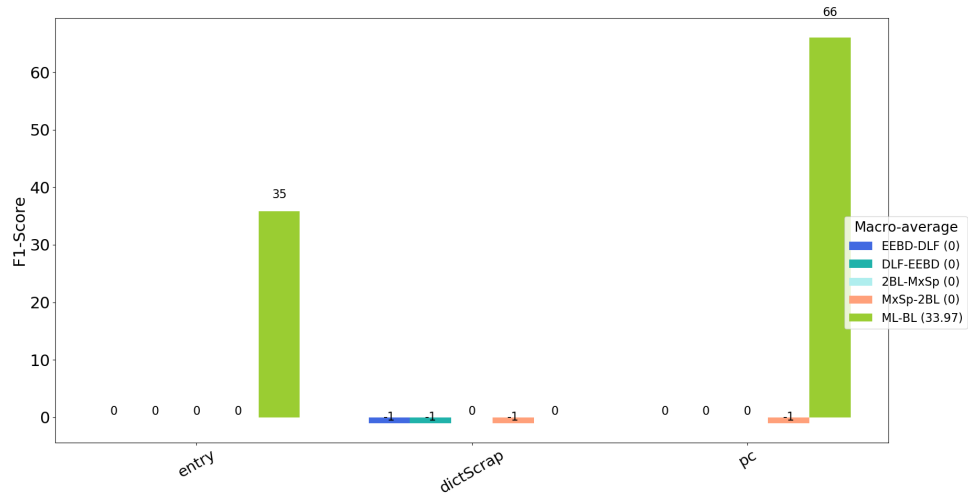


FIGURE 7.31: Evaluation of the Dictionary Body Segmentation Model Using the Best Templates and Testing with Unseen Sample

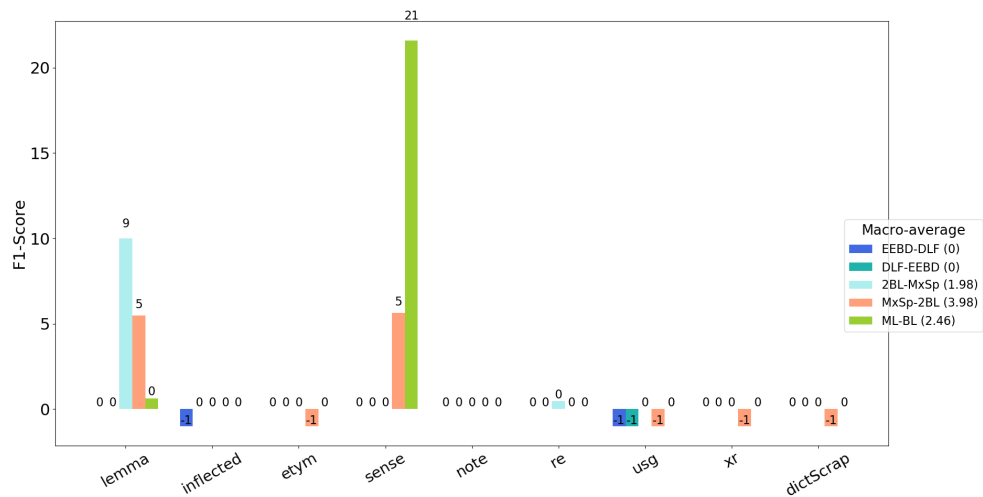


FIGURE 7.32: Evaluation of the Lexical Entry Model Using the Best Templates and Testing with Unseen Sample

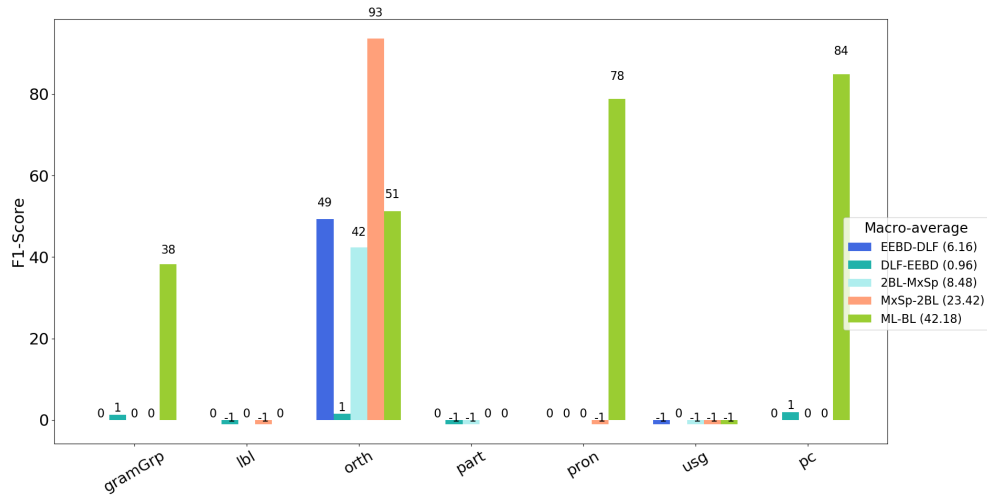


FIGURE 7.33: Evaluation of the Form Model Using the Best Templates and Testing with Unseen Sample

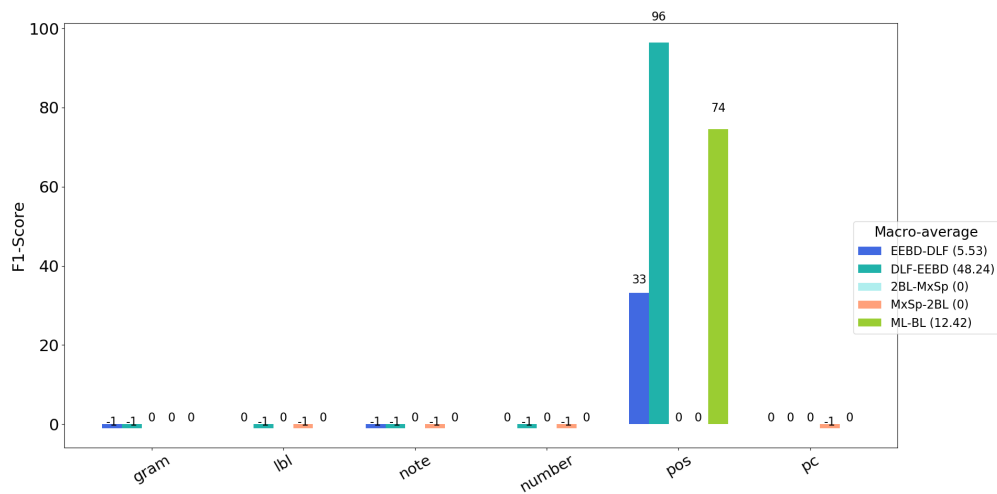


FIGURE 7.34: Evaluation of the GramGrp Model Using the Best Templates and Testing with Unseen Sample

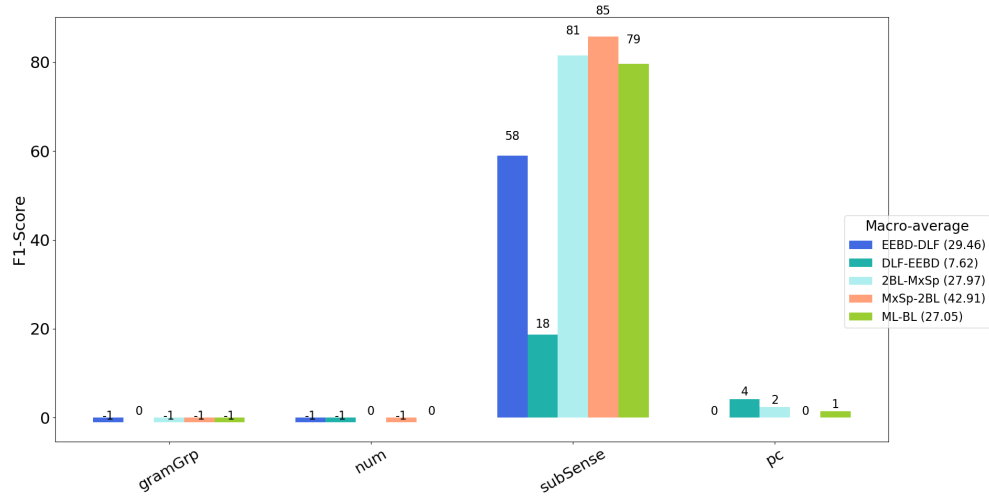


FIGURE 7.35: Evaluation of the Sense Model Using the Best Templates and Testing with Unseen Sample

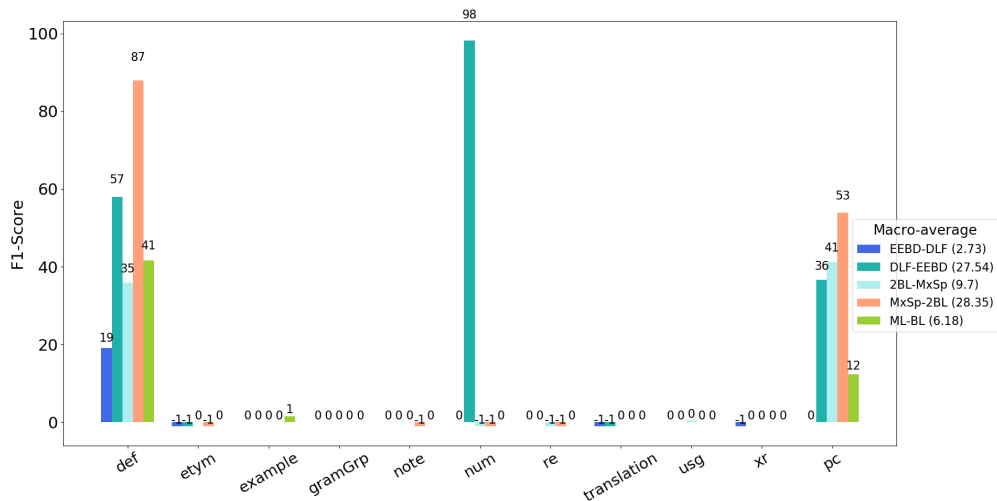


FIGURE 7.36: Evaluation of the Sub-Sense Model Using the Best Templates and Testing with Unseen Sample

This behaviour was somehow expected as theoretically, and often, either the models are trained to predict certain labels which do not exist in the testing sample or the testing sample has labels that the model was not trained to recognise. That is the case of <usg> and <xr> labels of lexical entry models where these classes of structure exist in the **2 BL** sample but the model trained with the **MxSp** sample has no clue about their recognition as they were not among the labels to predict during the training. For the labels which are in common between the training and test sets, the difference in the logical and physical structures seems to be too hard to be contained by the features activated in the training.

The question asked at the beginning of this section seems to be straightforward to answer, given the results of these experiments: in order to achieve accurate predictions, the models need to be trained on the testing sample.

7.6 Scaling up

The scalability aspect of our models was a constant concern during the preparation of this thesis. In this section, we address the issues involved, while putting our work in context with regard to other research projects. We carried out this investigation through two series of experiments: the first addresses scalability by using different categories of data and the second explores the possibility to shift to a new generation of machine-learning techniques.

7.6.1 Experiment Series 4: Beyond dictionaries

Exposing our system at the early stages of its development to scholarly users led us to explore the applicability of the models we designed for similarly structured documents, mainly entry-based ones. Such “side” investigations helped us to gain a better understanding of the behaviour of our parsing models, to find ways to widen their scope and to scale up their performance. Consequently, we were driven to become familiar with common issues in the digital humanities that could be solved by a tool like GROBID-Dictionaries.

Legacy Manuscript Auction Catalogs

Introduction Manuscript Sales Catalogues (MSC)s, called also Manuscript Auction catalogues, are essential documents for Catalogue business as well as for the traceability of the movements of historical documents. An increasing number of such documents is made available to the public and researchers in related humanities fields are keen to acquire methods and tools for extracting structured information from such material. Most of these available catalogues are legacy documents that may date back to the beginning of the 19th century.

The documents in Figure 7.38 represent excerpts from an Encyclopaedic Dictionary and an MSC. One could see how both samples appear similar and that it could be difficult to differentiate between them at first sight. Such an

observation pushed us to investigate, as a first step, the applicability of our lexical encoding schema to this new class of documents.

- 49 **Kourakin** (le prince Alexis B.), frère du précédent, homme d'Etat russe. — Billet aut. sig., en français, à M. Monférand, 1 p. in-8. 2 »
- 50 **Labanoff** (le prince Alex.), célèbre général et écrivain russe, historien de Marie Stuart. — L. a. s., en français, 1835, 1 p. in-4. 3 »
- 51 **Ladislav IV**, roi de Pologne, célèbre par ses succès contre les Russes, époux de Marie de Gonzague. — L. sig., en latin, au cardinal de Montalte; Varsovie, 1645, 1 p. in-f. 8 »
- 52 **Lafayette**, illustre général. — L. a. sig. de ses initiales à M. Masclet; Washington, 13 août 1825, 1 p. 1/4 in-4. Un peu fatiguée. 15 »
 Très-curieuse lettre sur le voyage qu'il fit en Amérique, de 1824 à 1825. « C'est avec de bien tendres regrets que je quitterai cette terre américaine, le bon, grand et heureux peuple des Etats-Unis auquel je suis amalgamé depuis près d'un demi-siècle, et qui vient encore de me combler de ses bontés. J'y ai vu les miracles de l'indépendance, de la liberté, égalité et *self government*; le problème des institutions républicaines a été résolu ici sur une grande échelle et jamais expérience n'a si bien réussi. » Il comptait retourner comme il était venu, sur un paquebot-poste, mais le peuple et le gouvernement en ont disposé autrement. On a donné le nom de *Brandywine* à une superbe frégate qui est chargée de le ramener en France.
- 53 **La Roncière** (Emile-Clément de), fils du général, condamné pour tentative de viol. — L. a. s. aux officiers et élèves de l'école de Saumur; Paris, mai 1836, 3 p. pet. in-4. 10 »
 Très-curieuse lettre toute relative à son procès.
- 54 **Lassalle** (A.-Ch.-L. de), le plus brillant général de cavalerie des guerres de la République et de l'Empire, né à Metz, tué à la bataille de Wagram. — L. a. s. au général Dugua; 1 p. in-f. 10 »
 Superbe lettre sur la campagne d'Egypte. Il profite du départ du général Desaix pour lui donner des nouvelles. Desaix lui laisse le commandement de la colonne qui doit poursuivre Mourad-Bey, et qui se compose de 400 hommes de cavalerie, 4 pièces de canon et 160 dromadaires. Le général Boyer a, dans une petite affaire, tué 10 mameloucks et 40 arabes, etc.

