# Contribution to natural language generation : systems and evaluation

Moussa Kamal Eddine

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2022IPPAX143

Thèse de doctorat

# Contributions to Natural Language Generation : Systems and Evaluation

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à l'École Polytechnique

École doctorale n°626 : l'École Doctorale de l'Institut Polytechnique de Paris (ED IP Paris)
Spécialité de doctorat : Mathématiques et Informatique

Thèse présentée et soutenue à Palaiseau, le 16/12/2022, par

Moussa Kamal Eddine

Composition du Jury :

| | |
|---|---|
| Ioana Manolescu<br>Directrice de recherche, Inria (Saclay) | Président |
| Eduard Hovy<br>Professeur. Language Technologies Institute | Rapporteur |
| Eric Gaussier<br>Professeur, University Grenoble Alps (LIG) | Rapporteur |
| Nizar Habash<br>Professeur,New York University Abu Dhabi (CAMeL) | Examinateur |
| Jie Tang<br>Professeur, Tsinghua University | Examinateur |
| Alexandros Potamianos<br>Chargé de recherche, National Technical University of Athens | Examinateur |
| Michalis Vazirgiannis<br>Professeur, École Polytechnique (LIX) | Directeur de thèse |
| Nadi Tomeh<br>Professeur, Université Sorbonne Paris Nord (LIPN) | Invité |

# ABSTRACT

In recent years, the Natural Language Generation (NLG) field has changed drastically. This shift, which can be partially attributed to the notable advance in hardware, led to recent efforts in NLG to be focused on data-driven methods leveraging large pretrained Neural Networks (NNs). However, this progress gave rise to new challenges related to computational requirements, accessibility, and evaluation strategies, to name a few. In this dissertation, we are primarily concerned with contributing to the efforts to mitigate these challenges.

To address the lack of monolingual generative models for some languages, we start by introducing *BARThez* and *AraBART*, the first large-scale pretrained seq2seq models for French and Arabic, respectively. Being based on BART, these models are particularly well-suited for generative tasks. We evaluate BARThez on five discriminative tasks from the FLUE benchmark and two generative tasks from a novel summarization dataset, OrangeSum, that we created for this research. We show BARThez to be very competitive with state-of-the-art BERT-based French language models such as CamemBERT and FlauBERT. We also continue the pretraining of a multilingual BART on BARThez' corpus, and show our resulting model, mBARThez, to significantly boost BARThez' generative performance. On the other hand, We show that AraBART achieves the best performance on multiple abstractive summarization datasets, outperforming strong baselines.

Finally, we focus on the NLG system evaluation by proposing *DATScore* and *FrugalScore*. DATScore uses data augmentation techniques to improve the evaluation of machine translation and other NLG tasks. Our main finding is that introducing data augmented translations of the source and reference texts is greatly helpful in evaluating the quality of the generated translation. We also propose two novel score averaging and term weighting strategies to improve the original score computing process of BARTScore. Experimental results on WMT show that DATScore correlates better with human meta-evaluations than the other recent state-of-the-art metrics, especially for low-resource languages. On the other hand, FrugalScore is an approach to learn a fixed, low-cost version of any expensive NLG metric while retaining most of its original performance. Experiments with BERTScore and MoverScore on summarization and translation show that FrugalScore is on par with the original metrics (and sometimes better), while having several orders of magnitude fewer parameters and running several times faster. On average overall learned metrics, tasks, and variants, FrugalScore retains 96.8% of the performance, runs 24 times faster, and has 35 times fewer parameters than the original metrics.

# RÉSUMÉ

Ces dernières années, le domaine de la génération du langage naturel (GLN) a radicalement changé. Ce changement, qui peut être en partie attribué à l'avancée notable du matériel, a conduit les récents efforts du GLN à se concentrer sur des méthodes basées sur les données tirant parti de grands réseaux de neurones pré-entraînés. Cependant, ces progrès ont donné lieu à de nouveaux défis liés aux exigences de calcul, à l'accessibilité et aux stratégies d'évaluation, pour n'en nommer que quelques-uns. Dans cette thèse, nous nous intéressons principalement à contribuer aux efforts visant à atténuer ces défis.

Pour remédier au manque de modèles génératifs monolingues pour certaines langues, nous commençons par présenter *BARThez* et *AraBART*, les premiers modèles seq2seq pré-entraînés à grande échelle pour le Français et l'Arabe, respectivement. Basés sur BART, ces modèles sont particulièrement bien adaptés aux tâches génératives. Nous évaluons BARThez sur cinq tâches discriminantes du benchmark FLUE et deux tâches génératives d'un nouvel ensemble de données de résumé, OrangeSum, que nous avons créé pour cette recherche. Nous montrons que BARThez est très compétitif avec les modèles de langue française basés sur BERT tels que CamemBERT et FlauBERT. Nous poursuivons également le pré-entraînement d'un BART multilingue sur le corpus de BARThez, et montrons que notre modèle résultant, mBARThez, améliore considérablement les performances génératives de BARThez. D'autre part, nous montrons qu'AraBART obtient les meilleures performances sur plusieurs ensembles de données de résumé abstractif, surpassant des bases de référence solides.

Enfin, nous nous concentrons sur l'évaluation des systèmes GLN en proposant *DATScore* et *FrugalScore*. DATScore utilise des techniques d'augmentation des données pour améliorer l'évaluation de la traduction automatique et d'autres tâches GLN. Notre principale conclusion est que l'introduction de traductions enrichies de données des textes source et de référence est très utile pour évaluer la qualité de la traduction générée. Nous proposons également deux nouvelles stratégies de calcul de la moyenne des scores et de pondération des termes pour améliorer le processus original de calcul des scores de BARTScore. Les résultats expérimentaux sur WMT montrent que DATScore est mieux corrélé avec les méta-évaluations humaines que les autres métriques récentes de l'état de l'art, en particulier pour les langues à faibles ressources. D'autre part, FrugalScore est une approche pour apprendre une version fixe et peu coûteuse de toute métrique GLN coûteuse tout en conservant la plupart de ses performances d'origine. Des expériences avec BERTScore et MoverS-

core sur sur le résumé et la traduction montrent que FrugalScore est comparable avec les métriques d'origine (et parfois mieux), tout en ayant plusieurs ordres de grandeur de moins de paramètres et en s'exécutant plusieurs fois plus rapidement. En moyenne, sur l'ensemble des métriques, tâches et variantes apprises, FrugalScore conserve 96,8% des performances, s'exécute 24 fois plus rapidement et comporte 35 fois moins de paramètres que les métriques d'origine.

Dans ce qui suis on resume nos contibutions dans cette thèse :

1. **Modèles pré-entraînés de séquence à séquence.** Contrairement aux modèles basés sur BERT, de nombreuses langues, y compris les langues à haute ressource, manquent encore de modèles pré-entraînés de séquence à séquence. Cependant, ces modèles ont montré des performances remarquables dans les tâches de GLN qui ne peuvent pas être réalisées en utilisant des modèles basés sur BERT. Dans ce contexte, nous apportons les contributions suivantes :
   - Nous avons proposé les premiers modèles pré-entraînés de séquence à séquence basés sur BART pour le français et l'arabe. Ces modèles, nommés BARThez et AraBART, respectivement, sont des auto-encodeurs pré-entraînés pour reconstruire un texte corrompu. Deux fonctions de bruit sont utilisées pour corrompre le texte d'entrée : *text infilling* et *sentence permutation*. BARThez et AraBART ont été pré-entraînés sur un grand corpus pendant 60 heures en utilisant 128 GPU NVidia V100.
   - Pour évaluer notre modèle, nous avons collecté un ensemble de données de résumé abstrait - OrangeSum, un équivalent français de XSUM (Narayan, Cohen et Lapata, 2018a).
   - Nous avons évalué automatiquement nos modèles proposés par rapport à des baselines solides, y compris des modèles avec des capacités plus élevées, et montré qu'ils les surpassaient dans la plupart des configurations.
   - En plus de l'évaluation automatique, nous avons effectué une évaluation manuelle en utilisant le *BEST-Worst Scaling*. Cette évaluation était similaire à l'évaluation automatique tout en mettant en évidence une marge plus importante entre nos modèles et les références.
   - Nous rendons publics nos modèles et notre ensemble de données afin que la communauté NLP puisse les utiliser dans des recherches futures.
2. **DATScore : métrique d'évaluation de la GLN.** Avec les récents progrès dans le domaine de la GLN, les métriques standard pour l'évaluation des systèmes sont devenues moins efficaces. De nouvelles métriques qui correspondent mieux au jugement humain sont proposées. Notre contribution à ce domaine est répertoriée dans ce qui suit :
   - Nous avons proposé DATScore, une métrique automatique pour l'évaluation de la des systèmes de traduction automatique. DATScore est obtenu

en agrégeant huit probabilités conditionnelles centrées sur l'hypothèse : La probabilité de générer l'hypothèse étant donné : la source, la référence, une traduction de la source et une traduction de la référence. en plus de, la probabilité de générer, la source, la référence, une traduction de la source et une traduction de la référence étant donné l'hypothèse. Pour calculer les différentes probabilités des directions, nous utilisons un modèle pré-entraîné multilingue de traduction automatique. Dans notre travail, nous capitalisons sur le modèle M2M-100 (Fan et al., 2021).

- Nous avons introduit une nouvelle méthode one-vs-rest pour calculer la moyenne des scores pour différentes directions de génération avec des poids différents.
- Nous avons proposé un nouveau schéma basé sur l'entropie pour pondérer les termes cibles afin que les tokens plus informatifs reçoivent plus d'importance.
- Nous menons une étude d'ablation confirmant la contribution positive des différentes directions, de la méthode d'agrégation one-vs-rest et du schéma de pondération basé sur l'entropie à la performance globale.
- Nous montrons que notre métrique surpasse des baselines solides, y compris BERTScore et BARTScore, surtout sur les langues à faible ressource. Notre évaluation est effectuée sur WMT17 et WMT18.
- Une implémentation de notre métrique proposée est rendue publique.

3. **FrugalScore : une approche de distillation pour les métriques de la GLN.** Malgré la meilleure performance garantie par les récentes métriques de la GLN, ces métriques souffrent d'une grande complexité due à la grande taille de leurs modèles sous-jacents. Cette grande complexité implique des exigences importantes en termes de temps d'exécution et de mémoire, ce qui peut entraver le progrès des expériences d'évaluation. Dans ce contexte, nous avons proposé ce qui suit :
   - Nous présentons FrugalScore, une approche de distillation pour apprendre une version fixe et peu coûteuse des métriques coûteuses de la GLN tout en conservant la plupart de leurs performances. FrugalScore exploite des modèles compacts pré-entraînés en les entraînant sur un ensemble de données synthétique annoté avec la métrique coûteuse à apprendre.
   - Pour construire l'ensemble de données synthétique, nous avons proposé trois sources différentes : *Résumé abstractif*, *Rétrotraduction* et *Débruitage BART*. Dans notre travail, nous montrons la contribution positive de chaque source à la performance finale.

— Nous évaluons notre approche sur les jeux de données WMT20 et TAC (2008-2011) en utilisant deux métriques : BERTScore et MoverScore. L'évaluation montre que notre approche conserve la plupart des performances (et dans certains cas dépasse) des métriques originales tout en réduisant significativement le temps d'exécution et les besoins en mémoire.

— Notre code et nos modèles entraînés sont rendus publics.

# PUBLICATIONS

The following publications are included in parts or in an extended version in this thesis:

Kamal Eddine, Moussa, Antoine Tixier, and Michalis Vazirgiannis (Nov. 2021). « BARThez: a Skilled Pretrained French Sequence-to-Sequence Model. » In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 9369–9390. DOI: 10.18653/v1/2021.emnlp-main.740. URL: https://aclanthology.org/2021.emnlp-main.740.

Kamal Eddine, Moussa, Guokan Shang, Antoine Tixier, and Michalis Vazirgiannis (May 2022a). « FrugalScore: Learning Cheaper, Lighter and Faster Evaluation Metrics for Automatic Text Generation. » In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 1305–1318. DOI: 10.18653/v1/2022.acl-long.93. URL: https://aclanthology.org/2022.acl-long.93.

Kamal Eddine, Moussa, Guokan Shang, and Michalis Vazirgiannis (2022). « DATScore: Evaluating Translation with Data Augmented Translations. » In: *arXiv preprint arXiv:2210.06576*.

Kamal Eddine, Moussa, Nadi Tomeh, Nizar Habash, Joseph Le Roux, and Michalis Vazirgiannis (2022b). « AraBART: a Pretrained Arabic Sequence-to-Sequence Model for Abstractive Summarization. » In: *WANLP 2022*.

---

Other contributions during the preparation of this dissertation:

Abdine, Hadi, Christos Xypolopoulos, Moussa Kamal Eddine, and Michalis Vazirgiannis (2021). « Evaluation Of Word Embeddings From Large-Scale French Web Content. » In: *ArXiv* abs/2105.01990.

Abdine, Hadi, Moussa Kamal Eddine, Michalis Vazirgiannis, and Davide Buscaldi (2022). « Word Sense Induction with Hierarchical Clustering and Mutual Information Maximization. » In: *arXiv preprint arXiv:2210.05422*.

Gehrmann, Sebastian, Abhik Bhattacharjee, Abinaya Mahendiran, Alex Wang, Alexandros Papangelis, Aman Madaan, Angelina McMillan-Major, Anna Shvets, Ashish Upadhyay, Bingsheng Yao, et al. (2022). « GEMv2: Multilingual NLG Benchmarking in a Single Line of Code. » In: *arXiv preprint arXiv:2206.11249*.

Guo, Yanzhu, Chloé Clavel, Moussa Kamal Eddine, and Michalis Vazirgiannis (2022). « Questioning the Validity of Summarization Datasets and Improving Their Factual Consistency. » In: *EMNLP*.

# ACKNOWLEDGMENTS

First, I would like to thank my supervisor, Prof. **Michalis Vazirgiannis**, who opened the door for me to integrate the LIX laboratory and join his great team DaSciM. He chose me to conduct this Ph.D. and ensured the best environment and resources to work on this dissertation.

Secondly, I would like to thank the two reviewers, Prof. **Eduard Hovy** and Prof. **Eric Gaussier** who kindly accepted to read my dissertation and to deliver a report evaluating my work.

Furthermore, I want to thank all the great jury members who kindly accepted to be in my Ph.D. defense committee. These members are: Prof. **Nizar Habash**, Prof. **Jie Tang**, Prof. **Ioana Manolescu** and Prof. **Alexandros Potamianos**.

Then I would like to thank all my colleagues at DaSciM, especially my two senior fellows, **Antoine Tixier** and **Guokan Shang**. Antoine and Guokan generously guided me at the beginning of my Ph.D. by delivering important tips, and handed me some of their expertise by co-authoring several papers. With them, I was able to realize the beauty of research after months of fighting research windmills.

In addition, I am grateful to the two professors from the NLP Arabic community: Prof. **Nizar Habash** and Prof. **Nadi Tomeh**, with whom I had the privilege to take my first steps in the world of Arabic NLP by collaborating on the AraBART project.

I will not forget to thank my two friends: **Hadi Abdine** and **Ahmad Chamma**, who were next to me during the whole period of my Ph.D. Hadi and Ahmad are examples of what we call "true friends" in a time and place where a true friend is considered science fiction.

Finally, my most enormous thanks go to my beloved family: my **father**, **mother**, **brothers**, **sister**, and **wife**. These people are the most beautiful blessing I have had, and I will ever have. I would fairly say that without their moral support, I could have surrendered without completing this dissertation. I found them by my side at every turn, encouraging and helping me rise again.

Moussa KAMAL EDDINE
Palaiseau, October 2022

# CONTENTS


1 Introduction 1
- 1.1 NLG Challenges . . . 1
  - 1.1.1 Computational Requirements . . . 1
  - 1.1.2 Language inequity . . . 4
  - 1.1.3 Automatic Evaluation . . . 5
- 1.2 Thesis statement . . . 7
- 1.3 Summary of contributions . . . 8
  - 1.3.1 Pretraind sequence-to-sequence models . . . 8
  - 1.3.2 DATScore: NLG evaluation metric . . . 8
  - 1.3.3 FrugalScore: a distillation approach for NLG metrics . . . 9
- 1.4 Software and libraries . . . 10
- 1.5 Outline of the thesis . . . 10

2 Preliminaries 11
- 2.1 Transformers . . . 11
- 2.2 Transfer Learning . . . 14
- 2.3 NLG Systems Evaluation . . . 17
  - 2.3.1 NLG Metrics Examples . . . 19
  - 2.3.2 Evaluating NLG Evaluation Metrics . . . 21

3 Pretrained Sequence-to-Sequence Models 23
- 3.1 Introduction . . . 23
- 3.2 Related work . . . 25
- 3.3 BARThez and AraBART . . . 27
  - 3.3.1 Architecture . . . 27
  - 3.3.2 Vocabulary . . . 28
  - 3.3.3 Self-supervised learning . . . 28
  - 3.3.4 Pretraining corpus . . . 28
  - 3.3.5 Training details . . . 29
- 3.4 mBARThez . . . 30
- 3.5 OrangeSum . . . 32
- 3.6 BARThez Experiments . . . 34
  - 3.6.1 Summarization . . . 34
  - 3.6.2 Discriminative tasks . . . 38
- 3.7 AraBART Experiments . . . 39
  - 3.7.1 Datasets . . . 39
  - 3.7.2 Baselines . . . 40

3.7.3 Training and Evaluation . . . 41
3.7.4 Results . . . 41
3.8 Human Evaluation . . . 43
3.8.1 Quality Evaluation . . . 44
3.8.2 Faithfulness Evaluation . . . 44
3.8.3 AraBART vs AraT5 . . . 45
3.9 Conclusion . . . 46
4 FrugalScore 47
4.1 Introduction . . . 47
4.2 Background . . . 48
4.2.1 Unsupervised metrics . . . 48
4.2.2 Supervised metrics . . . 50
4.2.3 Knowledge distillation . . . 51
4.2.4 Differences with BLEURT . . . 51
4.3 Our approach . . . 52
4.3.1 Synthetic dataset . . . 52
4.3.2 Metric learning . . . 54
4.4 Experiments . . . 56
4.5 Results . . . 56
4.6 Fine-tuning on human annotations . . . 59
4.7 Impact of data sources . . . 59
4.8 BEAMetrics . . . 61
4.9 Conclusion . . . 61
4.10 Acknowledgments . . . 62
5 DATScore 63
5.1 Introduction . . . 63
5.2 Related work . . . 64
5.2.1 Translation evaluation metrics . . . 64
5.2.2 Data augmentation . . . 66
5.3 DATScore . . . 67
5.4 Experiments . . . 69
5.4.1 Experimental settings . . . 69
5.4.2 Main results . . . 72
5.5 Other NLG tasks . . . 73
5.6 Ablation study . . . 75
5.7 Conclusion . . . 78
6 Conclusion 81
6.1 Summary of Contributions . . . 81
6.2 Epilogue . . . 82

Bibliography 83

I Appendix

A Appendix Apresents some examples of the output of the various 107
A.1 OrangeSum Examples . . . . . 107
B Appendix B 120
B.1 Detailed TAC evaluation per year . . . . . 120
B.2 Detailed WMT evaluation per language . . . . . 121
B.3 Detailed TAC evaluation per year (system level) . . . . . 122
B.4 Detailed WMT evaluation per language (system level) . . . . . 123
B.5 Correlation with learned metric (TAC) . . . . . 124
B.6 Correlation with learned metric (WMT) . . . . . 125

# LIST OF FIGURES


Figure 1.1 The evolution of the number of pretrained language models' parameters since 2018. . . . . 2

Figure 1.2 The variation of the correlation of BERTScore with the *Pyramid Score* (Harnly et al., 2005) with respect to the capacity of the underlying model. The used models are variations of BERT with increasing sizes. Between parentheses are the number of parameters of each model. . . . . 7

Figure 4.1 Relative improvement in Pearson correlation compared to a dataset covering all sources. Left: TAC. Right: WMT. . . . . 60

Figure 5.1 Dashed arrows denote the generation directions covered by BARTScore. Solid black arrows indicate our newly introduced directions for calculating DATScore of the example *hypothesis* in English ($\text{Hypo}_{en}$). $\text{Trans1}_{xx}$ and $\text{Trans2}_{yy}$ represent data *augmented translations* in any languages xx and yy, obtained by applying a translation model (grey arrows) to the example *source* in French ($\text{Src}_{fr}$) and example *reference* in English ($\text{Ref}_{en}$), respectively. . . . . 67

Figure 5.2 (a): The horizontal bars represent the Kendall correlations of **each individual generation direction**. (b): The horizontal bar represents the Kendall correlation of **a variant of DATScore with excluding the single generation direction** of the line. Both in (a) and (b), the dashed vertical lines represent the Kendall correlation of the vanilla and **complete DATScore**. Correlation results of `to-English` (in green) and `from-English` (in red) cases are calculated w.r.p human judgments, and averaged over all languages pairs. Experiments are conducted on WMT18. . . . . 76