(a) Excerpt from a Manuscripts Auction Catalogue (1889)

ABERDEEN [*aberdin'*], v. d'Ecosse, ch.-l. de comté; port sur la mer du Nord; 170.000 h. Université.

ABERDEEN (G. H. Gordon, *comte d'*), homme d'Etat anglais, né à Edimbourg. Premier ministre en 1852, il conclut avec la France une alliance contre la Russie (1784-1860).

ABER-VRACH, fl. côtier du Finistère (Atlantique); 34 kil. Station marémotrice d'essai.

ABGAR, nom de huit rois d'Edesse, en Mésopotamie (132 av. J.-C.-216 apr.).

ABIA, roi de Juda, fils de Roboam, vainqueur de Jéroboam, roi d'Israël (957-955 av. J.-C.).

ABIDJAN, ch.-l. de la Côte-d'Ivoire (A.-O. F.), sur une vaste lagune navigable; 15.000 h.

ABIMÉLECH [*lèk*], fils de Gédéon. Il devint Juge d'Israël, après avoir fait égorger ses frères; il établit son pouvoir sur Sichem et fut tué au siège de Thèbes, en Palestine (vers 1100 av. J.-C.).

ABIRON, lévite qui fut englouti dans la terre avec Coré et Dathan, tous trois révoltés contre Moïse et Aaron (*Bible*).

(b) Excerpt from an Encyclopedic Section in Petit Larousse Illustré Dictionary (1948)

FIGURE 7.37: Resemblance between MSCs and Encyclopedic Dictionaries

In Khemakhem et al., 2018a, we propose, in collaboration with interested

humanities experts, two encodings for both categories of documents (see Table 7.38).

```

<entry>
  <num>49</num>
  <form type="lemma">
    <surName>Kourakin</surName>
    <addName>(Le prince Alexis B.),</addName>
    <desc> frère du précédent, homme d'Etat russe.</desc>
  </form>
  <sense>
    <pc>-</pc>
    <def>
      <bibl>Billet auto sig., en francais, à M. Monférand, 1 p, in-8.</bibl>
      <num type="price">2 »</num>
    </def>
  </sense>
</entry>
....
....
<entry>
  <num>54</num>
  <form type="lemma">
    <surName>Lassalle</surName>
    <addName>(A.-Ch.-L. de)</addName>,
    <desc>le plus brillant général de cavalerie des
      guerres de la République et de l'Empire, né à Metz, tué à la bataille de
      Wagram</desc>
  </form>
  <sense> .-
    <def>
      <bibl>L. a. s. au général Dugua; 1 p. in- f.</bibl>
      <num type="price">10 »</num>
    </def>
    <note>Superbe lettre sur la campagne d'Egypte. Il profite du départ du général Desaix
      pour lui donner des nouvelles. Desaix lui laisse le commandement de la colonne qui
      doit poursuivre Mourad-Bey, et qui se compose de 400 hommes de cavalerie, 4 pièces
      de canon et 160 dromadaires. Le général Boyer a, dans une petite affaire, tué 10
      mameloucks et 40 arabes, etc.</note>
  </sense>
</entry>

```

FIGURE 7.38: Encoding of an MSC

To reach this target, we use the following from our modelling schema:

- `<name>` is used to encode headwords at the Form level to be then parsed by a future subsequent model to differentiate `<persName>` `<addName>` and `<surName>` constructs.
- `<desc>` element to encode the brief description coming after the headword of an article in both types of documents
- `<def>`, `<bibl>` and `<note>` to model semantic information

We customised the schema of `<entry>` to allow `<num>` to markup the ordering on entries in dictionaries and MSCs. And given the fact that the whole semantic description is related to one sense, the humanities experts in charge of the annotation chose to annotate `<note>` at the **Sense** level and not **Sub-Sense**.

As a second step, we wanted to experiment the ability of our lexical models to learn the recognition of constructs in such documents. In the following section, we describe the experiments we conducted to this end, based on a study focused on parsing MSCs (Rondeau Du Noyer et al., 2019).

Experiments To study the performance of the models on MSCs, we carried out two experiments. The first, named **Cat**, is focused on training and testing the models with different catalogues, whereas the second, named **Cat+Dict**, makes use of the same catalogues as well as all the dictionaries in our pool to train and test the models.

We used 5 models only after we excluded the GramGrp and Sub-Sense models, since the information in the MSCs did not require an activation of these two models. Figures 7.39, 7.40, 7.41, 7.42 and 7.43 present the evaluation of the different models with the Bigram templates, which gave the best results. In these Figures, we also recall the results of the models trained with only the combination of all dictionaries, already reported in previous experiments as **All**. Since the focus of these experiments is on the MSCs, we report the F1-score of individual labels that occur in these documents and exclude the tags which occur only in dictionary samples.

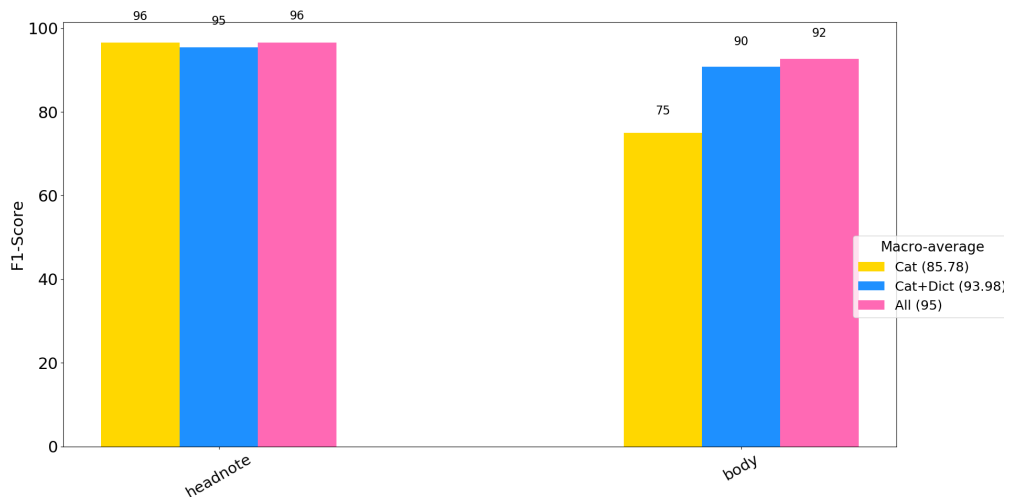


FIGURE 7.39: Evaluation of the Dictionary Segmentation Model Trained and Tested with MSCs and Dictionaries

Discussion The evaluation of the different models shows very accurate prediction results for the **Cat** models with the F1-score exceeding 90%, except for the Dictionary Segmentation model where more annotated data seem to be required. The performance of the models for the recognition of MSCs constructs is very comparable to the results achieved by the models trained to recognise only lexical structures. The combination of samples, **Cat+Dict**, does not harm the learning and the models show a great generalization capacity, except for the Dictionary Body segmentation model where a rarely occurring label <dictScrap> is the origin of a low macro-average score. These experiments confirm the possibility of scaling up our lexical models to a new category of documents by simply applying the modelling schemes in cascade with a minor customisation of the TEI encoding. The possibility of building super-models for parsing similar structures in different documents that

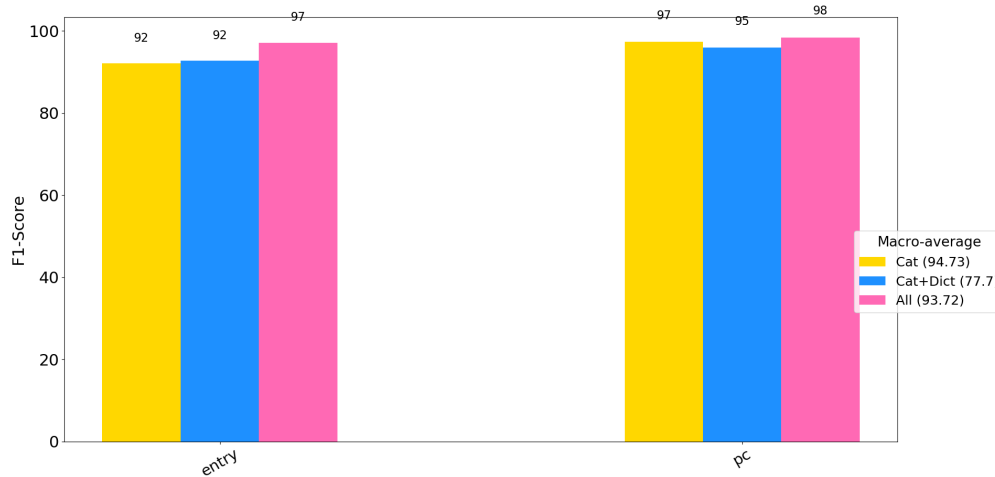


FIGURE 7.40: Evaluation of the Dictionary Body Segmentation Model Trained and Tested with MSCs and Dictionaries

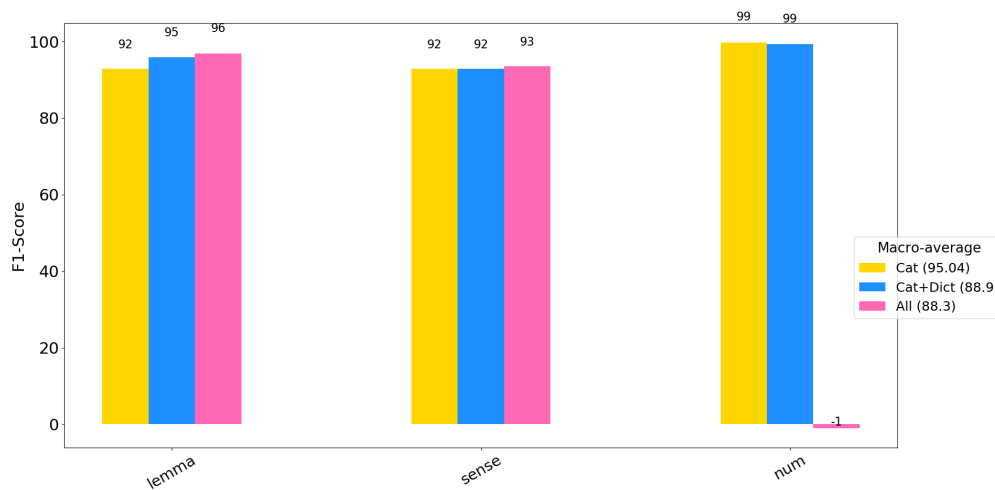


FIGURE 7.41: Evaluation of the Lexical Entry Model Trained and Tested with MSCs and Dictionaries

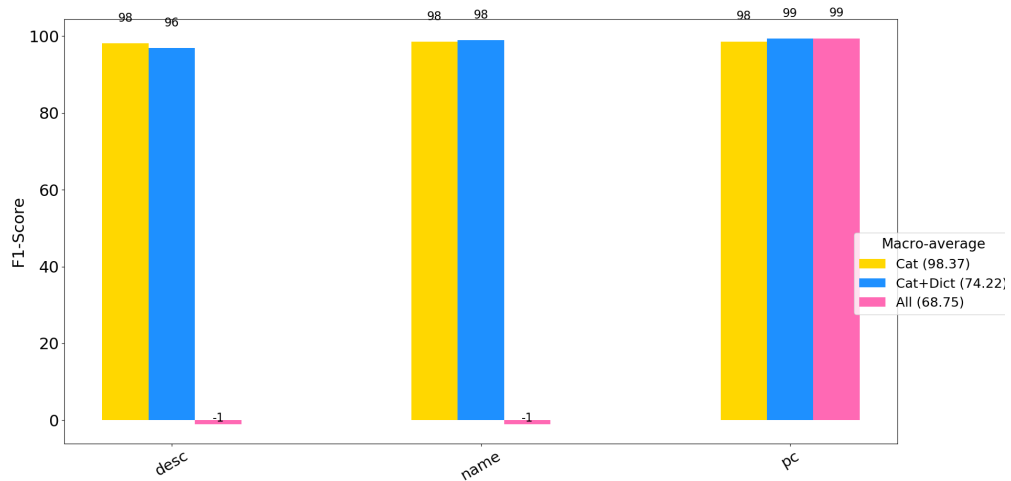


FIGURE 7.42: Evaluation of the Form Model Trained and Tested with MSCs and Dictionaries

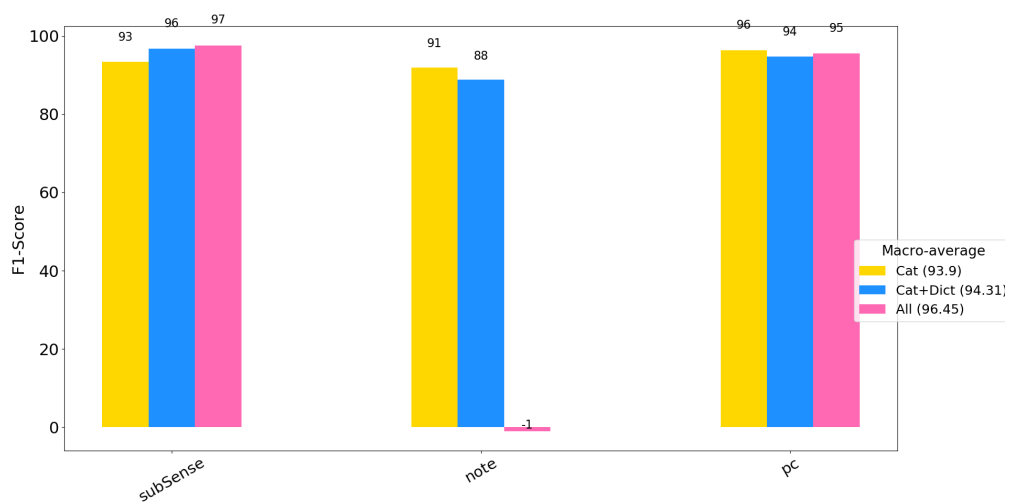


FIGURE 7.43: Evaluation of the Sense Model Trained and Tested with MSCs and Dictionaries

encapsulate lexical and encyclopedic-like content, and are organised in an entry-based layout, is a second aspect that is confirmed by these experiments.