# LIST OF TABLES


Table 3.1 BARThez pretraining corpus breakdown (sizes in GB, after cleaning). . . . . 29

Table 3.2 Sizes (column 2) are given in thousands of documents. Document and summary lengths are in words. Vocab sizes are in thousands of tokens. . . . . 30
Table 3.3 Degree of abstractivity of OrangeSum compared with that of other datasets, as reported in Narayan, Cohen, and Lapata (2018b). It can be observed that XSum and OrangeSum are more abstractive than traditional summarization datasets. . 30
Table 3.4 Doc 19233 from OrangeSum's test set, and associated summaries. Incorrect information in orange. C2C stands for CamemBERT2CamemBERT. . . . . 31
Table 3.5 Summary of the models used in our experiments. Parameters are given in millions, vocab sizes in thousands, and corpus sizes in GB. C2C stands for CamemBERT2CamemBERT. . . . 33
Table 3.6 Results on OrangeSum. The two BertScore scores are with-/without rescaling (Zhang et al., 2019). . . . . 34
Table 3.7 Proportion of novel n-grams in the generated summaries. C2C stands for CamemBERT2CamemBERT. Note that C2C's high scores are misleading as many of the introduced words are irrelevant. . . . . 35
Table 3.8 Summary statistics. . . . . 35
Table 3.9 Human evaluation using Best-Worst Scaling. . . . . 36
Table 3.10 Accuracy on discriminative tasks. We report the average accuracy over 3 runs, with standard deviation as subscript. † are taken from Le et al. (2019). . . . . 36
Table 3.11 Statistics of Gigaword subsets, as well as XL-Sum summaries (XL-S) and titles (XL-T). The first two columns show the average document and summary lengths. The last three columns show the percentage of n-grams in the summary that do not occur in the input article, used here as a measure of abstractiveness (Narayan, Cohen, and Lapata, 2018a). . . . . 40
Table 3.12 The performance of AraBART, mBART25, $\text{mT5}_{base}$, $\text{AraT5}_{base}$, and C2C (CAMeLBERT2CAMeLBERT) on all datasets in terms of ROUGE-1 (R1), ROUGE-2 (R2), ROUGE-L (RL) and BERTScore (BS). Macro averages are computed over all datasets. . . . . 42
Table 3.13 Human evaluation using Best-Worst Scaling (BWS). The numbers in the first five columns represent the percentage of the times the *row* model was chosen as better than the *column* model. The BWS score is the percentage of time the model's summary was chosen as best minus the percentage of time it was chosen as worst. . . . . 43

Table 3.14 Faithfulness results in terms of the average number of unfaithful spans of text in summaries (less is more faithful), and the percentage of faithful words in summaries (higher is more faithful). . . . . 45
Table 4.1 Scores are summary-level (TAC) and segment-level (WMT) Pearson correlations averaged over 2008 to 2011 for TAC (pyramid score/responsiveness) and over all source languages for WMT-2019. Runtimes include preprocessing. Subscripts refer to row labels and indicate which metric-model combination was used to annotate pairs (e.g., for FrugalScore$_d$, it is row $d$, i.e., BERTScore-BERT-Base). . . . . 55
Table 4.2 Summary-level Pearson correlations with human judgments (Pyramid scores), averaged over 3 runs (standard deviation as subscript). Rows correspond to the training sets and columns to the test sets. . . . . 58
Table 4.3 **Correctness dimension**: Pearson coefficient (computed given a single human reference) between automatic metrics and human judgement for Correctness on the 10 human evaluation datasets. Results in top bloc were taken from the BEAMetrics paper. . . . . 60
Table 4.4 **Non Correctness dimensions**: Pearson coefficient (computed given a single human reference) between automatic metrics and human judgement for the dimensions other than Correctness. Flu, Sim, Rel, Coh, Obv, and Pos denote fluency, simplicity, relevance, obviousness, and possibility, respectively. Results in top bloc were taken from the BEAMetrics paper. . . . . 61
Table 5.1 Absolute Pearson correlation ($|r|$) for `to-English` and Kendall correlations ($\tau$) for `from-English` with segment-level human scores on WMT17. BB stands of Bert-Base, RL for RoBERTa-Large and BL for BART-Large. . . . . 70
Table 5.2 Kendall correlations ($\tau$) for `to-English` and `from-English` with segment-level human scores on WMT18. BB stands of Bert-Base, RL for RoBERTa-Large and BL for BART-Large. . . 71
Table 5.3 Pearson correlation results on WebNLG dataset. . . . . 73
Table 5.4 Pearson correlation results on two summarization datasets: REALSumm and SummEval. . . . . 74
Table 5.5 Pearson correlation Results on two Image Captioning datasets: Flickr8K and PASCAL-50S. . . . . 75

Table 5.6 The average Kendall correlation (to/from)-English when the entropy-based and one-vs-rest weighting are included or excluded. Experiments are conducted on WMT18. . . . . 77
Table A.1 C2C stands for CamemBERT2CamemBERT. OrangeSum document 12158. . . . . 108
Table A.2 C2C stands for CamemBERT2CamemBERT. OrangeSum document 33555. . . . . 109
Table A.3 C2C stands for CamemBERT2CamemBERT. OrangeSum document 25148. . . . . 110
Table A.4 C2C stands for CamemBERT2CamemBERT. OrangeSum document 34657. . . . . 111
Table A.5 C2C stands for CamemBERT2CamemBERT. OrangeSum document 22208. . . . . 112
Table A.6 C2C stands for CamemBERT2CamemBERT. OrangeSum document 22077. . . . . 113
Table A.7 C2C stands for CamemBERT2CamemBERT. OrangeSum document 22168. . . . . 114
Table A.8 C2C stands for CamemBERT2CamemBERT. OrangeSum document 22423. . . . . 115
Table A.9 C2C stands for CamemBERT2CamemBERT. OrangeSum document 19233. . . . . 116
Table A.10 C2C stands for CamemBERT2CamemBERT. OrangeSum document 22060. . . . . 117
Table B.1 Summary-level Pearson correlation (pyramid score/responsiveness). . . . . 120
Table B.2 Segment-level Pearson correlation. . . . . 121
Table B.3 System-level Pearson correlation (pyramid/responsiveness). 122
Table B.4 System-level Pearson correlation. . . . . 123
Table B.5 Summary-level Pearson correlation between the FrugalScore$_{d,e,f,g}$ and the metrics $d, e, f, g$ used to generate the annotations . . 124
Table B.6 Segment-level Pearson correlation between the FrugalScore$_{d,e,f,g}$ and the metrics $d, e, f, g$ used to generate the annotations. . . 125

# 1 INTRODUCTION

Natural language generation (NLG) is the subfield of artificial intelligence and computational linguistics that is concerned with the construction of computer systems that can produce understandable texts in English or other human languages from some underlying non-linguistic representation of information (Reiter and Dale, 1997). In their survey, Gatt and Krahmer (2018) mention that the field of NLG has changed drastically in the last 15 years. The popularity of statistical methods, based mainly on machine and deep learning techniques, significantly shifted the approaches to building NLG systems. In the last five years, another major shift has taken place. Combining inductive transfer learning with the *Transformer* (Vaswani et al., 2017), a recently proposed deep neural network architecture, led to setting a new state of the art on many NLP tasks. Leveraging pretrained Transformers is today the standard practice in almost all NLP tasks, including NLG tasks. Making the pretrained models public helped the research community to build on top of them by proposing new models initialized partially or entirely with the parameters of the released checkpoints (Rothe, Narayan, and Severyn, 2020a). However, despite this remarkable progress in the field, many challenges that keep arising are still to be targeted. These challenges include, but not limited to, computational requirements, languages inequity, and automatic evaluation. We go into further detail about these challenges in the section that follows.

## 1.1 NLG CHALLENGES

### 1.1.1 *Computational Requirements*

In their work, (Sevilla et al., 2022) divide machine learning compute's history into three eras: *Pre Deep Learning Era*, *Deep Learning Era*, and the *Large-Scale Era*. Between 2015 and 2016, the Large-Scale Era started, exceeding the previous era by two orders of magnitudes in terms of parameters, with compute requirements doubling every 9.9 months. Particularly in NLP, we have seen the release of large language models with an impressive number of parameters in the past two years. For example, in 2020, GPT-3 (Brown et al., 2020) was released, with one of its variants having 175B parameters. To understand how big and compute-consuming GPT-3 is, we compare it to RoBERTa-Large (Liu et al., 2019). While the latter has 355M parameters and was pretrained during ~3 GPU-years, Li (2020) estimates that the pretraining time of the

former is 355 GPU-years. Figure 1.1 illustrates the exponentially increasing number of pretrained language models' parameters over time. According to this figure, we can estimate that between the beginning of 2018 and mid-2022, the number of released models' parameters doubled every ~2.5 months. These trends have serious implications that we discuss briefly in the following.

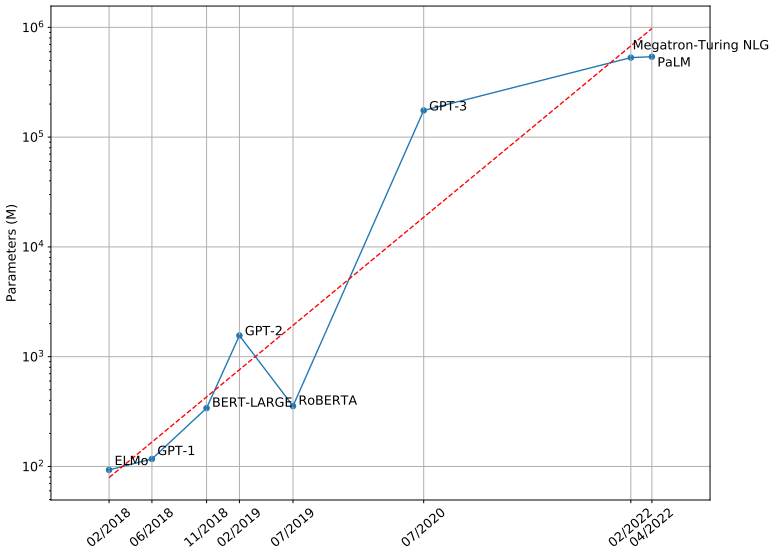


Figure 1.1 – The evolution of the number of pretrained language models' parameters since 2018.

*Technical and Financial Implications*

In the Large-Scale Era, setting a new state-of-the-art became closely related to a higher model's capacity and larger pretraining corpora. However, although the increasing number of parameters across the years allowed for more accurate models, this improvement comes at the cost of increasing "computational burden". In their paper, Thompson et al. (2020) discuss the economic and technical limits of the modern deep learning computational requirements trends. For example, they extrapolate the current NLP trend, and they estimate that a projection from a polynomial model would cost $10^{12}$ USD to achieve an error rate of 2% on the SQuAD 1.1 (Rajpurkar et al., 2016) dataset, and $10^{15}$ USD to achieve an error rate of 1%. If we project from an exponential model, this cost becomes $10^{34}$ USD and $10^{71}$ USD, respectively. This extrapolation makes it clear that continuing in the same trend and achieving the considered theoretical targets would not be technically and financially possible. Nevertheless, to be able to achieve significant improvement in the coming years, the

NLP community should focus on proposing more efficient hardware, algorithms and methods, which will become an inescapable research direction.

*Environmental Implications*

Many papers have recently discussed the effects of exponentially increasing computational requirements on the environment (Shang et al., 2019; Schwartz et al., 2020; Patterson et al., 2021). Obviously, similar to the number of parameters, the energy consumption to train language models was increasing exponentially, doubling every few months (Schwartz et al., 2020). To picture this effect, Patterson et al. (2021) compare the carbon emission of a direct round trip of a passenger jet between San Francisco and New York to GPT-3 pretraining and estimate that the latter emits three times more carbon than the former. Alerted by the significantly increasing environmental cost, Strubell, Ganesh, and McCallum (2019) urge NLP researchers in the industry and academia to promote computationally efficient algorithms and hardware. In addition, Schwartz et al. (2020) propose to adopt efficiency as an evaluation criterion, in parallel to the accuracy and other performance estimation metrics.

*Equity Implications*

By equity, we refer to equal opportunities among researchers and NLP practitioners to access computational resources. Before 2010 (Pre Deep Learning Era), it was typically possible for researchers to run state-of-the-art models on their personal computers. For example, text classification at that time was performed using standard machine learning algorithms like support vector machine (Joachims, 1998), naive Bayes (Eyheramendy, Lewis, and Madigan, 2003) and logistic regression (Genkin, Lewis, and Madigan, 2007). An efficient implementation would make running these approaches on a personal computer possible. However, this situation has changed with the transition to the deep learning era and the emergence of models having millions of trainable parameters. Data parallelization and replacing CPUs with GPUs became a necessity. Unfortunately, a significant percentage of NLP researchers do not have access to such facilities due to their limited budgets. A practical manifestation of this outcome is that most, if not all, of the state-of-the-art models on which the modern NLP capitalizes were developed and released by the industry and not the academia. This limitation would stifle creativity as researchers with promising ideas might be unable to execute them due to their limited access to the required compute (Strubell, Ganesh, and McCallum, 2019).

### 1.1.2 *Language inequity*

Today, although most of the new NLP technologies are promoted as language-agnostic, their applicability and benefits are not equally distributed across languages (Joshi et al., 2020). It is common, for example, that when a new model is introduced, it is first tested and validated on English data. It was historically the case for most of the efforts that contributed to the advancement of the NLP field (Mikolov et al., 2013b; Pennington, Socher, and Manning, 2014b; Devlin et al., 2018; Peters et al., 2018a; Radford et al., 2018; Lewis et al., 2019). Theoretically, these advancements can be applied to any language; this is why they are considered language-agnostic. However, due to the lack of resources and datasets in some cases and the computational requirements in other cases, it is not always possible to take advantage of the proposed technologies. One solution to this problem was to propose multilingual models (Conneau et al., 2019; Lample and Conneau, 2019; Liu et al., 2020b) to perform cross-lingual and monolingual tasks for languages covered by the pretraining corpora. However, multilingual models only partially solved the problem of language inequity. First, these models usually do not cover more than ~100 languages, while there are approximately 7000 languages across the world (Joshi et al., 2020). Second, these models are less performing compared to their monolingual counterparts. Let us consider BART (Lewis et al., 2019) as an illustrative example. The BART model was first proposed in 2019 after the great success of BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) in solving Natural language understanding (NLU) tasks. In contrast to the BERT-like models, which consist of a pretrained encoder, BART is sequence-to-sequence, having an encoder and a decoder jointly pretrained. Thanks to its architecture, BART was efficiently applied to NLG tasks and set a new state-of-the-art on abstractive summarization. However, BART is pretrained on an English corpus, and thus other languages could not take advantage of its success. In 2020, a multilingual BART (mBART-25) (Liu et al., 2020a) was released, covering 25 languages only. Later in 2021, BARThez (Kamal Eddine, Tixier, and Vazirgiannis, 2021) and AraBART (Kamal Eddine et al., 2022b) were introduced for French and Arabic Languages, respectively. It was possible to fine-tune mBART-25 on monolingual datasets covered by the 25 pretraining languages. However, Kamal Eddine, Tixier, and Vazirgiannis (2021) showed that mBART-25 would underperform compared to its monolingual French counterpart. To our knowledge, no other BART-based models were proposed for other languages at the time of writing. Besides, Kamal Eddine, Tixier, and Vazirgiannis (2021) mention in their paper that to evaluate their French BART, BARThez, they had to collect a novel French abstractive summarization dataset themselves because of the unavailability of such a dataset at the time of the pretraining. In this context, the problem of languages inequity can be summarized as follows:

— At the time of writing, it is not possible for models other than English, French, Arabic, and the 25 languages covered by the multilingual model to take advantage of the success of the BART approach.

— The performance on the downstream tasks is not optimal for the languages covered by the multilingual model.

— Applying the BART approach to a new language corpus is computationally expensive.

— Many languages lack specific tasks' datasets, which halts proper evaluation.

In light of the above, it becomes clear that the current trend in NLP makes it practically challenging to apply the proposed approaches equally across languages.

### 1.1.3 *Automatic Evaluation*

The ideal way to evaluate NLG systems is to provide human annotators with a sample of the output generated by each system. Then, following some guidelines, the annotators are asked to score or rank these outputs. However, the annotation process is usually time-consuming and, in many cases, requires the annotators to have a good knowledge of the task the systems are performing (e.g., abstractive summarization). In addition, human intervention is not possible for validation and monitoring. Such assessment can severely impede the field's progress by serving as a bottleneck (Sai, Mohankumar, and Khapra, 2022). For example, if a user is fine-tuning her model's hyper-parameters by performing a grid search, she has to compare the output's quality in each setting. In this case, a human evaluation could complicate the experiments' progress. Given the aforementioned reasons, the development of robust and accurate NLG automatic evaluation metrics has become a necessity.

The first proposed metrics dating back to the beginning of the 2000s were reference-based metrics that rely on the surface-form similarity between the reference and the system-generated sentences. BLEU (Papineni et al., 2002), for example, is a metric commonly used for evaluating machine translation (MT) systems. BLEU is a precision measure that computes an $n$-gram matching between the system's output and the reference. METEOR (Banerjee and Lavie, 2005), another MT evaluation metric, is an F-score based metric, allowing for a more relaxed matching, based on three forms: extract unigram, stemmed word, and synonyms. On the other hand, ROUGE (Lin, 2004a) is a metric commonly used to assess extractive and abstractive summarization systems. Similarly to BLEU, ROUGE uses surface-form matching to compute a recall similarity measure between two given sentences. These metrics, along with their variations, have been used solely in the last two decades. However, with the NLG field's advancement, surface-form metrics have become less reliable.

Peyrard (2019) highlighted this idea by showing that these metrics, that behave similarly, strongly disagree in the higher-scoring range, which suggests that at least some of them are significantly deviating from human judgment. In other words, the better the performance of the evaluated systems is, the less these metrics are reliable. This challenge pushed the community to produce new metric evaluation datasets involving better-performing systems across the years. For example, the workshop on machine translation (WMT) yearly releases since 2008 (Callison-Burch et al., 2008) a new dataset for evaluating MT evaluation metrics (a.k.a. meta-evaluation datasets). On the other hand, since 2014, there has been a rapid surge in the number of new NLG evaluation metrics (Sai, Mohankumar, and Khapra, 2022). Usually, the primary purpose behind these efforts is to provide a solution that better correlates with human judgment than standard metrics, especially in the higher-scoring range. One of the earliest attempts to overcome the limitations of surface-form matching was Word Mover's Distance (WMD) (Kusner et al., 2015). WMD leverages the *word2vec* (Mikolov et al., 2013b) embeddings to compute a dissimilarity score between two documents. This dissimilarity is obtained by solving the transportation problem, i.e., the minimum cost to travel from one document embedding sequence to the other. However, despite the usage of continuous vector representations, *word2vec* embeddings are static, and thus cannot account for the context and word orders. As in other NLP tasks, this issue was mitigated by capitalizing on large pretrained language models (Devlin et al., 2018; Lewis et al., 2019; Liu et al., 2019) that can produce contextual embeddings, i.e., a token representation depends on its context. BERTScore (Zhang et al., 2020), for example, leverages pretrained language models to compute a similarity score using a greedy matching between the tokens' embeddings. MoverScore (Zhao et al., 2019b), is yet another metric combining contextual embedding with a similarity score leveraging Word Mover's Distance. An important advantage of metrics leveraging pretrained models is that they are modular since the underlying model can be changed according to the requirements. This flexibility offers the possibility to improve performance by implementing a more robust model, which mitigates the problem of decreased performance in the higher-scoring range. Figure 1.2 illustrates the improvement of BERTScore's correlation with human judgment with respect to the underlying models capacity.

All the metrics mentioned earlier belong to the family of unsupervised reference-based metrics, i.e., they require the existence of a reference sentence, with no need for training on a corpus of annotated pairs. Other families of NLG metrics exist; we mention mainly reference-based supervised metrics. In addition to the reference sentences, the supervised metric is trained on a dataset of pairs of sentences annotated with a similarity score reflecting the quality of one sentence given the other. Finally, reference-less NLG metrics exist. These metrics do not require the existence of references and can be beneficial in the case of unsupervised NLG. Usually, reference-less

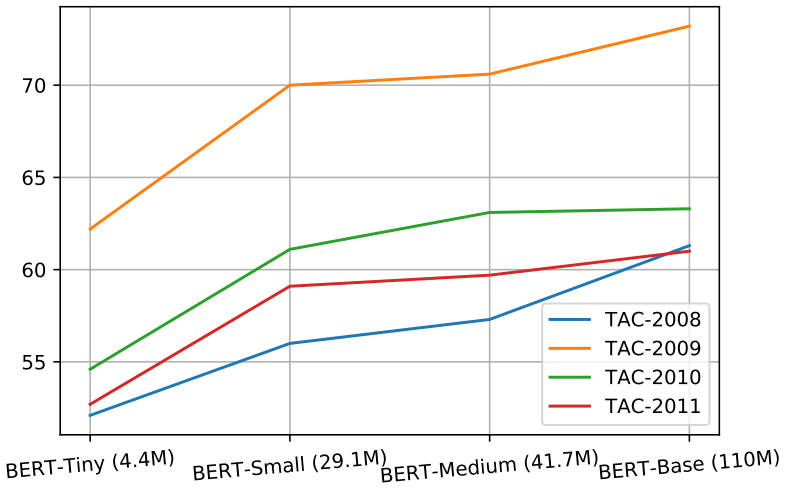


Figure 1.2 – The variation of the correlation of BERTScore with the *Pyramid Score* (Harnly et al., 2005) with respect to the capacity of the underlying model. The used models are variations of BERT with increasing sizes. Between parentheses are the number of parameters of each model.

metrics capitalize on the source fed at the input of the generating system to estimate the quality of the generation. For example, in the case of an abstractive summarization task, a reference-less metric will directly compare the summary to the input document to estimate its quality.