Address Directories

Introduction The lack of adequate information extraction tools and the enhanced usability of the GROBID-Dictionaries and its workflow attracted more researchers in the DH fields to try our models on newer categories of documents. In what follows we present preliminary work that we carried out in this direction, in collaboration with other researchers .

*Time Machine*¹² is a major large-scale project, that was launched after the interest raised by the potential of a seed project (Kaplan, 2015), aiming at analysing and valorising the content of legacy documents for the ultimate purpose of redrawing the historical, social and economical heritage of Europe. Browsing a legacy map representing a geographical snapshot of historical cities is far from being accomplished. The difficulty is firstly due to the lack of structured data allowing a system to map a given address to a throw-back location. Such information is abundantly available in dedicated paper resources, such as legacy address directories. But even digitised, mining the content of these resources remains limited due to the ad-hoc information extraction techniques that are currently employed.

Experiments In the context of joint studies within a local consortium, Paris Time Machine¹³, we explored the possibility of applying GROBID-Dictionaries' models on legacy address directories "Annaires-almanach" of Paris, made available by the French National Library¹⁴. We were struck by the similarities between the structures of dictionaries and address directories, where both resources share a semasiological representation. In fact, such directories could be perceived as encyclopedic resources where locations are described as unique concepts.

As a first step, we tried to find a TEI encoding that converged to the modelling of our lexical parsers. In Table 7.7, we distinguish between two categories of entries. The first is reserved for each entry describing a single occupant in a unique or a shared address. In other terms, to each number in a street, one or many occupants could be assigned and an entry for each one of them. The second category of entries gathers the description blocks of a common street. An entry in this case encapsulates information like the name of the street, its length, neighbouring streets, etc.

In Khemakhem et al., 2018b, we present promising preliminary results of first experiments carried out to extract macro-structures of the entries of these directories. The OCR quality was a serious obstacle for us to pursue more advanced experiments. Nevertheless, such attempts confirmed the potential of our approach and the resulting models to be applied on a larger scale for entry-based documents.

¹²<http://timemachineproject.eu>

¹³<https://paris-timemachine.huma-num.fr/groupe-adresses-et-annuaires/>

¹⁴<https://gallica.bnf.fr/ark:/12148/bpt6k9763088f/f1198.image>


Sample	Encoding
	<pre data-bbox="788 266 1289 512"> <entry> <form>PA S-D E-LA -M ULE (rue du)</form> <sense>(120m de longueur) (II fa lla it I'adresse et le pas de la m ule pour g r a r irsftrement cette rue autrefm a très escar- pée.) -i ' A r r . (Hotel-d e - v ille). Arsenal).&lt;-boul Beaumarchais, 31 et 33, -> place des Vosges, 22 et 23</sense> </entry> </pre> <pre data-bbox="788 562 1289 860"> <entry> <num>1</num> <form> <name> <surname>Couturier</surname> <addName>(J.)</addName> </name> </form>, <sense> <def>photo- graphe</def> </sense> </entry> </pre>

TABLE 7.7: Proposed TEI Encoding of Entries in Address Directories (Didot-Bottin, 1901)

7.6.2 Experiment Series 5: Deep Learning

Introduction

Deep Learning (DL) is a relatively new machine-learning paradigm, based on artificial neural networks, that is constantly gaining more territory in research and industry. In fact, the growing computing capacities, the abundance of data generated daily by the Internet users, and the democratisation of libraries and best practices for data science has resulted in a large movement to deep learn almost any computational task. In the NLP field, Machine Translation (Wu et al., 2016; Artetxe et al., 2017; Cheng, 2019), Part of Speech Tagging (Plank, Søgaard, and Goldberg, 2016; Gui et al., 2017; Meftah and Semmar, 2018; Martin et al., 2019) and Named Entity Recognition (Lample et al., 2016; Peters et al., 2017; Martin et al., 2019; Le et al., 2019) are typical tasks that have benefited from the deep learning race, where the last two tasks represent a certain analogy with the lexical parsing of print dictionaries.

Many readers of this thesis, like many fellow researchers, might well ask the obvious question: why did we not investigate the use of DL and gain inspiration from these analogous tasks to solve the present parsing task? In fact, several reasons lie behind our choice not to favour the investigation of such advanced machine-learning techniques. First, at the outset of our work, we did not have enough understanding of the obstacles that left state-of-the-art methods for parsing dictionaries so limited and did not profit from simpler machine learning techniques. Second, at the time we started our investigations, the understanding of DL architectures was not as developed as it has subsequently become. In addition, deep architectures still required a heavy implementation effort and resources that we were not sure to have. So

we considered that it was a bit risky to rely on such a technology for a very little explored territory. Therefore, we made our choice to build our work upon an evolving implementation, GROBID, which was based on established machine-learning techniques and offered us a minimal framework to start with and adapt.

By the time we had gathered enough knowledge about the requirements of the task we wanted to solve, we discovered several practical limitations of DL models in this sense. One could notice that most of the tasks solved by such models deal with text sequences that do not exceed a few words or sentences. With regard to the specificity of our parsing task, it is still technically impossible to use deep learning models to process batches of text (exceeding 1024 tokens) like the whole text of a dictionary to find macro structures such as the <body> in a dictionary page or <entry>s within a <body> construct. Even parsing separate dictionary articles can encounter this issue for the case of many legacy dictionaries, where the lexical description can take up several columns or pages. Parsing such large structures is vital for the first three models of our architecture, which are key models in the cascading parsing chain. For the rest of the models, we did not find enough annotated datasets that contained real-world data, as we explained at the beginning of this chapter (see Section 7.1).

The simpler technique that we used helped us to quickly explore the parsing complexity for the different lexical models. Moreover, the quick convergence and high accuracy of the models built enabled the annotation of raw data to be speeded up, by using automatic annotation and fewer manual post-corrections. Luckily, our bet on using GROBID did pay off in the end, as in parallel with our work, core functionality in this scholarly parsing system evolved and it became possible to integrate DL models.

In the following section, we present preliminary experiments with deep learning models, when it was technically possible.

Experiments

Through these experiments we aim at showing the possibility to integrate a new generation of DL models within our system. We also want to get insight into preliminary results of an example of a more advanced machine learning technique compared to the classic CRF we used to implement our sequence labelling models.

We used *DeLFT 2018–2020*, an advanced library implementing Deep Learning Architectures, which benefits from a native integration within GROBID. From this framework, we used an implementation of the architecture proposed by Lample et al., 2016 which relies on a BidLSTM-CRF model with glove-840B word embeddings. For these experiments, we skip the evaluation of the Dictionary Segmentation and Dictionary Body Segmentation models, given the technical limitation of such deep architectures to handle the corresponding long text segments, and we present the results of the other models. To train and test the new labelers, we chose the EEED sample given its mid-complexity and its fitness of the used word embeddings to the language and

its modernity. In Figures 7.44, 7.45, 7.46, 7.47 and 7.48, we report the evaluation of the DL and Wapiti models trained and tested with the same English dictionary.

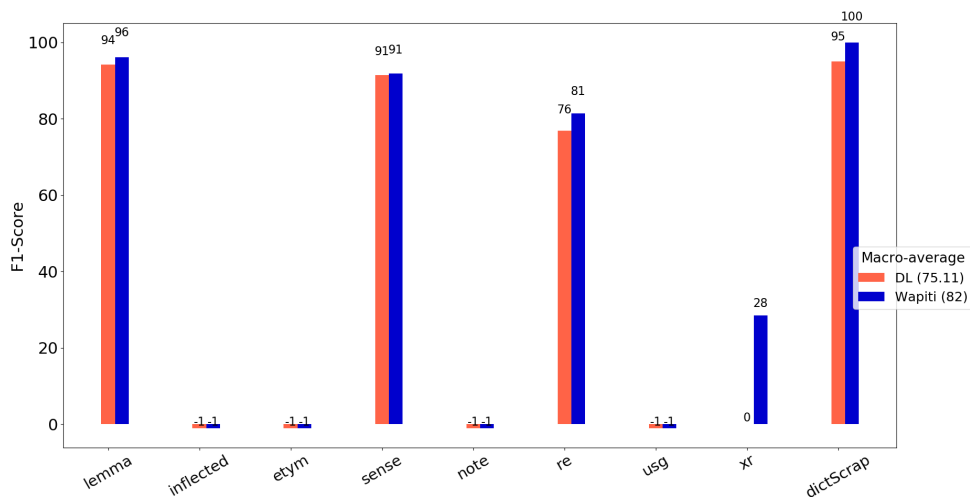


FIGURE 7.44: Evaluation of the Lexical Entry Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labelers

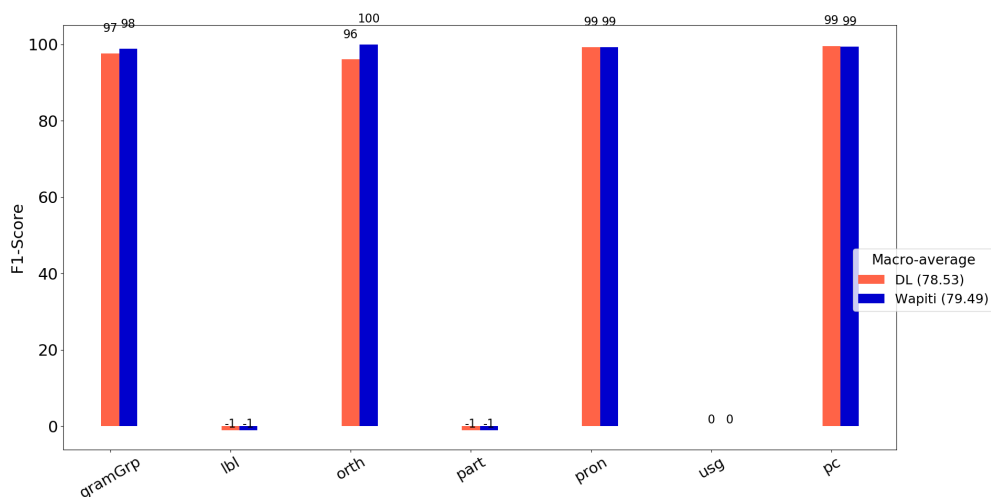


FIGURE 7.45: Evaluation of the Form Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labelers

The results of these experiments show comparable performances with identical scores for one model, GramGrp, and better results achieved by Wapiti for the rest of the models. More data and different tuning of the deep models need to be further experimented in order to improve the labelling capacities of the models at a larger scale and to overcome the limitations of classic machine learning models. Such a study goes beyond the goals of this

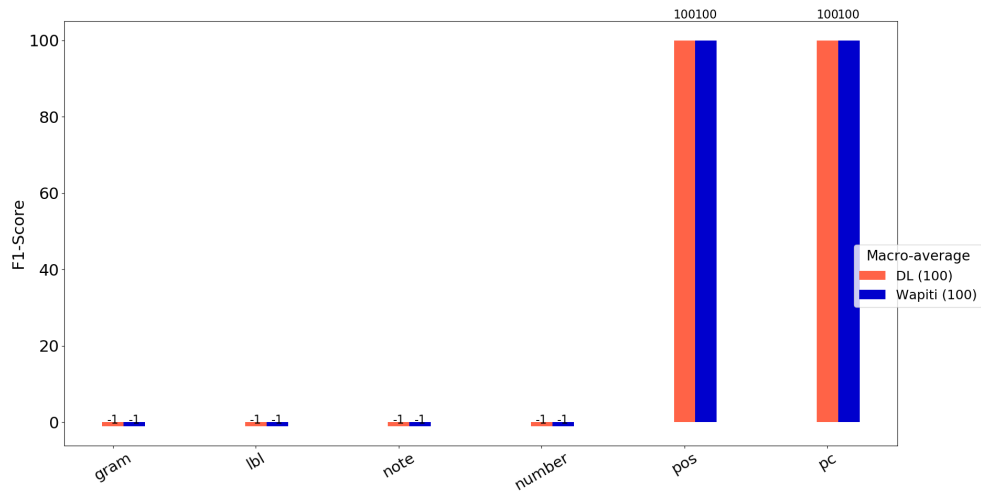


FIGURE 7.46: Evaluation of the GramGrp Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labels