Unlike NLU, NLG systems' automatic evaluation is still an open problem, and many efforts are currently focused on tackling this challenge.

## 1.2 THESIS STATEMENT

In this dissertation, we tackle some of the challenges related to NLG field. In our work, we contribute new models, approaches, and metrics; each mitigating some of the negative or challenging aspects of the current NLG paradigm discussed in Section 1.1. In particular, we contributed the following:

— The first two Seq2Seq models for French (BARThez) and Arabic (AraBART) respectively. These two models, based on BART, are particularly adapted to monolingual generative tasks and set new state-of-the-art on abstractive summarization tasks.

— DATScore, a metric for machine translation (MT) systems evaluation that leverages data-augmented translations to better correlate with human annotations.

— FrugalScore, a distillation approach to learning a fixed, low-cost version of any expensive NLG reference-based metric while retaining most of its original performance.

## 1.3 SUMMARY OF CONTRIBUTIONS

### 1.3.1 *Pretraind sequence-to-sequence models*

Unlike BERT-based models, many languages, including high-resource languages, still lack pretrained sequence-to-sequence models. However these models have shown remarkable performances in NLG tasks that cannot be achieved using BERT-based models. In this context, we contribute the following:

— We proposed the first French and Arabic BART-based pretrained sequence-to-sequence models. These models, named BARThez and AraBART, respectively, are auto-encoders pretrained to reconstruct a corrupted text. Two noise functions are used to corrupt the input text: *text infilling* and *sentence permutation*. BARThez and AraBART were pretrained on a large corpus during 60 hours using 128 NVidia V100 GPUs.

— We proposed OrangeSum, a French abstractive summarization set. At the time of this work, no abstractive summarization dataset existed for French. To evaluate our model, we collected an abstractive summarization - OrangeSum, a French equivalent to XSUM (Narayan, Cohen, and Lapata, 2018a).

— We automatically evaluated our proposed models against strong baselines, including models with higher capacities, and showed that they outperformed them in most configurations.

— In addition to the automatic evaluation, we carried out a manual evaluation using *BEST-Worst Scaling*. This evaluation was in line with the automatic one while highlighting a higher margin between our models and the baselines.

— We publicly release our models and dataset so the NLP community can use them in future research.

### 1.3.2 *DATScore: NLG evaluation metric*

With the recent advancements in the NLG field, the standard metrics for system evaluation have become less efficient. New metrics that better correlate with human judgment are proposed. Our contribution to this area is listed in the following:

— We proposed DATScore, an automatic metric for MT evaluation. DATScore is obtained by aggregating eight conditional probabilities centered on the hypothesis: The probability of generating the hypothesis given: the source, the

reference, a translation of the source, and a translation of the reference. In addition, the probability of generating, the source, the reference, a translation of the source, and a translation of the reference given the hypothesis. To compute the different directions' probabilities we use a pretrained multilingual machine translation model. In our work, we capitalize on the M2M-100 model (Fan et al., 2021).

— We introduced a novel one-vs-rest method to average the scores for different generation directions with different weights.

— We proposed a novel entropy-based scheme for weighting the target terms so that higher informative tokens receive more importance.

— We conduct an ablation study confirming the positive contribution of the different directions, the one-vs-rest averaging method, and the entropy-based weighting scheme to the overall performance.

— We show that our metric outperforms strong baselines, including BERTScore and BARTScore, especially on low-resource languages. Our evaluation is carried out on WMT17 and WMT18.

— An implementation of our proposed metric is publicly released.

### 1.3.3 *FrugalScore: a distillation approach for NLG metrics*

Despite the better performance guaranteed by the recent NLG metrics, these metrics suffer from a high complexity due to the large size of their underlying models. This high complexity implicates important runtime and high memory requirements, which can impede the progress of the evaluation experiments. In this context, we proposed the following:

— We introduce FrugalScore, a distillation approach to learning a fixed and low-cost version of expensive NLG metrics while retraining most of their performance. FrugalScore leverages pretrained compact models by training them on a synthetic dataset annotated with the expensive metric to be learned.

— To construct the synthetic dataset, we proposed three different sources: *Abstractive Summarization*, *BackTranslation* and *BART Denoising*. In our work, we show the positive contribution of each source to the final performance.

— We evaluate our approach on WMT20 and TAC (2008-2011) datasets using two metrics: BERTScore and MoverScore. The evaluation shows that our approach retains most of the performance (and in some cases outperforms) of the original metrics while reducing the runtime and memory requirements significantly.

— Our code and trained models are publicly released.

## 1.4 SOFTWARE AND LIBRARIES

The following are the primary libraries that were used in the context of this thesis:

— `Pytorch` (Paszke et al., 2019). A Python library that performs immediate execution of dynamic tensor computations with automatic differentiation and GPU acceleration.

— `Fairseq` (Ott et al., 2019a). An open-source sequence modeling toolkit that allows researchers and developers to train custom models for translation, summarization, language modeling, and other text generation tasks.

— `Transformers` (Wolf et al., 2019). Is a library dedicated to supporting Transformer-based architectures and facilitating the distribution of pretrained models.

— `Datasets` (Wolf et al., 2019). Datasets is a community library designed to address the challenges of dataset management and access.

— `Pandas` (McKinney et al., 2011). A foundational Python library for data analysis and statistics.

— `Numpy` (Oliphant, 2006; Van Der Walt, Colbert, and Varoquaux, 2011). A fundamental package for scientific computing in Python.

— `Matplotlib` (Hunter, 2007). A library for creating static, animated, and interactive visualizations in Python.

Note that we made all source code and preprocessed data publicly available for reproducibility and for fostering research on the topics covered by this thesis.

## 1.5 OUTLINE OF THE THESIS

The next chapters of this dissertation are organized as follows. In chapter 2 we provide some preliminaries and basic knowledge useful for following the rest of the work. We dedicate chapters 3, 5 and 4 to present our three work contributing to solving some of the challenges faced in the field of NLG. Specifically, in chapter 3 we present our two pretrained seq2seq models BARThez and AraBART, chapter 4 is devoted to our knowledge distillation approach FrugalScore applied to NLG evaluation metrics, and in chapter 5, we present DATScore, our proposed NLG metric. Chapter 6 brings the dissertation to a close and offers some insight into potential future study topics.

# 2

# PRELIMINARIES

In this chapter, we describe some fundamental notions and provide the minimum background information required to follow the rest of the thesis. First, we start with an overview of the recent *Transformer* architecture primarily employed in this work. Then, we present a brief history of *transfer learning* in the context of NLP and discuss its concepts. Finally, we present the NLG metrics on which we capitalize in the following chapters and explain the methodology adopted to evaluate them (i.e., Meta-evaluation).

## 2.1 TRANSFORMERS

A *transformer* (Vaswani et al., 2017) is a recently proposed neural network architecture. It consists of a sequence-to-sequence model based solely on feed-forward networks and attention mechanisms, thus not requiring any recurrent or convolutional layers. Being a sequence-to-sequence model (i.e., autoencoder), a transformer has a bidirectional encoder and an auto-regressive decoder. The encoder takes a sequence of tokens as input and encodes it into some continuous intermediate representation. The decoder, On the other hand, is fed with the intermediate representation and a shifted version of the output and produces a sequence of vectors representing the output. In the following, we provide a simplified mathematical description of the transformer [1]:

### *Scaled Dot-Product Attention*

A transformer implements *Scaled Dot-Product Attention*. Given a sequence of vector representations $y_1, y_2, ..., y_n$, stacked together in a matrix $Y \in \mathbb{R}^{n \times d}$, the Scaled Dot-Product Attention can be written:

$$\text{Att}(Q, K, V) = \text{Softmax}(\frac{QK^T}{\sqrt{d}})V \tag{2.1}$$

Where $Q = YW_Q$, $K = YW_K$, and $V = YW_V$,
$W_Q \in \mathbb{R}^{d \times d}$, $W_K \in \mathbb{R}^{d \times d}$ and $W_V \in \mathbb{R}^{d \times d}$ are trainable parameters.

To Break things down, we can imagine $Q_i$ as a query representing the *ith* token $y_i$ in the sequence. First, $Q_i$ attends to all the sequences of tokens represented by the

1. Note that we omit all biases for simplicity.

key vectors ($K$); this is $Q_i K^T$. The result is scaled by $\sqrt{d}$ [2] and passed to a softmax function to compute an importance score for each of the tokens; this is $softmax(\frac{Q_i K^T}{\sqrt{d_k}})$. Finally, a weighted sum of the Value vectors ($V$) is calculated as the new representation of the token at position $i$; this is $softmax(\frac{Q_i K^T}{\sqrt{d_k}})V$. Stacking all the queries together in one matrix $Q$, we obtain the Eq. 2.1.

*Multi-head attention*

A multi-head attention is simply a multiple realization of the Scaled Dot-Product Attention. Given $h$ heads, we linearly project $Q$, $K$ and $V$, $h$ times and we compute the Scaled Dot-Product Attention $H_i = Att(QW_i{}^Q, KW_i{}^K, VW_i{}^V)$.

The final attention output is obtained by linearly projecting the concatenation of Scaled Dot-Product Attention:

$$\text{MultiAtt}(Q, K, V) = \text{Concat}(H_1, H_2, ..., H_h)W \tag{2.2}$$

where $W_Q \in \mathbb{R}^{d \times d/h}$, $W_K \in \mathbb{R}^{d \times d/h}$, $W_V \in \mathbb{R}^{d \times d/h}$ and $W \in \mathbb{R}^{d \times d}$ are trainable parameters.

*Transformer Encoder Block*

A transformer encoder layer consists mainly of a multi-head attention layer and two feed-forward layers, in addition to layer normalization (LN) and skip connections. First the sequence matrix $Y \in \mathbb{R}^{n \times d}$ is passed to the multi-head attention layer which outputs a new sequence of vectors represented by $Y_1$:

$$Y_1 = \text{MultiAtt}(Y, Y, Y)$$

Note that all the parameters of the MultiAtt comes from the same sequence. This is why the attention mechanism applied in the encoder is called *self-attention*.
$Y_1$ is then fed to a Layer Normalization (LN) and a skip connection is applied to obtain $Y_2 = \text{LN}(Y_1 + Y)$.
$Y_2$ is then passed to two consecutive feed-forward networks with a ReLU non-linearity in between:

$$Y_3 = \text{ReLU}(Y_1 W_1)W_2$$

where $W_1 \in \mathbb{R}^{d \times d'}$ and $W_2 \in \mathbb{R}^{d' \times d}$ are trainable parameters.

Finally, we apply again Layer Normalization (LN) with a skip connection on top of the feed-forward layers to obtain the output of the encoder block:

$$\text{EBOut} = \text{LN}(Y_3 + Y_2)$$

2. The scaling factor is to avoid vanishing gradient effect when $Q_i K^T$ becomes large.

*Transformer Decoder Block*

A transformer decoder block has the same components as an encoder block, with an additional multi-head attention layer following the first one. While the first attention layer perform *self-attention*, the additional one performs *cross-attention* or *encoder-decoder attention*. Again, given a sequence of vector representations $Y \in \mathbb{R}^{n \times d}$, we have $Y_1 = \text{MultiAtt}(Y, Y, Y)$ and $Y_2 = \text{LN}(Y_1 + Y)$. Now instead of passing $Y_2$ directly to the feed-forward layers, we fed them to another multi-head attention layer with queries coming from the encoder output, and the keys and values from the decoder, denoting the encoder's output EOut we have:

$$Y_3 = \text{MultiAtt}(\text{EOut}, Y, Y)$$

Similar to an encoder block, $Y_3$ finally goes into two feed-forward layers followed by a layer normalization with a skip connection.

*Input Representation*

Having a set of tokens $V = \{x_1, x_2, ..., x_k\}$, a sequence of tokens $x_1, x_2, ..., x_n$ is passed to an embedding layer followed by a positional encoding. Formally, having a trainable embedding matrix $W_{emb} \in \mathbb{R}^{d \times k}$, we map each token in the sequence to a vector in the embedding space of dimension $\mathbb{R}^d$.

$$v_i = \text{OneHot}(x_i) W_{emb}$$

In the case of RNNs, the embedded vector sequence is directly passed to the first recurrent layer, however in the case of the transformer, the different employed layers do not take into consideration the order of the sequence. In other words, the vectors in the sequence $v_1, v_2, ..., v_n$ will have the same encoding if the sequence is shuffled. This is why a positional encoding is added on the top of the embedding layer. The positional encoding (PE) injects information about the position of the tokens in the sequence. The input to the transformer can then be written:

$$\text{Input} = v_i + \text{PE}(i)$$

where

$$\text{PE}_{2j}(i) = \sin\left(i/10000^{2j/d}\right)$$
$$\text{PE}_{2j+1}(i) = \cos\left(i/10000^{2j/d}\right)$$

Where $2j$ and $2j + 1$ represents even and odd dimensions respectively.

*Putting Everything Together*

The Transformer consists of $N$ stacked encoder blocks and $N'$ stacked decoder blocks where each block's input is the preceding one's output. The input to the first encoder block is an input sequence mapped to a vector representation using an embedding matrix and positional encoding as described previously. On the other hand, a right-shifted version of the output is fed to the decoder, and a beginning-of-sentence (BOS) special token is appended to the beginning of the sequence. To prevent the decoder from looking at future positions, we use a masking matrix that puts the attention weights of next tokens to zero. In the context of NLG, we add on top of the decoder a feed-forward layer linearly projecting the output to a vector of dimension $k$ representing the generation probabilities of the tokens in the vocabulary set.

## 2.2 TRANSFER LEARNING

In their recently published tutorial, Ruder et al. (2019) define *transfer learning* as a set of methods that extend the *learning in isolation* approach by leveraging data from additional domains or tasks to train a model with better generalization properties. In this section, we will present an overview of the current trend of neural-based unsupervised approaches for transfer learning in the context of NLP.

One of the earliest attempts to take advantage of unannotated corpora to boost the performance on supervised tasks dates back to 2008, when Collobert and Weston (2008) showed that *semi-supervised learning* (i.e., using part of the input as a target) can improve the generalization capabilities of a CNN model. Given a set of sentences $\mathcal{S}$ extracted from the English Wikipedia corpus, they train a language model to predict whether the middle word in a sentence is replaced by a random word or not using the following loss function:

$$L = \sum_{s \in \mathcal{S}} \sum_{w} \max(0, 1 - \mathrm{LM}(s^{+}) + \mathrm{LM}(s^{-w}))$$

Where LM refers to the language model, $s^{+}$ is a positive example, and $s^{-w}$ is the negative example where the middle word is replaced with $w$. An important breakthrough came in 2013 with *word2vec* and its two variants, continuous Bag-of-Words (CBOW) and continuous Skip-gram (Mikolov et al., 2013a,b). In the former, a neural network is trained by predicting the middle word of a sentence given its context. While in the latter, a neural network is trained by predicting the context of a given word. As a side effect of this training, an embedding matrix with randomly initialized weights is trained, and its vectors can be transferred to other tasks as continuous representations of words. For example, in the case of CBOW, we consider all

the sets of windows $\mathcal{W}$ of size $k$ in a given corpus, obtained by sliding a window over the whole corpus. For each window $w$, we denote the middle word $t^w$ and the context $c^w = (c_1^w, ..., c_{k-1}^w)$. The goal is to maximize the probability $p(t^w|c^w)$, which is estimated with a shallow NN with only two trainable matrices $W_{in} \in \mathbb{R}^{|V|\times d}$ and $W_{out} \in \mathbb{R}^{d\times|V|}$.

$$\hat{p}(t^w|c^w) = -\text{OneHot}(t^w)(\text{softmax}(\sum_{i=1}^{k-1}(\text{OneHot}(c_i^w)W_{in})W_{out}))$$

$|V|$ is the vocabulary size, and $d$ is the dimension of the embedding space. Finally, the NN is trained by minimizing the Negative log-likelihood loss of the estimated prediction:

$$L = -\sum_{w} \log\hat{p}(t^w|c^w)$$

$W_{in}$ and $W_{out}$ being trained can be used in other tasks by initializing the NN weights with their entries instead of random values. In contrast to CBOW, Skip-gram trains the NN by predicting the context given the target word. Otherwise, the two approaches are similar.

Inspired by the success of word2vec, other approaches started to emerge, such as GloVe (Pennington, Socher, and Manning, 2014a) and fastText (Bojanowski et al., 2017). However, the common limitation between all these word embeddings pretraining approaches is that they are *static*. Being static means that the vector representation of a given word will be the same regardless of its context. For example, the embedding of the word *bank* has several senses, some related to finance and others to geography. Using static embedding, these senses that can be inferred from the context will be represented by the same vector. To mitigate this problem, recent efforts proposed training models that can produce contextual embeddings. These models represent the current trend of NLP unsupervised (or semi-supervised) transfer learning. In the following, we briefly present some of the popular and widely adopted models to produce contextual embeddings; these are ELMo (Peters et al., 2018a), GPT (Radford et al., 2018), BERT (Devlin et al., 2019), and BART (Lewis et al., 2019).

*ELMo*

ELMo, which stands for *Embeddings from Language Models*, is the earliest attempt to train contextual embeddings. ELMo trains a multi-layer bidirectional LSTM on the language modeling objective, that is, given a sequence of tokens $x_1, x_2, ..., x_n$, estimating the probability of the next token given the previous ones $p(x_i|x_{i'<i})$. This probability is modeled by an LSTM that produces at each position $i$ a representation vector $h_i$ (hidden state). Adding a feed-forward network followed by a softmax

on top of $h_i$ the NN can estimate the probability of each token in the vocabulary. ELMo uses a bidirectional LSTM that jointly models the forward and the backward contexts. We denote the hidden states produced by the forward and the backward LSTMs $h_i^f$ and $h_i^b$. The objective of ELMo is then to maximize:

$$\sum_i (\log p(x_i|x_{i'<i}; \theta^f) + \log p(x_i|x_{i'>i}; \theta^b))$$

Where $\theta^f$ and $\theta^b$ are the trainable parameters of the forward and backward LSTMs, respectively. Once the network is trained, the contextual embeddings are obtained by feeding the target sentence to the model and extracting the internal vector representations of the bidirectional LSTM. A simple choice is to consider the hidden representations produced by the top LSTM layers. In this case, the vector representation of the word at position i would be $h_i^f|h_i^b$ where | is the concatenation operator.

*GPT*

GPT stands for *Generative Pre-trained Transformer*. Similar to ELMo, GPT trains a deep neural network to maximize the language modeling objective with three main differences:

— GPT only models the left-to-right context.

— GPT uses a transformer-based architecture instead of LSTMs.

— ELMo uses a feature-based approach; that is, it extracts features from a pretrained model and injects them in a separate model finetuned on the downstream task. In contrast, GPT uses a finetuning approach where the whole pretrained model is finetuned on the downstream task by adding a layer (e.g., classification head) on top of it to adapt it to the task.

Note that due to the autoregressive nature of the language modeling task, GPT only uses the decoder [3] part of the transformer.

*BERT*

BERT stands for *Bidirectional Encoder Representations from Transformers*. In contrast to ELMo and GPT, BERT pretrains a deep neural network by **jointly** conditioning on the left and right context. BERT, which uses the encoder part of the transformer, is pretrained on two unsupervised tasks: *MASKED LM* and *NEXT SENTENCE PREDICTION*. In the former, a number of tokens in the input sentence are replaced with a special [MASK] token, and the model is trained to predict them. Formally, given

3. The mutli-head attention sub-layers performing cross-attention are excluded because of the absence of the encoder.

a set of sentences $\mathcal{S} = \{s_1, s_2, ..., s_k\}$ where $s_i = (w_1^i, w_2^i, ..., w_n^i)$, for each sequence a subset of the positions $\mathcal{J}_i$ is randomly chosen, and a special token replaces their corresponding tokens. We denote the modified sequences $s_i^m$. The model is pretrained to maximize the log-likelihood of predicting the true tokens:

$$\sum_{i=0}^{k} \sum_{j \in \mathcal{J}_i} \log p(w_j^i | s_i^m)$$

On the other hand, the next sentence prediction is a binary classification task where the model learns to predict if two sentences are the actual consecutive sentences in the original corpus. It was shown later with RoBERTa (Liu et al., 2019) that the next sentence prediction task does not improve the performance of the pretrained model on the downstream tasks, and thus it was excluded from the pretraining in later efforts.