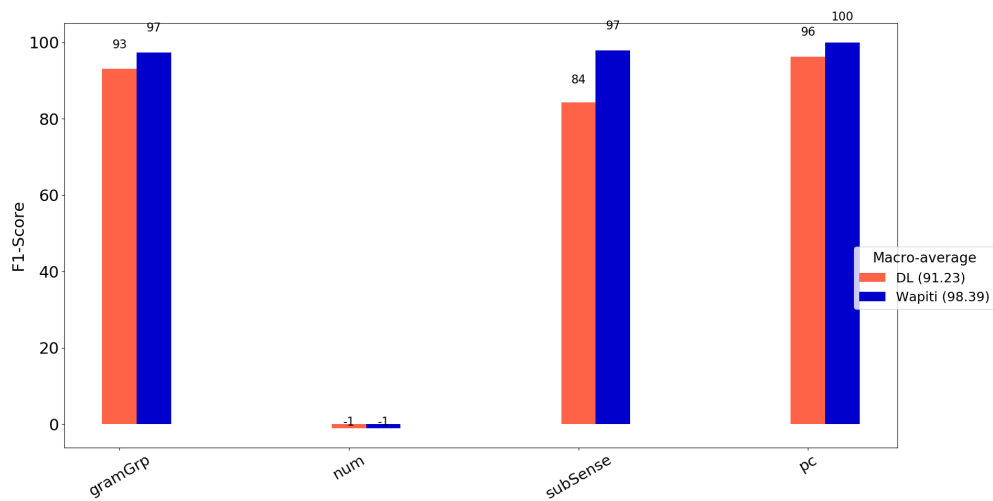


FIGURE 7.47: Evaluation of the Sense Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labels

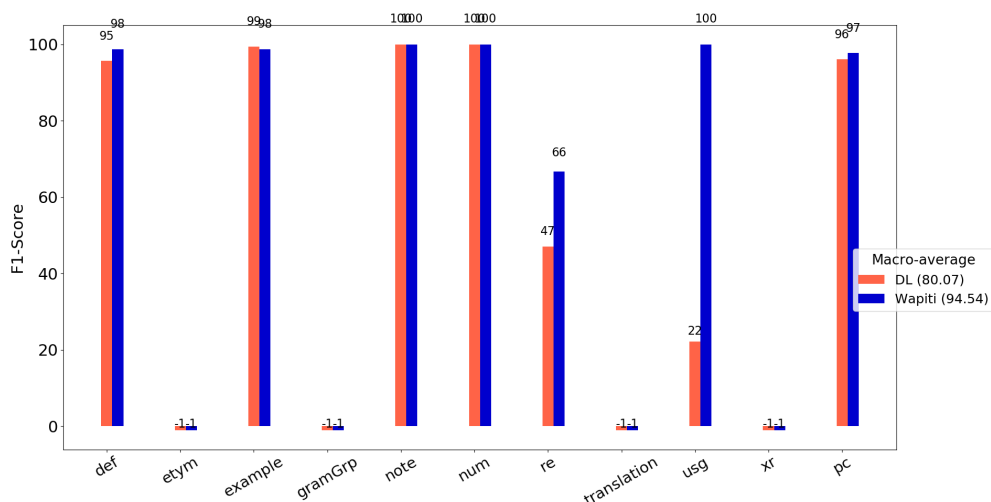


FIGURE 7.48: Evaluation of the Sub-Sense Model Trained and Tested with EEBD Using Deep Learning and Wapiti Labelers

thesis. Nevertheless, we have already started to explore in this direction in the context of new research projects that build upon the results of the current work.

7.7 Chapter Summary

This chapter breaks down the complexity of parsing print dictionaries from an empirical perspective. We also presented the different factors and features that impact on the performance of our lexical models and showed different scaling up alternatives.

The introduction of the machine-learning setup, with regard to the specificity of the lexical parsing task, gave an overview of aspects and details that should be taken into consideration to properly conduct machine-learning experiments. Through multiple extensive experiments, we studied the impact of feature engineering and observed how it can vary according to the lexical parsing level and the sample. The CRF models implemented in GROBID-Dictionaries showed the ability to adapt and parse all the dictionaries we presented, to varying degrees. The positive combination of samples was also an aspect that we explored along with the limitation of trained models to differentiate structures in unseen samples. Through drawing the learning curves, we also demonstrated that the technique we have used does not require a large amount of annotated data to achieve accurate parsing results. More side exploratory investigations on the application of our architecture were successful and we managed by simply applying our annotation scheme to enable the parsing of new categories of entry-based documents. Scaling up the technique to more sophisticated models was also addressed and preliminary experiments showed the possibility of integrating DL models at different levels of our architecture.

Chapter 8

Summary & Perspectives

8.1 Summary

Parsing print dictionaries using lexical models that are able to dismantle the multi-fold complexity of such resources and the generation of standardised output enabling a maximum exchange, are two tasks that we addressed in this thesis. We presented a novel approach relying on standard-based lexical models that build on the leading encoding standards and extends the hitherto modest state-of-the-art of parsing techniques.

A standard-based lexical model, as presented in this thesis, can be assimilated as an automatic labeller:

- that uses a supervised machine-learning technique
- based on typographic and textual features
- to learn the recognition of structures in text sequences
- and serialises the recognised constructs into a standardised resource

Our initial goals were to:

- Understand why parsing dictionaries did not benefit from the great advances in machine-learning techniques
- Find an encoding that can generalise over most of dictionaries
- Build models that can fit this encoding and accelerate the structuring of print dictionaries

We tackled the analysis of the challenges for such a parsing task by presenting the specificity of digital-born and digitised documents, and the varieties in the logical and physical structures of print dictionaries. We also presented related work, which, in general, is limited and experimental, and our inspiration from an existing successful implementation in an infrastructure for parsing bibliographic information.

We have accordingly implemented an end-to-end system, called GROBID-Dictionaries, which relies on cascading machine-learning models to orchestrate the analysis and the extraction of lexical structures from an input dictionary. We presented the different challenges that we encountered to build our models, which required both engineering and lexicographic skills. For

the engineering part, we managed to adapt core functionalities of GROBID, the implementation of our above-mentioned inspiration, and build on top of them a platform for training multi-level CRF models for parsing dictionaries. For the lexicographic part, we succeeded in enhancing the usability of the system for its use in the humanities. In fact, we have lightened the setup and the annotation processes and exposed the enhanced environment to real users who, consequently, provided us with valuable feedback to shape the design of our models and to spot real expert issues.

Building the parsing infrastructure went hand in hand with the definition of the final standardised output. Actually, finding the match between the final output and the designed cascading models was the guiding line of this thesis and the most challenging part, given the implementation constraints and the fact that the final output relies on the outcome of ongoing standardisation discussions.

To define the final output, we studied the most widely-used standards and formats for modelling lexical resources, namely TEI, LMF and OntoLex-Lemon, and we discussed their strengths as well as their limitations. Based on that, we presented our contribution in the shaping of two emerging schemes, TEI-Lex-0 and the new LMF, that try to unify the lexical modelling practices by following two different approaches. Our standardised output was massively inspired from these evolving schemes and this can be observed in the serialisation model we have presented in this thesis. We managed to make our output converge to a model for serialising the lexical entry macrostructure that satisfies most of the modelling of TEI-Lex-0 and LMF models. For more granular lexical parsing, we have anticipated based on the samples we encountered.

After building the architecture of lexical models, we aimed at studying their performance and behaviour at each parsing level. To this end, we needed annotated data that suited the design and the encoding of our models. Such a requirement was not easy to satisfy given the challenging aspects of lexical data annotation and real-world data. Thanks to the efficiency of the technique implemented for our models and the enhanced usability of the system, we managed to semi-automatically create sliver-standard data carrying suitable annotations.

Through an extensive experimentation, we analysed the behaviour of our CRF models as well as ways to improve their performance, chiefly through feature engineering. We also showed the impact of combining samples for the training on the generalisation capacities of such sequence labellers, that could be maintained and often improved. In addition, we have shown empirically the limits of these models in the parsing of samples that were not used in the training.

Our cascading approach showed a great ability to adapt for parsing new categories of documents. The flexibility of the models has been tested with non-dictionary resources that share certain physical and logical aspects with

print dictionaries. Simply through a proper annotation of the new samples, the implemented models demonstrated a significant genericity for parsing and encoding dictionaries and entry-based documents, mainly with encyclopaedic content. Finally, the implementation of our approach showed enough flexibility to easily integrate a more advanced category of machine-learning models, in particular DL technology. Such a scalability can unlock and speed up experiments aiming at advancing research in such a direction.

From a standardisation perspective, this thesis gives ideas and practical solutions for structuring an increasing number of raw lexical and entry-based documents into exchangeable resources. Such research work represents a concrete use case where related initiatives are exposed to real extraction scenarios to confirm choices and to open up discussion about the applicability of some lexical modelling decisions.

8.2 Perspectives

The work presented in this thesis has the potential to pave the way for massive information extraction from a wide range of digitised documents, mainly dictionaries and other entry-based documents. What we have so far achieved can be extended in several ways.

Besides carrying on the work on implementing the missing models in GROBID-Dictionaries, in parallel with the refinement of the final encoding, scaling up the performance of the built architecture can be investigated by either performing more feature engineering or integrating more DL models on more data. Regarding data, after major obstacles were clarified and overcome, this thesis can be used to develop guidelines and a protocol for large-scale annotation of more diverse samples. Given the rapid convergence of our models, the generation of training data can be exponentially accelerated to find larger sets of data to enable meaningful investigations of advanced machine-learning techniques. For that, user interaction for the training functionalities could be further improved by making them more accessible to a greater number of users with more enhanced usability (e.g., implementing a user interface). Given the very promising results on Catalogs, we have already started to concretise ideas in such research directions by launching GROBID-Cat¹ where we plan to make an in-depth customization of the architecture and encoding for more fine grained extraction. Such collaboration can unlock research in related fields by making more structured legacy documents available.

Being compatible with the GROBID-family tools, our models can be exploited to build new tools based on models coming from different systems. A first attempt in this direction to parse large bibliographic collections (see Figure 8.1) was successful by combining models of GROBID-Dictionaries and GROBID (Lindemann, Khemakhem, and Romary, 2018). We also plan to further enrich the output of GROBID-Dictionaries by applying models from

¹<https://github.com/MedKhem/grobid-cat>

Named Entity Recognition and Disambiguation system (Foppiano and Romary, 2018). Another avenue that is worth exploring in order to enrich the output of our system is the application of tools for downstream tasks.

Titelstrecke V

- 21638 V.:** Wozu bedarf ein Deutscher eines deutschen Wörterbuchs? In: *Illustrierte Zeitung* Nr. 1220 v. 17. 11. 1866, 327. [Presstext].
- 21639 v. B.:** Duden – Das große Wörterbuch der deutschen Sprache in 6 Bänden. Über 500 000 Stichwörter und Definitionen: von *A* bis *Zytotoxizität*. Bibliographisches Institut, Mannheim. Dudenverlag. Preis je Band: 58 DM. In: *Feld und Wald* (Essen) v. 28. 1. 1983. [Presstext].
- 21640 Vacalopoulou, Anna:** The 'Bank of English' and the 'British National Corpus': Evaluation form the Lexicographer's Perspective. Diss. [masch.] for M. A. in Lexicography. Univ. of Exeter. Exeter 1996.
- 21641 Vacek, Jaroslav:** Russian Dictionaries of Indian Languages – a Note. In: *Archív Orientální* 49. 1981, 66.
- 21642 Vacek, Jaroslav:** A Computer Aided Dictionary of Tamil. In: *Archív Orientální* 61. 1993, 317–319. [Rezensionsaufsatz].
Dazu Abstract in: *LLBA* 29. 1995, 2449.
- 21643 Vachková, Marie:** Bilinguale Lexikographie und Wortbildung. Das adjektivische Suffix *-haft* in kontrastiver Sicht. In: *Germanistica Pragensia* 14. 1997, 143–150.
- 21644 Vachková, Marie:** Das große deutsch-tschechische Wörterbuchprojekt im Kontext der zeitgenössischen deutsch-tschechischen Übersetzungsllexikographie. In: *Kunzmann-Müller/Zielinski* (Hrsg.) [...] 2002↑, 112–118.
- 21645 Vachková, Marie:** Wortbildung und zweisprachiges Wörterbuch. In: *Das Wort in Text und Wörterbuch* [...] 2002↑, 119–126.
- 21646 Vachková, Marie:** Der deutsch-tschechische Sprachvergleich im deutsch-tschechischen Übersetzungswörterbuch. In: *Brücken. Germanistisches Jahrbuch Tschechien-Slowakei* 11. 2003, 251–263.
- 21647 Vaciago, Paolo:** Old English Glosses to Latin Texts: A Bibliographical Handlist. In: *Medioevo e Rinascimento. Annuario del Dipartimento di Studi sul Medioevo e il Rinascimento dell' Università di Firenze* 7. NS. 4 1993, 1–67. [Bibliographie].
- 21648 Vaciago, Paolo:** Towards a Corpus of Carolingian Biblical Glossaries. A Research in Progress Report. In: *Les manuscrits des lexiques et glossaires* [...] 1996↑, 127–144.
- 21649 Vagianou, Maria:** Cartoon Humor in Children's Dictionaries and Reference Books and Its Effect on Learning and Vocabulary Retention. Diss. [masch.] for M. A. in Lexicography. Univ. of Exeter. Exeter 1996.