Similar to GPT, BERT follows a finetuning approach where the whole model parameters are finetuned on the downstream task by adding a simple layer on top of it.

#### *BART*

BART stands for *Bidirectional Auto-Regressive Transformers*. The most important advantage of BART, compared to BERT and GPT, is that it jointly pretrains the encoder and the decoder of the transformer. In other words, BART is an auto-encoder, particularly a denoising auto-encoder pretrained by learning to reconstruct a corrupted text and minimizing the negative log-likelihood between the predicted text and the original one. Formally, having a noising function $n$ and a set of documents $\mathcal{D} = \{d_1, d_2, ..., d_k\}$, the model learns to maximize the likelihood of the original documents given the noised ones:

$$\sum_{i=1}^{k} \log(d_i | n(d_i))$$

Thanks to its bidirectional encoder and autoregressive decoder, BART can perform conditional NLG tasks (e.g., abstractive summarization), which cannot be straightforwardly performed by BERT or GPT. In chapter 3 we provide more details about BART's pretraining and finetuning.

## 2.3 NLG SYSTEMS EVALUATION

Automatic evaluation metrics are today a key component in the progress of the machine learning field. A standard practice to evaluate a model is to divide the considered dataset into three splits: train, validation, and test. The train split is the one

the model is trained on to tune its parameters. (e.g., through backpropagation and SGD in the case of a NN). The validation split is used to monitor the progress of the model and for hyper-parameters selection. For example, having a binary classification model, it is possible to monitor the performance evolution of the model across the training steps by computing the model's accuracy on the validation split each $n$ steps. In addition, to choose the best hyper-parameters configuration, we choose the one with the best validation accuracy. Once the configuration is chosen and the training is done, the final model's score is computed by running the model on the test set, which is used to compare the performance of multiple models. The main difference between the test set and the validation set is that no information from the training is leaked to the model when evaluating on the test set, while it is the case when using the validation set to choose the *best training configuration*.

Fortunately, the evaluation of NLU systems is straightforward, as there are many intuitive and efficient metrics that can serve this purpose. For example, to evaluate a model's performance on a balanced binary classification dataset we can simply choose the *accuracy* metric. The accuracy metric can be calculated as follows:

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

Where:

— $TP$ stands for true positive: the number of examples classified correctly as positives.

— $TN$ stands for true negative: the number of examples classified correctly as negatives.

— $FP$ stands for false positive: the number of examples classified incorrectly as positives.

— $FN$ stands for false negative: the number of examples classified incorrectly as negatives.

In the case of an unbalanced binary dataset, the accuracy metric could be misleading. For example, if we have a dataset with 80% of positive examples and 20% of negative examples, a dummy model that always predicts positive classes will have an accuracy of 80%, which is significantly higher than a random model (i.e., a model with uniformly random predictions) that would have an accuracy of ~50%. In this case, it is recommended to use other metrics such as precision, recall, and F-score that can be respectively expressed as:

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN} \qquad Fscore = \frac{2 \times P \times R}{P + R}$$

The precision is interpreted as *among all the positively predicted labels, how many are truly positive?*, the recall as *among all the truly positive labels, how many are positively predicted?*. Finally, the F-score is the geometric mean of the precision and the recall.

Unfortunately, the robustness and simplicity of metrics used in NLU tasks are not guaranteed in the case of NLG tasks. This is mainly due to the nature of NLG systems outputs which consists of sequences rather than real values.

A common way to evaluate NLG systems is to compare their outputs to *reference* sentences and compute a similarity score between them. In the following we provide an overview of the reference-based metrics used in the next chapters.

### 2.3.1 *NLG Metrics Examples*

#### *ROUGE*

In the original paper (Lin, 2004a) ROUGE was proposed as a modified recall measure. ROUGE is based on the proportion of $n$-gram overlap between the system generated sentence and one or more reference sentences. Given a set $\mathcal{S}$ of reference sentences, and a generated sentence $g$, ROUGE-$n$ Recall is computed as:

$$\text{ROUGE}-n = \frac{\sum_{s \in \mathcal{S}} \sum_{gram_n \in s} C_{match}(gram_n, s, g)}{\sum_{s \in \mathcal{S}} \sum_{gram_n \in s} C(gram_n, s)} \tag{2.3}$$

where $n$ is the length of $n$-gram, $C(x, y)$ is the number of occurrences and $C_{match}(x, y, z)$ is the maximum number of co-occurrences of $x$ in $y$ and $z$. ROUGE-$n$ Precision is similar to the ROUGE-$n$ Recall after replacing the denominator in equation 2.3 with the total number of tokens in the generated sentence instead of the reference sentences (i.e., $|S| \sum_{gram_n \in g} C(gram_n, g)$).

Another variant of ROUGE is ROUGE-L which considers the Longest Common Subsequence between the reference and the generated sentence. Thus ROUGE-L Recall and Precision can be written:

$$\text{ROUGE-L}_{\text{Recall}} = \frac{\sum_{s \in \mathcal{S}} \text{LCS}(s, g)}{\sum_{s \in \mathcal{S}} |s|} \qquad \text{ROUGE-L}_{\text{Precision}} = \frac{\sum_{s \in \mathcal{S}} \text{LCS}(s, g)}{\sum_{s \in \mathcal{S}} |g|}$$

A common practice to evaluate abstractive and extractive summarization systems, is to report ROUGE-1, ROUGE-2 and ROUGE-L F-scores.

#### *BERTScore and MoverScore*

The recently introduced BERTScore (Zhang et al., 2019) and MoverScore (Zhao et al., 2019b) are general-purpose NLG evaluation metrics that are becoming widely used. The main difference between BERTScore and MoverScore lies in the function used to compute the similarity between the representations of the two sequences $\mathbf{x} = \langle x_1, ..., x_k \rangle$ and $\mathbf{y} = \langle y_1, ..., y_l \rangle$. In chapter 4 we experiment with these two metrics, so we provide more details about them. BERTScore first computes the pairwise

cosine similarity between the representations of the tokens in each sequence, and uses greedy matching to match each token to the most similar one in the other sequence. Given two pre-normalized vector sequences $\mathbf{x}$ and $\mathbf{y}$, BERTScore computes:

$$R_{BERT} = \frac{1}{|\mathbf{x}|} \sum_{x_i \in \mathbf{x}} \max_{y_j \in \mathbf{y}} x_i^T y_j \tag{2.4}$$

and:

$$P_{BERT} = \frac{1}{|\mathbf{y}|} \sum_{y_i \in \mathbf{y}} \max_{x_j \in \mathbf{x}} y_i^T x_j \tag{2.5}$$

The F1-score is classically obtained as:

$$F_{BERT} = 2\frac{P_{BERT} R_{BERT}}{P_{BERT} + R_{BERT}} \tag{2.6}$$

On the other hand, MoverScore uses an $n$-gram generalization of the Word Mover's Distance (WMD) (Kusner et al., 2015) as their (dis)similarity function. More specifically, they solve for the optimal transportation flow matrix $F \in \mathbb{R}^{|\mathbf{x}| \times |\mathbf{y}|}$ between the two weighted sequences of $n$-grams:

$$WMD(\mathbf{x}, \mathbf{y}) = min_F \langle C, F \rangle \tag{2.7}$$
$$s.t. \quad F\mathbf{1} = f_{\mathbf{x}}, \quad F^T \mathbf{1} = f_{\mathbf{y}}$$

Where $C$ is the transportation cost matrix ($C_{ij}$ is the Euclidean distance between $x_i$ and $y_j$) and $f_x \in \mathbb{R}_+^{|\mathbf{x}|}$ and $f_y \in \mathbb{R}_+^{|\mathbf{y}|}$ are the $n$-gram weight vectors.

*BARTScore*

In contrast to BERTScore and MoverScore, BARTScore (Yuan, Neubig, and Liu, 2021) does not compute a similarity score by directly comparing the generated and the reference sentence. Instead, BARTScore capitalizes on pretrained sequence-to-sequence models, which are pretrained to estimate the conditional probability of an output sequence given an input. With a proper pretraining objective, we can use this probability as an evaluation score by placing the generated and the reference sentences at the input and the output of the model respectively and vice-versa. We consider two sequences $\mathbf{x} = \langle x_1, ..., x_k \rangle$ and $\mathbf{y} = \langle y_1, ..., y_l \rangle$, where $\mathbf{x}$ is the sequence to evaluate and $\mathbf{y}$ the reference. We compute a precision and recall scores as follows:

$$BARTScore_P = \sum_{t=1}^{k} w_t \log P(x_t | \mathrm{y}, \{x_{t'}\}_{t'=1}^{t-1}; \theta)$$

$$BARTScore_R = \sum_{t=1}^{l} w_t \log P(y_t | x, \{y_{t'}\}_{t'=1}^{t-1}; \theta)$$

### 2.3.2 *Evaluating NLG Evaluation Metrics*

The ideal way to evaluate NLG metrics (i.e., Meta-Evaluation) is to collect human annotations scoring the metrics' scores. However, collecting annotations for every emerging metric would not be feasible given the growing number of proposed metrics every year. This is why the community adopted an automatic approach to perform the evaluation. This approach consists of collecting human scores evaluating the generated sentences of a given dataset, then computing a correlation measure between these annotations and the scores generated by the automatic metric. For example, given a translation dataset and several machine translation models with a varying performance, we ask human annotators to assess the quality of the models based on some criteria and we compute the correlation between the scores of the metrics to be evaluated and the human assessment.

The correlation measures mostly used by the community are: *Pearson correlation*, *Kendall's Tau correlation* and *Kendall's Tau-like correlation* that we present in the following:

*Pearson correlation*

Given the human scores $x_1, ..., x_n$ and the metrics scores $y_1, ..., y_n$ the Pearson correlation coefficient can be written:

$$\rho(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=0}^{n}(x_i - \frac{1}{n}\sum_{i=1}^{n} x_i)(y_i - \frac{1}{n}\sum_{i=1}^{n} y_i)}{\sqrt{\sum_{i=0}^{n}(x_i - \frac{1}{n}\sum_{i=1}^{n} x_i)^2}\sqrt{\sum_{i=0}^{n}(y_i - \frac{1}{n}\sum_{i=1}^{n} y_i)^2}}$$

The Pearson correlation expresses a linear relation between $\mathbf{x}$ and $\mathbf{y}$.