Bereitgestellt von | Universitätsbibliothek Hildesheim
Angemeldet
Heruntergeladen am | 04.05.18 14:03

FIGURE 8.1: Excerpt from the Bibliography Collection (Wiegand, 2014) used for Experimenting the Combination of GRO-BID and GROBID-Dictionaries Models

Appendix A

Descriptive Vectors and Feature Templates

A.1 Descriptive Vectors

We used two categories of descriptive vectors: line based and token based

A.1.1 Descriptive Vectors for Dictionary Segmentation Model (a.k.a. GROBID's First Model)

These are line based vectors, where each vector is described by:

0. First Token
1. Second token
2. Lower-cased first token
3. Prefix 1 character
4. Prefix 2 characters
5. Prefix 3 characters
6. Prefix 4 characters
7. Block information (e.g. BLOCKSTART, BLOCKIN, BLOCKEND)
8. Page information (e.g. PAGESTART, PAGEIN)
9. Font status (e.g. NEWFONT, SAMEFONT)
10. Font size information (e.g. HIGHERFONT, SAMEFONTSIZE, LOWER-FONT)
11. Is Bold (e.g. 0 when false, 1 when true)
12. Is Italic (e.g. 0 when false, 1 when true)
13. Capitalisation (e.g. INITCAP, NOCAP, ALLCAP)
14. Digit information (e.g. NODIGIT, ALLDIGIT, CONTAINSDIGITS)

15. Character information (e.g 0 when false, 1 when true)
16. Is a Proper Name (e.g. 0 when false, 1 when true)
17. Is a Common Name (e.g. 0 when false, 1 when true)
18. Is a First Name (e.g. 0 when false, 1 when true)
19. Is a Year (e.g. 0 when false, 1 when true)
20. Is a Month (e.g. 0 when false, 1 when true)
21. Is a Email (e.g. 0 when false, 1 when true)
22. Is a HTTP (e.g. 0 when false, 1 when true)
23. Punctuation information (in case the token is a punctuation, NO otherwise)
24. Relative Document Position
25. Relative page position coordinate
26. String Profile
27. Current line length
28. Is a bitmap around (e.g. 0 when false, 1 when true)
29. Is a vector around (e.g. 0 when false, 1 when true)
30. Is a repetitive pattern (e.g. 0 when false, 1 when true)
31. Is a first repetitive pattern (e.g. 0 when false, 1 when true)
32. Is the block is in the page main area (e.g. 0 when false, 1 when true)

A.1.2 Descriptive Vectors for Dictionary Body Segmentation Model Onward

These are token based vectors, where each vector is described by:

0. Token
1. Lower-cased token
2. Prefix 1 character
3. Prefix 2 characters
4. Prefix 3 characters
5. Prefix 4 characters
6. Suffix 1 character

7. Suffix 2 characters
8. Suffix 3 characters
9. Suffix 4 characters
10. Font size information (e.g. HIGHERFONT, SAMEFONTSIZE, LOWERFONT)
11. Is Bold (e.g. 0 when false, 1 when true)
12. Is Italic (e.g. 0 when false, 1 when true)
13. Capitalisation (e.g. INITCAP, NOCAPS, ALLCAPS)
14. Punctuation information (e.g. NOPUNCT, PUNCT))
15. Line status (e.g. LINESTART, LINEIN, LINEND)
16. Font status (e.g. NEWFONT, SAMEFONT)

A.2 Feature Templates

Feature templates define the selection patterns from the corresponding descriptive vectors. We present for each template category, its description using the Wapiti syntax. Each line starting by a "#" means that the line is about an inactive feature or simply a comment.

A.2.1 Unigram Templates of Dictionary Segmentation Model (a.k.a. GROBID's First Model)

First Token (0)

U00:%x[-4,0]

U01:%x[-3,0]

U02:%x[-2,0]

U03:%x[-1,0]

U04:%x[0,0]

U05:%x[1,0]

U06:%x[2,0]

U07:%x[3,0]

U08:%x[4,0]

U08:%x[5,0]

Second token (1)

U00:%x[-4,1]

U01:%x[-3,1]

U02:%x[-2,1]

U03:%x[-1,1]

U04:%x[0,1]

U05:%x[1,1]

U06:%x[2,1]

U07:%x[3,1]

U08:%x[4,1]

Lower-cased first token (2)

U10:%x[-2,2]

U11:%x[-1,2]

U12:%x[0,2]

U13:%x[1,2]

U14:%x[2,2]

Prefix 1-4 characters (3-6)

U20:%x[0,3]

U21:%x[0,4]

U22:%x[0,5]

U23:%x[0,6]

Block info (7)

U40:%x[-1,7]

U41:%x[0,7]

U42:%x[1,7]

page info (8)

U60:%x[-1,8]

U61:%x[0,8]

U62:%x[1,8]

Font info (9-10)

U70:%x[-1,9]

U71:%x[0,9]

U72:%x[1,9]

U80:%x[-1,10]

U81:%x[0,10]

U82:%x[1,10]

Bold info (11)

U90:%x[-1,11]

U91:%x[0,11]

U92:%x[1,11]

Italic info (12)

UA0:%x[-1,12]

UA1:%x[0,12]

UA2:%x[1,12]

Capitalisation (13)

UB0:%x[0,13]

UB1:%x[1,13]

UB2:%x[-1,13]

```
# Digits (14)
UC0:%x[0,14]
UC1:%x[-1,14]
UC2:%x[1,14]
```

```
# Char info (15)
UD0:%x[0,15]
UD1:%x[-1,15]
UD2:%x[1,15]
```

```
# Dict info (16-22)
UE0:%x[0,16]
UE1:%x[0,17]
UE2:%x[0,18]
UE3:%x[0,19]
UE4:%x[0,20]
UE5:%x[0,21]
UE6:%x[0,22]
UEI:%x[1,16]
UEJ:%x[1,17]
UEK:%x[1,18]
UEL:%x[1,19]
UEM:%x[1,20]
UEN:%x[1,21]
UEO:%x[1,22]
```

```
# punctuation info (23)
UG0:%x[-1,23]
UG1:%x[0,23]
UG2:%x[1,23]
```

```
# relative document position (24)
UH0:%x[-1,24]
UH1:%x[0,24]
UH2:%x[1,24]
```

```
# relative page position coordinate (25)
UI0:%x[-1,25]
UI1:%x[0,25]
UI2:%x[1,25]
```

```
# number of punctuation charcaters in the line (26)
UI0:%x[-1,26]
UI1:%x[0,26]
UI2:%x[1,26]
```

```
# (scaled) line length (27)
UI0:%x[-1,27]
UI1:%x[0,27]
UI2:%x[1,27]
```



```
# bitmap connected to the current block (28)
UG0:%x[-2,28]
UG1:%x[-1,28]
UG2:%x[0,28]
UG3:%x[1,28]
UG4:%x[2,28]
UG5:%x[3,28]

# vector graphic connected to the current block (29)
UG0:%x[-2,29]
UG1:%x[-1,29]
UG2:%x[0,29]
UG3:%x[1,29]
UG4:%x[2,29]
UG5:%x[3,29]

# pattern repeated on several pages
UH0:%x[-2,30]
UH1:%x[-1,30]
UH2:%x[0,30]
UH3:%x[1,30]
UH4:%x[2,30]
UH5:%x[3,30]

# if we have a repeated pattern
UI1:%x[-1,31]
UI2:%x[0,31]
UI3:%x[1,31]

# if the block is in the page main area (1)
UJ1:%x[-2,32]
UJ1:%x[-1,32]
UJ2:%x[0,32]
UJ3:%x[1,32]
UJ3:%x[1,32]

# BigramActivated
#B
```

A.2.2 Unigram Templates from Dictionary Body Segmentation Model Onward

```
# Token
U00:%x[-4,0]
U01:%x[-3,0]
U02:%x[-2,0]
U03:%x[-1,0]
U04:%x[0,0]
U05:%x[1,0]
```

```
U06:%x[2,0]
U07:%x[3,0]
U08:%x[4,0]
U09:%x[-1,0]/%x[0,0]
U0A:%x[0,0]/%x[1,0]
U0B:%x[1,0]/%x[2,0]
U0C:%x[-2,0]/%x[-1,0]
U0E:%x[-2,0]/%x[-1,0]/%x[0,0]
U0E:%x[0,0]/%x[1,0]/%x[2,0]
```

```
# Lower-cased token
```

```
U10:%x[-2,1]
U11:%x[-1,1]
U12:%x[0,1]
U13:%x[1,1]
U14:%x[2,1]
```

```
# Prefix 1-4 characters
```

```
U20:%x[0,2]
U21:%x[0,3]
U22:%x[0,4]
U23:%x[0,5]
```

```
# Suffix 1-4 characters
```

```
U30:%x[0,6]
U31:%x[0,7]
U32:%x[0,8]
U33:%x[0,9]
```

```
# Font Size information
```

```
U40:%x[0,10]
U41:%x[1,10]
U42:%x[-1,10]
```

```
# Is Bold
```

```
U50:%x[0,11]
U51:%x[-1,11]
U52:%x[1,11]
```

```
# Is Italic
```

```
U60:%x[0,12]
U61:%x[-1,12]
U62:%x[1,12]
```

```
# Capitalisation
```

```
U70:%x[0,13]
U71:%x[-1,13]
U72:%x[1,13]
```

```
# Punctuation information
```

```
UA0:%x[0,14]
UA1:%x[-1,14]
UA3:%x[1,14]
```

```
# Line status
UB0:%x[-1,15]
UB1:%x[0,15]
UB2:%x[1,15]
```

```
# Font status
UC0:%x[-1,16]
UC1:%x[0,16]
UC2:%x[1,16]
```

```
# Bigram Activated
#B
```

A.2.3 Bigram Templates of Dictionary Segmentation Model (a.k.a. GROBID's First Model)

```
# First Token (0)
U00:%x[-4,0]
U01:%x[-3,0]
U02:%x[-2,0]
U03:%x[-1,0]
U04:%x[0,0]
U05:%x[1,0]
U06:%x[2,0]
U07:%x[3,0]
U08:%x[4,0]
U08:%x[5,0]
```

```
# Second token (1)
U00:%x[-4,1]
U01:%x[-3,1]
U02:%x[-2,1]
U03:%x[-1,1]
U04:%x[0,1]
U05:%x[1,1]
U06:%x[2,1]
U07:%x[3,1]
U08:%x[4,1]
```

```
# Lower-cased first token (2)
U10:%x[-2,2]
U11:%x[-1,2]
U12:%x[0,2]
U13:%x[1,2]
U14:%x[2,2]
```

Prefix 1-4 characters (3-6)

U20:%x[0,3]

U21:%x[0,4]

U22:%x[0,5]

U23:%x[0,6]

Block info (7)

U40:%x[-1,7]

U41:%x[0,7]

U42:%x[1,7]

page info (8)

U60:%x[-1,8]

U61:%x[0,8]

U62:%x[1,8]

Font info (9-10)

U70:%x[-1,9]

U71:%x[0,9]

U72:%x[1,9]

U80:%x[-1,10]

U81:%x[0,10]

U82:%x[1,10]

Bold info (11)

U90:%x[-1,11]

U91:%x[0,11]

U92:%x[1,11]

Italic info (12)

UA0:%x[-1,12]

UA1:%x[0,12]

UA2:%x[1,12]

Capitalisation (13)

UB0:%x[0,13]

UB1:%x[1,13]

UB2:%x[-1,13]

Digits (14)

UC0:%x[0,14]

UC1:%x[-1,14]

UC2:%x[1,14]

Char info (15)

UD0:%x[0,15]

UD1:%x[-1,15]

UD2:%x[1,15]

Dict info (16-22)

UE0:%x[0,16]

UE1:%x[0,17]

UE2:%x[0,18]

UE3:%x[0,19]

UE4:%x[0,20]

UE5:%x[0,21]

UE6:%x[0,22]

UEI:%x[1,16]

UEJ:%x[1,17]

UEK:%x[1,18]

UEL:%x[1,19]

UEM:%x[1,20]

UEN:%x[1,21]

UEO:%x[1,22]

punctuation info (23)

UG0:%x[-1,23]

UG1:%x[0,23]

UG2:%x[1,23]

relative document position (24)

UH0:%x[-1,24]

UH1:%x[0,24]

UH2:%x[1,24]

relative page position coordinate (25)

UI0:%x[-1,25]

UI1:%x[0,25]

UI2:%x[1,25]

number of punctuation charcaters in the line (26)

UI0:%x[-1,26]

UI1:%x[0,26]

UI2:%x[1,26]

(scaled) line length (27)

UI0:%x[-1,27]

UI1:%x[0,27]

UI2:%x[1,27]

bitmap connected to the current block (28)

UG0:%x[-2,28]

UG1:%x[-1,28]

UG2:%x[0,28]

UG3:%x[1,28]

UG4:%x[2,28]

UG5:%x[3,28]

vector graphic connected to the current block (29)

```

UG0:%x[-2,29]
UG1:%x[-1,29]
UG2:%x[0,29]
UG3:%x[1,29]
UG4:%x[2,29]
UG5:%x[3,29]

# pattern repeated on several pages
UH0:%x[-2,30]
UH1:%x[-1,30]
UH2:%x[0,30]
UH3:%x[1,30]
UH4:%x[2,30]
UH5:%x[3,30]

# if we have a repeated pattern
UI1:%x[-1,31]
UI2:%x[0,31]
UI3:%x[1,31]

# if the block is in the page main area (1)
UJ1:%x[-2,32]
UJ1:%x[-1,32]
UJ2:%x[0,32]
UJ3:%x[1,32]
UJ3:%x[1,32]

# BigramActivated
B

```

A.2.4 Bigram Templates from Dictionary Body Segmentation Model Onward

```

# Token
U00:%x[-4,0]
U01:%x[-3,0]
U02:%x[-2,0]
U03:%x[-1,0]
U04:%x[0,0]
U05:%x[1,0]
U06:%x[2,0]
U07:%x[3,0]
U08:%x[4,0]
U09:%x[-1,0]/%x[0,0]
U0A:%x[0,0]/%x[1,0]
U0B:%x[1,0]/%x[2,0]
U0C:%x[-2,0]/%x[-1,0]
U0E:%x[-2,0]/%x[-1,0]/%x[0,0]
U0E:%x[0,0]/%x[1,0]/%x[2,0]

```

```
# Lower-cased token
U10:%x[-2,1]
U11:%x[-1,1]
U12:%x[0,1]
U13:%x[1,1]
U14:%x[2,1]

# Prefix 1-4 characters
U20:%x[0,2]
U21:%x[0,3]
U22:%x[0,4]
U23:%x[0,5]

# Suffix 1-4 characters
U30:%x[0,6]
U31:%x[0,7]
U32:%x[0,8]
U33:%x[0,9]

# Font Size information
U40:%x[0,10]
U41:%x[1,10]
U42:%x[-1,10]

# Is Bold
U50:%x[0,11]
U51:%x[-1,11]
U52:%x[1,11]

# Is Italic
U60:%x[0,12]
U61:%x[-1,12]
U62:%x[1,12]

# Capitalisation
U70:%x[0,13]
U71:%x[-1,13]
U72:%x[1,13]

# Punctuation information
UA0:%x[0,14]
UA1:%x[-1,14]
UA3:%x[1,14]

# Line status
UB0:%x[-1,15]
UB1:%x[0,15]
UB2:%x[1,15]

# Font status
```

```
UC0:%x[-1,16]
UC1:%x[0,16]
UC2:%x[1,16]
```

```
# Bigram Activated
B
```

A.2.5 Engineered Templates from Dictionary Body Segmentation Model Onward

```
# Token
U00:%x[-7,0]
U01:%x[-6,0]
U02:%x[-5,0]
U03:%x[-4,0]
U04:%x[-3,0]
U05:%x[-2,0]
U06:%x[-1,0]
U07:%x[0,0]
U08:%x[1,0]
U09:%x[2,0]
U0A:%x[3,0]
U0B:%x[4,0]
U0C:%x[5,0]
U0D:%x[6,0]
U0E:%x[7,0]
U0F:%x[-1,0]/%x[0,0]
U0G:%x[0,0]/%x[1,0]
U0H:%x[1,0]/%x[2,0]
U0E:%x[-2,0]/%x[-1,0]
U0F:%x[-2,0]/%x[-1,0]/%x[0,0]
U0I:%x[0,0]/%x[1,0]/%x[2,0]
```

```
# Lower-cased token
U10:%x[-5,1]
U11:%x[-4,1]
U12:%x[-3,1]
U13:%x[-2,1]
U14:%x[-1,1]
U15:%x[0,1]
U16:%x[1,1]
U17:%x[2,1]
U18:%x[3,1]
U19:%x[4,1]
U0A:%x[5,1]
```

```
# Prefix 1-4 characters
U20:%x[0,2]
U21:%x[0,3]
U22:%x[0,4]
```


U23:%x[0,5]

Suffix 1-4 characters

U30:%x[0,6]

U31:%x[0,7]

U32:%x[0,8]

U33:%x[0,9]

FontSize information

U40:%x[-4,10]

U41:%x[-3,10]

U42:%x[-2,10]

U43:%x[-1,10]

U44:%x[0,10]

U45:%x[1,10]

U46:%x[2,10]

U47:%x[3,10]

U48:%x[4,10]

Is Bold

U50:%x[-4,11]

U51:%x[-3,11]

U52:%x[-2,11]

U53:%x[-1,11]

U54:%x[0,11]

U55:%x[1,11]

U56:%x[2,11]

U57:%x[3,11]

U58:%x[4,11]

Is Italic

U60:%x[-4,11]

U61:%x[-3,11]

U62:%x[-2,11]

U63:%x[-1,11]

U64:%x[0,12]

U65:%x[1,12]

U66:%x[2,12]

U67:%x[3,12]

U68:%x[4,11]

Capitalisation

U70:%x[-4,13]

U71:%x[-3,13]

U72:%x[-2,13]

U73:%x[-1,13]

U74:%x[0,13]

U75:%x[1,13]

U76:%x[2,13]

U77:%x[3,13]

U78:%x[4,13]

Punctuation information

UA0:%x[-4,14]

UA1:%x[-3,14]

UA2:%x[-2,14]

UA3:%x[-1,14]

UA4:%x[0,14]

UA5:%x[1,14]

UA6:%x[2,14]

UA7:%x[3,14]

UA8:%x[4,14]

Line status

UB0:%x[-4,15]

UB1:%x[-3,15]

UB2:%x[-2,15]

UB3:%x[-1,15]

UB4:%x[0,15]

UB5:%x[1,15]

UB6:%x[2,15]

UB7:%x[3,15]

UB8:%x[4,15]

Font status

UC0:%x[-4,16]

UC1:%x[-3,16]

UC2:%x[-2,16]

UC3:%x[-1,16]

UC4:%x[0,16]

UC5:%x[1,16]

UC6:%x[2,16]

UC7:%x[3,16]

UC8:%x[4,16]

#Bigram Activated

B

Appendix B

Models Call for the "Parse Full Dictionary" Level in GROBID-Dictionaries' Web Application

Figures B.1, B.2, B.3, B.4, B.5, B.6 and B.7 illustrate the different model selection cases for parsing the components of a lexical entry (i.e the output of the **Lexical Entry** model). The parsing of some structures, like related entries and cross-references, can be activated at, either the lexical entry level, or all the levels where they appear. These two parsing alternatives can be activated by selecting respectively, "at entries" and "all" options. The "Download TEI Result" button in Figure B.7 enables saving the TEI output of the different activated models. This option is possible after each call of the models of a parsing level.

The screenshot shows the GROBID-Dictionaries web application interface. At the top, there is a navigation bar with links for 'About', 'Dictionary services', 'Bibliography services', 'Admin', and 'Doc'. Below this, the 'Parsing Level' dropdown menu is open, showing the following options: 'Parse full dictionary' (selected), 'Morphological & Grammatical information', 'form' (checked), 'form & gramGrp', and 'Skip'. To the right of the menu, there are three dropdown menus for 're (at entry)', 'xr (at entry)', and 'subEntry'. Below these are a 'Select file' button and a 'Submit' button. At the bottom of the page, there is a copyright notice: '© GROBID-Dictionaries contributors - 2020'.

FIGURE B.1: Selecting Morphological and Grammatical Models

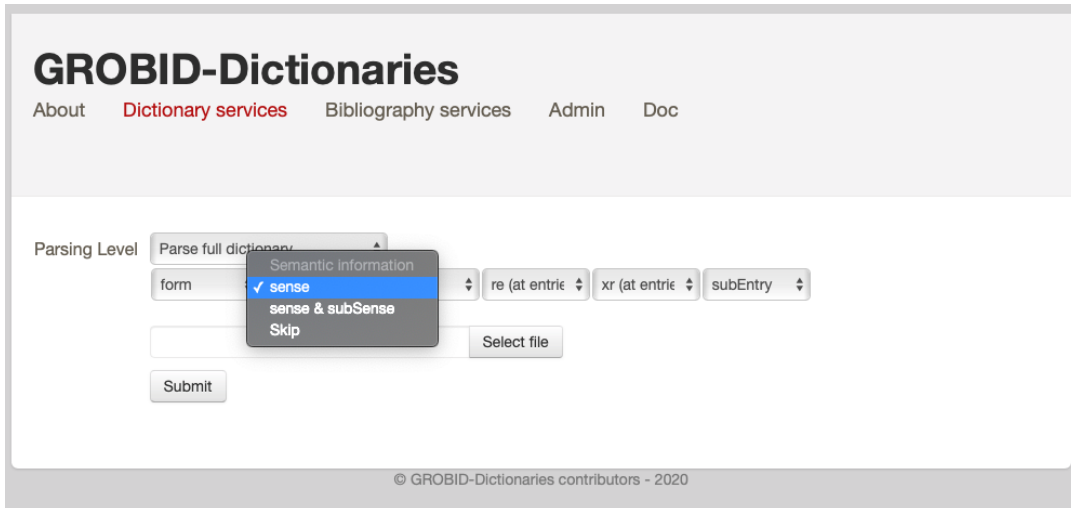


FIGURE B.2: Selecting Semantic Models

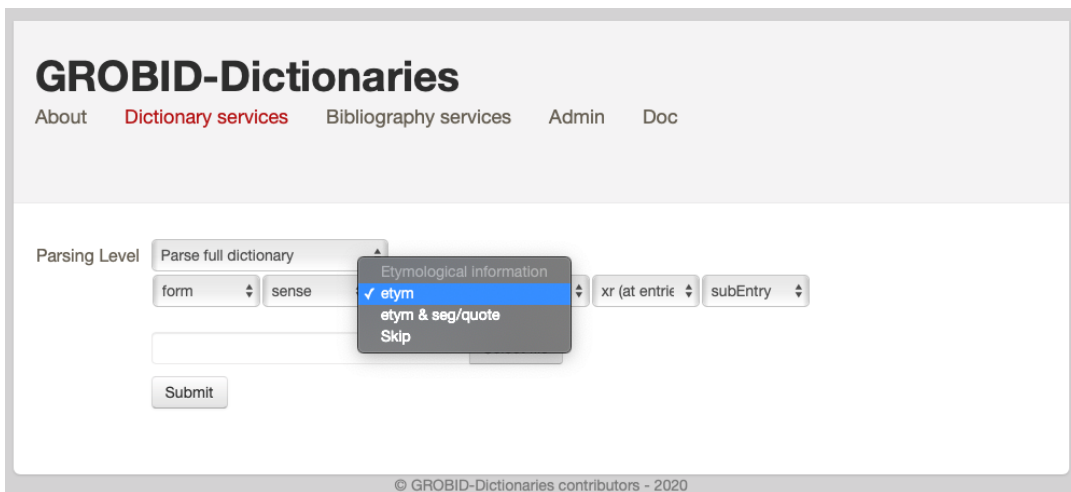


FIGURE B.3: Selecting Etymological Models

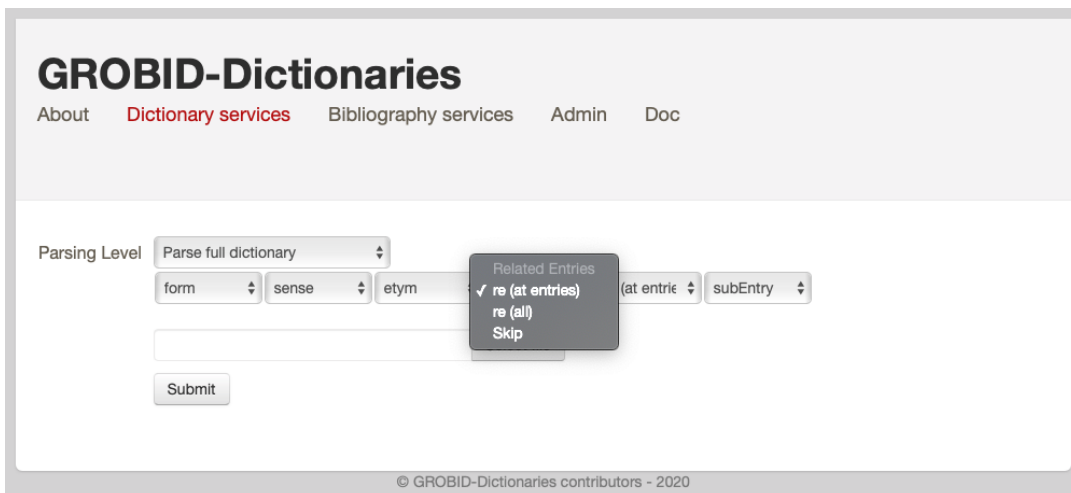


FIGURE B.4: Selecting Lexical Entry Model for Parsing Related Entries

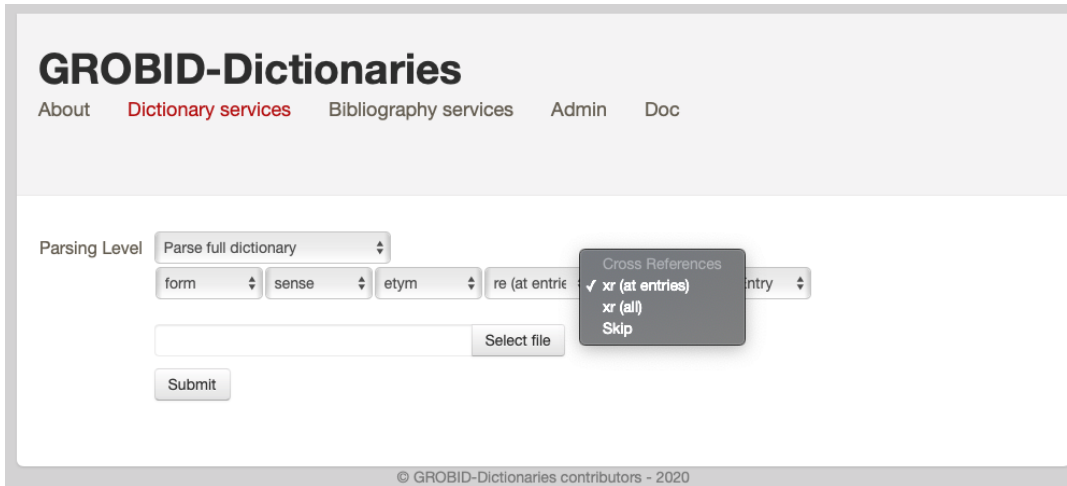


FIGURE B.5: Selecting Cross-Reference Models

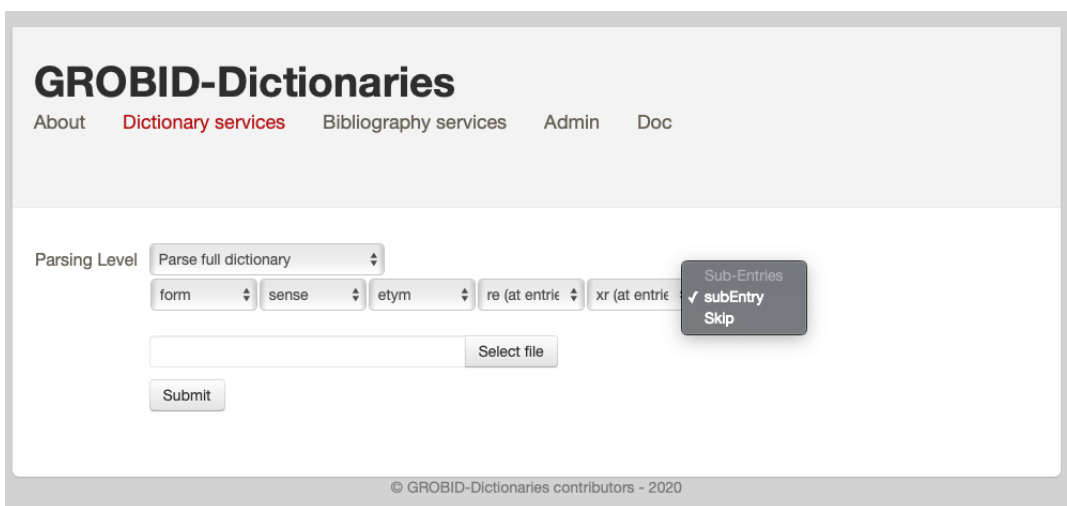


FIGURE B.6: Selecting Lexical Entry Model for Parsing Sub-Entries

The screenshot displays the GROBID-Dictionaries web application interface. At the top, the title "GROBID-Dictionaries" is followed by navigation links: "About", "Dictionary services", "Bibliography services", "Admin", and "Doc". Below this, the "Parsing Level" is set to "Parse full dictionary". A row of dropdown menus includes "form", "sense", "etym", "re (at entrée)", "xr (at entrée)", and "subEntry". A file input field contains "basicEnglish-213-215.pdf", with "Change" and "Remove" buttons. Below the file input are "Submit", "Download TEI Result", and "Get Lemmas" buttons. The main content area shows the resulting TEI XML output:

```
<?xml version="1.0" encoding="UTF-8"?>
<TEI
  xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader>
    <encodingDesc>
      <appInfo>
        <application version="0.5.6-SNAPSHOT" ident="GROBID" when="2020-05-18T15:59+0000">
          <ref target="https://github.com/MedKhem/grobid-dictionaries">GROBID_Dictionaries - A machine
learning software for structuring digitized dictionaries</ref>
        </application>
      </appInfo>
    </encodingDesc>
    <fileDesc>
      <titleStmt>
        <title level="a" type="main"/>
      </titleStmt>
    </fileDesc>
  </teiHeader>
  <text>
```

FIGURE B.7: State after of Full Parsing Selection

Primary Source References

- Almonjid (2014). *The Dictionary of Language and Proper Nouns*. Beirut, Lebanon: Dar el-Machreq.
- Alvarado, Francisco de (1593). "Vocabulario en Lengua Mixteca. Hecho por los Padres de la Orden de Predicadores". In:
- Berthelot, André (1886). *La grande encyclopédie: inventaire raisonné des sciences, des lettres et des arts*. Vol. 1. Société anonyme de la Grande encyclopédie.
- Didot-Bottin (1901). *Annuaire-almanach du commerce, de l'industrie, de la magistrature et de l'administration : ou almanach des 500.000 adresses de Paris, des départements et des pays étrangers : Firmin Didot et Bottin réunis*. Librairie de Firmin-Didot.
- Ernest, Klein (1966). *A Comprehensive etymological dictionary of the English language*. Elsevier Publishing Company.
- Ernout, Alfred et al. (1951). *Dictionnaire étymologique de la langue latine: histoire des mots*. Klincksieck Paris.
- Furetière, Antoine (1701). *Dictionnaire Universel, contenant généralement tous les Mots François tant vieux que modernes, et les Termes des Sciences et des Arts*. Vol. 1-3. The Hague: Arnoud et Reinier Leers.
- Galley, Samuel (1964). *Dictionnaire fang-français et français-fang: suivi d'une grammaire fang*. H. Messeiller.
- Goedel, Gustav (1902). *Etymologisches Wörterbuch der deutschen Seemannssprache*. Vol. Bd. 1 (A – K). Kiel, Leipzig.
- Hindley, Alan et al. (2000). *Old French-English Dictionary*. Cambridge University Press Cambridge, UK.
- Hindley, Geoffrey (1971). *Larousse encyclopedia of music*. Chartwell Books.
- Hornby, Albert Sydney et al. (1974). *Oxford advanced learner's dictionary of current English*. Vol. 1428. Oxford university press.
- Kluge, Friedrich and Frederick Lutz (1898). *English Etymology. A Select Glossary Serving as an Introduction to the History of the English Language*. Boston.
- Lamy, Marie-Noklle and Richard Towell (1998). *The Cambridge French-English Thesaurus*. Cambridge University Press.
- Larousse (1972). *Dictionnaire des noms communs en couleurs*. Larousse France loisirs, Paris.
- Larousse, Librairie et al. (1982). *Grand dictionnaire encyclopédique Larousse*. Librairie Larousse.
- Larousse, Pierre (1948). *Petit Larousse illustré 1948*. Larousse.
- Liddell, Henry George and Robert Scott (1896). *An intermediate greek-english lexicon*. Harper Brothers.
- Littré, Emile (1873). *Dictionnaire de la langue française*. L. Hachette et Cie.
- Mueller, Eduard (1878). *Etymologisches Wörterbuch der englischen Sprache*. Vol. Bd. 1 (A – K). Cöthen.

- Muller, Charles and Michael Beddow (2002). "Moving into XML Functionality: The Combined Digital Dictionaries of Buddhism and East Asian Literary Terms". In: *Journal of Digital Information* 3.2.
- Publishing, Bloomsbury (2009). *Easier English Basic Dictionary: Pre-Intermediate Level. Over 11,000 terms clearly defined.* Easier English. Bloomsbury Publishing. ISBN: 9781408102022. URL: <https://books.google.de/books?id=nwVCBAAAQBAJ>.
- Roget, Peter Mark (1911). *Roget's Thesaurus of English Words and Phrases...* TY Crowell Company.
- Uhlenbeck, C. C (1900). *Kurzgefaßtes etymologisches Wörterbuch der gotischen Sprache.* Amsterdam.
- Urdang, Laurence et al. (1986). *Longman synonym dictionary.* Longman.
- Wiegand, Herbert Ernst (2014). *Internationale Bibliographie zur germanistischen Lexikographie und Wörterbuchforschung: Band 4: Nachträge.* De Gruyter Mouton.

Scholarly References

- Abel, Andrea (2012). "Dictionary Writing Systems and Beyond". In: *Electronic Lexicography*, pp. 83–106.
- Aduriz, Itziar et al. (1998). "EDBL: a Multi-Purposed Lexical Support for the Treatment of Basque". In: *Proceedings of the First International Conference on Language Resources and Evaluation*. Vol. 2. Citeseer, pp. 821–826.
- Artetxe, Mikel et al. (2017). "Unsupervised Neural Machine Translation". In: *CoRR abs/1710.11041*. arXiv: [1710.11041](https://arxiv.org/abs/1710.11041).
- Atkins, Beryl T Sue (1991). "Building a Lexicon The Contribution of Lexicography". In: *International Journal of lexicography* 4.3, pp. 167–204.
- Atkins, BT Sue and Michael Rundell (2008). *The Oxford guide to practical lexicography*. Oxford University Press.
- Bago, Petra and Nikola Ljubešić (2015). "Using machine learning for language and structure annotation in an 18th century dictionary". In: *Electronic lexicography in the 21st century: linking lexical data in the digital age*.
- Banski, Piotr, Jack Bowers, and Tomaz Erjavec (2017). "TEI-lex0 Guidelines for the Encoding of Dictionary Information on Written and Spoken forms". In: *Electronic lexicography in the 21st century: Proceedings of eLex 2017 conference*. Lexical Computing, pp. 485–494.
- Beinborn, Lisa, Torsten Zesch, and Iryna Gurevych (2013). "Cognate production using character-based machine translation". In: *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pp. 883–891.
- Bertagna, Francesca et al. (2004). "Content Interoperability of Lexical Resources: Open Issues and "MILE" Perspectives." In: *LREC*.
- Breuel, Thomas M (2008). "The OCRopus open source OCR system". In: *Document Recognition and Retrieval XV*. Vol. 6815. International Society for Optics and Photonics, 68150F.
- Buchanan, Bruce G and Richard O Duda (1983). "Principles of rule-based expert systems". In: *Advances in computers*. Vol. 22. Elsevier, pp. 163–216.
- Budin, Gerhard, Stefan Majewski, and Karlheinz Mörth (2012). "Creating Lexical Resources in TEI P5. A Schema for Multi-purpose Digital Dictionaries". In: *Journal of the Text Encoding Initiative* 3.
- Burges, Christopher JC (1998). "A tutorial on support vector machines for pattern recognition". In: *Data mining and knowledge discovery 2.2*, pp. 121–167.
- Calzolari, Nicoletta (2008). "Approaches towards a 'Lexical Web': the Role of Interoperability". In: *ICGL2008*, pp. 34–42.

- Calzolari, Nicoletta, Antonio Zampolli, and Alessandro Lenci (2002). "Towards a standard for a multilingual lexical entry: The EAGLES/ISLE initiative". In: *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, pp. 264–279.
- Cheng, Yong (2019). "Semi-supervised learning for neural machine translation". In: *Joint Training for Neural Machine Translation*. Springer, pp. 25–40.
- Copestake, Ann (1992). "The ACQUILEX LKB: representation issues in semi-automatic acquisition of large lexicons". In: *Proceedings of the third conference on Applied natural language processing*. Association for Computational Linguistics, pp. 88–95.
- Crist, Sean (2011). "Processing the Text of Bilingual Print Dictionaries". In: URL: http://www.sean-crist.com/all/crist_dictionaries_20111210.pdf.
- Czaykowska-Higgins, Ewa, Martin D Holmes, and Sarah M Kell (2014). "Using TEI for an endangered language lexical resource: The Nxa?amxcín Database-Dictionary Project". In: *Language Documentation & Conservation* 8, pp. 1–37.
- Declerck, Thierry, Karlheinz Mörth, and Piroska Lendvai (2012). "Accessing and standardizing Wiktionary lexical entries for the translation of labels in Cultural Heritage taxonomies". In: *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pp. 2511–2514.
- DeLFT (2018–2020). <https://github.com/kermitt2/delft>. swb: 1 : dir : 54eb292e1c0af764e27dd179596f64679e44d06e.
- DeMenthon, Daniel and Marc Vuilleumier (2003). *LAMP HMM*. Version 0.9. URL: http://www.cfar.umd.edu/daniel/LAMP_HMM.zip.
- Dendien, Jacques and Jean-Marie Pierrel (2003). "Le trésor de la langue française informatisé. Un exemple d'informatisation d'un dictionnaire de langue de référence". In: *TAL. Traitement automatique des langues* 44.2, pp. 11–37.
- Eckle-Kohler, Judith and Iryna Gurevych (2012). "Standardizing lexical-semantic resources - Fleshing out the abstract standard LMF". In: *KONVENS*.
- Eckle-Kohler, Judith, John Philip McCrae, and Christian Chiarcos (2015). "lemonUby—A large, interlinked, syntactically-rich lexical resource for ontologies". In: *Semantic Web* 6.4, pp. 371–378.
- Ehrmann, Maud et al. (2014). "A multilingual semantic network as linked data: lemon-babelnet". In: *3rd Workshop on Linked Data in Linguistics: Multilingual Knowledge Resources and Natural Language Processing*, p. 72.
- Erjavec, Tomaz (2004). "MULTEXT-East Version 3: Multilingual Morphosyntactic Specifications, Lexicons and Corpora." In: *LREC*.
- Fahmy, Daaa Mohamed Fayed1 Aly Aly and Mohsen Abdelrazek Rashwan3 Wafaa Kamel Fayed (2014). "Towards Structuring an Arabic-English Machine-Readable Dictionary Using Parsing Expression Grammars". In:
- Fayed, Daaa et al. (2014). "Towards Structuring an Arabic-English Machine-Readable Dictionary Using Parsing Expression Grammars, International Journal of Computational Linguistics Research". In: 5, pp. 1–13.

- Flekova, Lucie and Iryna Gurevych (2015). "Personality profiling of fictional characters using sense-level links between lexical resources". In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1805–1816.
- Foppiano, Luca and Laurent Romary (2018). "entity-fishing: a DARIAH entity recognition and disambiguation service". In: *Digital Scholarship in the Humanities*. Tokyo, Japan.
- Fowler, Martin and Kendall Scott (2004). *UML distilled: a brief guide to the standard object modeling language*. Addison-Wesley Professional.
- Francopoulo, Gil et al. (2006). "Lexical Markup Framework (LMF)". In: *LREC*.
- George, Monte (2013). "LMF in US Government Language Resource Management". In: *LMF Lexical Markup Framework*, pp. 243–261.
- Gibbon, Dafydd (2000). "Computational lexicography". In: *Lexicon Development for speech and language processing*. Springer, pp. 1–42.
- GROBID (2008–2020). <https://github.com/kermitt2/grobid>. swb: 1:dir:dab86b296e3c3216e2241968f0d63b68e8209d3c.
- Gui, Tao et al. (2017). "Part-of-speech tagging for twitter with adversarial neural networks". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2411–2420.
- Gurevych, Iryna et al. (2012). "Uby: A large-scale unified lexical-semantic resource based on LMF". In: *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 580–590.
- Hanks, Patrick (2013). "Lexicography from Earliest Times to the Present". In: *The Oxford Handbook of the History of Linguistics*, pp. 503–536.
- Hauser, Ralf and Angelika Storrer (1993). "Dictionary entry parsing using the LexParse system". In: *Lexikographica* 9, pp. 174–219.
- Horák, Aleš and Adam Rambousek (2007). "Dictionary management system for the DEB development platform". In: *Proceedings of the 4th International Workshop on Natural Language Processing and Cognitive Science (NLPCS, aka NLUCS)*, pp. 129–138.
- Joffe, David and Gilles-Maurice De Schryver (2004). "TshwaneLex: a state-of-the-art dictionary compilation program". In: *11th EURALEX International Congress (EURALEX-2004)*. Faculté des Lettres et des Sciences Humaines, pp. 99–104.
- Kahle, Philip et al. (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. 4. IEEE, pp. 19–24.
- Kaplan, Frédéric (2015). "The venice time machine". In: *Proceedings of the 2015 ACM Symposium on Document Engineering*, pp. 73–73.
- Karagol-Ayan, Burcu, David Doermann, and Bonnie J Dorr (2003). "Acquisition of bilingual MT lexicons from OCR'd dictionaries". In: *Proceedings of the 9th MT Summit*, pp. 208–215.
- Kenning, Marie-Madeleine (2010). "What are parallel and comparable corpora and how can we use them?" In: *The Routledge handbook of corpus linguistics*. Routledge, pp. 515–528.

- Khan, Anas (2018). "Towards the Representation of Etymological Data on the Semantic Web". In: *Information* 9.12, p. 304.
- Khan, Fahad et al. (2017). "The Challenges of Converting Legacy Lexical Resources to Linked Open Data using Ontolex-Lemon: The Case of the Intermediate Liddell-Scott Lexicon". In: *LDK Workshops*.
- Khemakhem, Aida et al. (2016). "ISO standard modeling of a large Arabic dictionary". In: *Natural Language Engineering* 22.6, pp. 849–879.
- Khemakhem, Aida et al. (2009). "Towards an automatic conversion approach of editorial Arabic dictionaries into LMF-ISO 24613 standardized model". In: *Proceedings of the Second International Conference on Arabic Language Resources and Tools, Cairo, Egypt*.
- Klyne, Graham (2004). "Resource description framework (RDF): Concepts and abstract syntax". In: <http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>.
- Kuzmina, Vera and Anna Rylova (2010). "ABBY Lingvo electronic dictionary platform and Lingvo ?ontent dictionary writing system". In: *eLexicography in the 21st Century: New Challenges, New applications*, p. 419.
- Lample, Guillaume et al. (2016). "Neural Architectures for Named Entity Recognition". In: *CoRR* abs/1603.01360. arXiv: 1603.01360.
- Lavergne, Thomas, Olivier Cappé, and François Yvon (2010). "Practical very large scale CRFs". In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 504–513.
- Le, Hang et al. (2019). "FlauBERT: Unsupervised Language Model Pre-training for French". In: *arXiv preprint arXiv:1912.05372*.
- Lemnitzer, Lothar and Claudia Kunze (2005). *Lexical Acquisition*. ESSLI Course, Heriot-Watt University.
- Lipinski, Mario et al. (2013). "Evaluation of header metadata extraction approaches and tools for scientific PDF documents". In: *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*. ACM, pp. 385–386.
- Lopresti, Daniel (2009). "Optical character recognition errors and their effects on natural language processing". In: *International Journal on Document Analysis and Recognition (IJ DAR)* 12.3, pp. 141–151.
- Ma, Huanfeng and David Scott Doermann (2003). "Bootstrapping structured page segmentation". In: *Document Recognition and Retrieval X*. Vol. 5010. International Society for Optics and Photonics, pp. 179–188.
- Ma, Huanfeng et al. (2003). "Parsing and tagging of bilingual dictionaries". In: *Traitement Automatique Des Langues* 44.2, pp. 125–149.
- Magazine, D-Lib (1998). "The Perseus Project and beyond". In: *D-Lib Magazine*.
- Maks, Isa, Carole Tiberius, and Remco van Veenendaal (2008). "Standardising Bilingual Lexical Resources According to the Lexicon Markup Framework". In: *LREC*.
- Mangeot-Nagata, Mathieu (2006). "Dictionary building with the Jibiki platform". In: *XII Euralex International Congress*, pp. 185–188.
- Martin, Louis et al. (2019). *CamemBERT: a Tasty French Language Model*. arXiv: 1911.03894 [cs.CL].

- Matuschek, Michael and Iryna Gurevych (2013). "Dijkstra-wsa: A graph-based approach to word sense alignment". In: *Transactions of the Association for Computational Linguistics* 1, pp. 151–164.
- Maxwell, Michael and Aric Bills (2017). "Endangered data for endangered languages: Digitizing print dictionaries". In: *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pp. 85–91.
- McBride, Brian (2004). "The resource description framework (RDF) and its vocabulary description language RDFS". In: *Handbook on ontologies*. Springer, pp. 51–65.
- McCallum, Andrew Kachites (2002). *Mallet: A machine learning for language toolkit (2002)*.
- McCrae, John, Dennis Spohr, and Philipp Cimiano (2011). "Linking lexical resources and ontologies on the semantic web with lemon". In: *Extended Semantic Web Conference*. Springer, pp. 245–259.
- McCrae, John et al. (2012). "Interchanging lexical resources on the Semantic Web". In: *Language Resources and Evaluation* 46.4, pp. 701–719. ISSN: 1574-0218.
- McCrae, John P et al. (2017). "The OntoLex-Lemon Model: Development and Applications". In: *Electronic lexicography in the 21st century: Proceedings of eLex 2017 conference*. Lexical Computing, pp. 587–597.
- McGuinness, Deborah L, Frank Van Harmelen, et al. (2004). "OWL web ontology language overview". In: *W3C recommendation* 10.10, p. 2004.
- McNamara, Michael (2003). "Dictionaries for all: XML to Final Product". In: *XML Conference*.
- Meftah, Sara and Nasredine Semmar (2018). "A neural network model for part-of-speech tagging of social media texts". In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.
- Miller, Tristan et al. (2016). "Sense-annotating a Lexical Substitution Data Set with Ubyline". In: *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. Portorož, Slovenia.
- Mutuvi, Stephen et al. (2018). "Evaluating the Impact of OCR Errors on Topic Modeling". In: *Maturity and Innovation in Digital Libraries*. Ed. by Milena Dobрева, Annika Hinze, and Maja Žumer. Cham: Springer International Publishing, pp. 3–14.
- Mykowiecka, Agnieszka, Piotr Rychlik, and Jakub Waszczuk (2012). "Building an electronic dictionary of old polish on the base of the paper resource". In: *Proceedings of the Workshop on Adaptation of Language Resources and Tools for Processing Cultural Heritage at LREC*, pp. 16–21.
- Navigli, Roberto and Simone Paolo Ponzetto (2012). "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network". In: *Artificial Intelligence* 193, pp. 217–250.
- Okazaki, Naoaki (2007). "CRFsuite: a fast implementation of conditional random fields (CRFs)". In: *URL* <http://www.chokkan.org/software/crfsuite>.
- Peters, Matthew E. et al. (2017). "Semi-supervised sequence tagging with bidirectional language models". In: *CoRR* abs/1705.00108. arXiv: [1705.00108](https://arxiv.org/abs/1705.00108).

- Plank, Barbara, Anders Søgaard, and Yoav Goldberg (2016). "Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss". In: *CoRR abs/1604.05529*. arXiv: 1604.05529.
- Pustejovsky, James and Amber Stubbs (2012). *Natural Language Annotation for Machine Learning: A guide to corpus-building for applications*. " O'Reilly Media, Inc."
- Ranaivo-Malançon, Bali et al. (2017). "Transforming Semi-Structured Indigenous Dictionary into Machine-Readable Dictionary". In: *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 9.3-11, pp. 7–11.
- Romary, Laurent (2010). *Using the TEI framework as a possible serialization for LMF*. Rendering endangered languages lexicons interoperable through standards harmonization.
- (2015). "TEI and LMF crosswalks". In: *JLCL* 30, pp. 47–70.
- Romary, Laurent and Patrice Lopez (2015). "GROBID-Information Extraction from Scientific Publications". In: *ERCIM News* 100.
- Romary, Laurent and Toma Tasovac (2018). "TEI Lex-0: A Target Format for TEI-Encoded Dictionaries and Lexical Resources". In: *TEI Conference and Members' Meeting*. Tokyo, Japan.
- Saleem, Ozair and Seemab Latif (2012). "Information extraction from research papers by data integration and data validation from multiple header extraction sources". In: *Proceedings of the World Congress on Engineering and Computer Science*. Vol. 1, pp. 177–180.
- Soria, Claudia, Monica Monachini, and Piek Vossen (2009). "Wordnet-LMF: fleshing out a standardized format for wordnet interoperability". In: *Proceedings of the 2009 international workshop on Intercultural collaboration*. ACM, pp. 139–146.
- Sperberg-McQueen, C Michael, Lou Burnard, et al. (1994). *Guidelines for electronic text encoding and interchange*. Vol. 1. Text Encoding Initiative Chicago and Oxford.
- Steingrímsson, Steinbór (2018). "Digitizing the Icelandic-Danish Blöndal Dictionary". In: *DHN*.
- Sutton, Charles, Andrew McCallum, et al. (2012). "An introduction to conditional random fields". In: *Foundations and Trends® in Machine Learning* 4.4, pp. 267–373.
- Svensén, Bo (2009). "A handbook of lexicography". In: *The Theory and Practice of Dictionary-Making*.
- Thaiprayoon, Santipong and Alisa Kongthon Choochart Haruechaiyasak (2016). "PDF Extraction Based on Lexical Analysis for Thai Texts". In: *International Journal of Applied Computer Technology and Information Systems* 5.1.
- Tiedemann, Jörg (2014). "Improved text extraction from PDF documents for large-scale natural language processing". In: *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, pp. 102–112.
- Tkaczyk, Dominika et al. (2018). "Machine Learning vs. Rules and Out-of-the-Box vs. Retrained: An Evaluation of Open-Source Bibliographic Reference and Citation Parsers". In: *JCDL*.

- Torre, Maria F. De La et al. (2018). "MATESC: Metadata-Analytic Text Extractor and Section Classifier for Scientific Publications". In: *KDIR*.
- Wick, Christoph, Christian Reul, and Frank Puppe (2018). "Calamari-A High-Performance Tensorflow-based Deep Learning Package for Optical Character Recognition". In: *arXiv preprint arXiv:1807.02004*.
- Wu, Yonghui et al. (2016). "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation". In: *CoRR* abs/1609.08144. arXiv: [1609.08144](https://arxiv.org/abs/1609.08144).
- Zhu, Xiaojin (2010). *Cs838-1 advanced nlp: Conditional random fields*. Tech. rep.

Author's Own References

- Bowers, Jack, Mohamed Khemakhem, and Laurent Romary (2019). "TEI Encoding of a Classical Mixtec Dictionary Using GROBID- Dictionaries". In: *ELEX 2019: Smart Lexicography*. Sintra, Portugal.
- Khemakhem, Mohamed, Luca Foppiano, and Laurent Romary (2017). "Automatic extraction of TEI structures in digitized lexical resources using conditional random fields". In: *electronic lexicography, eLex 2017*. Leiden, Netherlands.
- Khemakhem, Mohamed, Axel Herold, and Laurent Romary (2018). "Enhancing Usability for Automatically Structuring Digitised Dictionaries". In: *GLOBALLEX workshop at LREC 2018*. Miyazaki, Japan.
- Khemakhem, Mohamed et al. (2018a). "Automatically Encoding Encyclopedic-like Resources in TEI". In: *The annual TEI Conference and Members Meeting*. Tokyo, Japan.
- Khemakhem, Mohamed et al. (2018b). "Fueling Time Machine: Information Extraction from Retro-Digitised Address Directories". In: *JADH2018 "Leveraging Open Data"*. Tokyo, Japan.
- Khemakhem, Mohamed et al. (2019). "How OCR Performance can Impact on the Automatic Extraction of Dictionary Content Structures". In: *19th annual Conference and Members' Meeting of the Text Encoding Initiative Consortium (TEI) -What is text, really? TEI and beyond*. Graz, Austria.
- Lindemann, David, Mohamed Khemakhem, and Laurent Romary (2018). "Retro-digitizing and Automatically Structuring a Large Bibliography Collection". In: *European Association for Digital Humanities (EADH) Conference*.
- Romary, Laurent et al. (2017). *Report on Standardization (draft)*. Technical Report Deliverable 4.2. Inria.
- Romary, Laurent et al. (2019). "LMF Reloaded". In: *AsiaLex 2019: Past, Present and Future*. Istanbul, Turkey.
- Rondeau Du Noyer, Lucie et al. (2019). "Scaling up Automatic Structuring of Manuscript Sales Catalogues". In: *TEI 2019: What is text, really? TEI and beyond*. Graz, Austria.