*Kendall's Tau correlation*

Given the human scores $x_1, ..., x_n$ and the metrics scores $y_1, ..., y_n$ the Kendall's Tau correlation coefficient can be written:

$$\tau = \frac{2}{n(n-1)} \sum_{i<j} \text{sign}(x_i - x_j)\text{sign}(y_i - y_j)$$

We can interpret this equation as, for each pair of positions $i$ and $j$ such as $i < j$ how many times the sort order of $x_i, x_j$ agrees with the one of $y_i, y_j$.

Kendall's Tau correlation measures the monotonic relationship between the two vectors $\mathbf{x}$ and $\mathbf{y}$. In other words, it quantifies the similarity of the rankings of $\mathbf{x}$ and $\mathbf{y}$.

*Kendall's Tau-like correlation*

In some cases, the human annotators are asked to compare two systems' output instead of scoring each output solely. For instance, for each pair of generated sentences for the same example, the annotators choose whether the first one is better or the second. Kendall's Tau-like correlation coefficient can be written:

$$\tau' = \frac{|Concordant| - |Discordant|}{|Concordant| + |Discordant|}$$

where $|Concordant|$ is the number of pairs on which the metric agrees with the human relative ranking, and $|Discordant|$ is the number of examples where they disagree.

# 3

# PRETRAINED SEQUENCE-TO-SEQUENCE MODELS

Inductive transfer learning has taken the entire NLP field by storm, with models such as BERT and BART setting new state of the art on countless NLU tasks. However, most of the available models and research have been conducted for English. In this chapter, we introduce BARThez and AraBART, the first large-scale pretrained seq2seq models for French and Arabic respectively. Being based on BART, these models are particularly well-suited for generative tasks. We evaluate BARThez on five discriminative tasks from the FLUE benchmark and two generative tasks from a novel summarization dataset, OrangeSum, that we created for this research. We show BARThez to be very competitive with state-of-the-art BERT-based French language models such as CamemBERT and FlauBERT. We also continue the pretraining of a multilingual BART on BARThez' corpus, and show our resulting model, mBARThez, to significantly boost BARThez' generative performance. On the other hand, We show that AraBART achieves the best performance on multiple abstractive summarization datasets, outperforming strong baselines including a pretrained Arabic BERT-based model, multilingual BART, Arabic T5, and a multilingual T5 model. Code [1] [2], data [3] and models [4] are publicly available, along with a web interface [5].

## 3.1 INTRODUCTION

Inductive transfer learning, that is, solving tasks with models that have been pretrained on very large amounts of data, was a game changer in computer vision (Krizhevsky, Sutskever, and Hinton, 2012). In NLP, while annotated data are scarce, raw text is virtually unlimited and readily available. It thus emerged that the ability to learn good representations from plain text could greatly improve general natural language understanding.

Trained on gigantic amounts of raw data and with hundreds of GPUs, models based on the Transformer architecture (Vaswani et al., 2017), such as GPT (Radford et al., 2018) and BERT (Devlin et al., 2018), have set new state-of-the-art performance in every NLU task. Moreover, users around the world can easily benefit

1. https://github.com/moussaKam/BARThez
2. https://github.com/moussaKam/arabart
3. https://github.com/moussaKam/OrangeSum
4. https://huggingface.co/moussaKam
5. http://nlp.polytechnique.fr/

from these improvements, by finetuning the publicly available pretrained models to their specific applications. This also saves considerable amounts of time, resources and energy, compared with training models from scratch.

BART (Lewis et al., 2019) combined a BERT-like bidirectional encoder with a GPT-like forward decoder, and pretrained this seq2seq architecture as a denoising autoencoder with a more general formulation of the masked language modeling objectives of BERT. Since not only BART's encoder but also its decoder is pretrained, BART excels on tasks involving text generation.

While the aforementioned efforts have made great strides, most of the research and resources were dedicated to the English language, despite a few notable exceptions. In this chapter, we partly address this limitation by contributing BARThez [6] and AraBART the first pretrained seq2seq model for French and Arabic respectively.

BARThez and AraBART, based on BART, were pretrained on a very large monolingual corpora from past research that we adapted to suit BART's specific perturbation schemes. Unlike already existing BERT-based language models such as CamemBERT (Martin et al., 2019) and AraBERT (Antoun, Baly, and Hajj, 2020), our seq2seq models are particularly well-suited for generative tasks. We evaluate BARThez and AraBART on two generative tasks (title and headline generation) and we show that our models set a new state-of-the-art on summarization tasks. In addition, we evaluate BARThez on five sentiment analysis, paraphrase identification, and natural language inference tasks from the recent FLUE benchmark, and we show that BARThez is very competitive with CamemBERT, FlauBERT (Le et al., 2019), and mBART (Liu et al., 2020a). We also continue the pretraining of an already pretrained multilingual BART on BARThez's corpus. Our resulting model, mBARThez, significantly boosts BARThez' performance on generative tasks.

Our contributions are as follows:

• We publicly release the first large-scale pretrained seq2seq models dedicated to the French and Arabic languages. BARThez and AraBART, featuring 140M parameters, and trained on 100 GB of text for 60 hours with 128 GPUs. We evaluate BARThez and AraBART on abstractive summarization tasks, with automated and human evaluation, and show that our models are very competitive with the state of the art.

• To address the lack of generative tasks in the existing FLUE benchmark, we put together a novel dataset for summarization in French, OrangeSum, that we publicly release [7] and analyze in this paper. OrangeSum is more abstractive than traditional summarization datasets, and can be considered the French equivalent of XSum (Narayan, Cohen, and Lapata, 2018b).

6. named after a legendary French goalkeeper, Fabien Barthez: https://en.wikipedia.org/wiki/Fabien_Barthez
7. https://github.com/Tixierae/OrangeSum

• We continue the pretraining of a multilingual BART on BARThez' corpus, and show that our resulting model, named mBARThez, offers a significant boost over BARThez on generative tasks.
• We publicly release our code and models [8]. Our models were also integrated into the highly-popular Hugging Face `Transformers` library [9]. As such, they can easily be distributed and deployed for research or production within a standard, industrial-strength framework. They also have their own APIs and can be interactively tested online.

## 3.2 RELATED WORK

Learning without labels is enabled via self-supervised learning [10], a setting in which a system learns to predict part of its input from other parts of its input. In practice, one or more supervised tasks are created from the unlabeled data, and the model learns to solve these tasks with custom objectives.

Some of the earliest and most famous self-supervised representation learning approaches in NLP are word2vec (Mikolov et al., 2013b), GloVe (Pennington, Socher, and Manning, 2014a) and FastText (Bojanowski et al., 2017). While these methods were significant advancements, they produce static representations, which is a major limitation, as words have different meanings depending on the unique contexts in which they are used.

**Deep pretrained language models**. ELMo (Peters et al., 2018b) provided the first contextualized embeddings, by extracting and combining the internal states of a pretrained deep bi-LSTM language model. Except for the word embeddings and the softmax layer, the forwards and backwards RNNs have different parameters. The authors of ELMo showed that the learned representations could be transferred with great benefits to downstream architectures, to solve a variety of supervised NLU tasks.

Beyond simply combining internal states, Howard and Ruder (2018) proposed ULMFiT, a universal transfer learning method for text classification where the language model is pretrained on a large, general dataset, finetuned on a specific dataset, and finally augmented with classification layers trained from scratch on downstream tasks.

With the OpenAI GPT, Radford et al. (2018) capitalized on the Transformer architecture (Vaswani et al., 2017), superior and conceptually simpler than recurrent neural networks. More precisely, they pretrained a left-to-right Transformer decoder as

8. https://github.com/moussaKam/BARThez
9. https://huggingface.co/moussaKam
10. a term coined by Yann LeCun.

a general language model, and finetuned it on 12 language understanding tasks by applying different transformations to the input.

By combining ideas from all the aforementioned models, and introducing bidirectional pretraining, BERT (Devlin et al., 2018) disrupted the NLP field by setting new state-of-the-art performance on 11 NLU tasks, with very wide margins. More precisely, BERT uses a bidirectional Transformer encoder with a masked language model objective, making the learned representations capture both the left and the right contexts, instead of just the left context. The sheer size of BERT, with up to 24 Transformer blocks, plays a role in performance too.

With GPT-2, a version of GPT with over an order of magnitude more parameters than GPT, Radford et al. (2019) showed that as long as they have very large capacities, general language models can reach reasonable performance on many specific NLU tasks out-of-the-box, without any finetuning, i.e., accomplish zero-shot transfer. This demonstrates the fundamental nature and importance of the language modeling objective for inductive transfer learning.

In RoBERTa, Liu et al. (2019) showed that the performance of BERT could be improved by optimizing its hyperparameters and training procedure. The study of why and how BERT works so well has now its own dedicated research field, known as BERTology (Rogers, Kovaleva, and Rumshisky, 2020).

**Languages**. Following the success of BERT for the English language, some BERT models were pretrained and evaluated in other languages. Some examples include Arabic (Antoun, Baly, and Hajj, n.d.), Dutch (Vries et al., 2019; Delobelle, Winters, and Berendt, 2020), French (Le et al., 2019; Martin et al., 2019), Italian (Polignano et al., 2019), Portuguese (Souza, Nogueira, and Lotufo, 2019), Russian (Kuratov and Arkhipov, 2019), and Spanish (Cañete et al., 2020).

In addition to the aforelisted monolingual models, multilingual models were also proposed, notably mBERT (Devlin et al., 2018), XLM (Lample and Conneau, 2019) and XLM-R (Conneau et al., 2019).

**Abstractive summarization**. Abstractive summarization is an important and challenging task, requiring diverse and complex natural language understanding and generation capabilities. A good summarization model needs to read, comprehend, and write well.

GPT-2 can be used for summarization, by sampling a certain numbers of tokens from a given start seed. However, while the generated text is grammatical and fluent, summarization performance is only slightly superior to that of a random extractive baseline.

Being a bidirectional encoder, BERT cannot be used out-of-the-box for language generation, unlike GPT-2. Furthermore, BERT produces single-sentence representations, whereas for summarization, reasoning over multiple sentence and paragraph representations is necessary. Liu and Lapata (2019) proposed a way to over-

come these challenges. At the input level, they introduced special tokens to encode individual sentences, interval segment embeddings, and used more position embeddings than in BERT. Then, they combined a pretrained BERT encoder with a Transformer-based decoder initialized at random and jointly trained the two models with different optimizers and learning rates.

**BART and mBART**. BART (Lewis et al., 2019) is a denoising auto-encoder that jointly pretrains a bidirectional encoder (like in BERT) and a forward decoder (like in GPT) by learning to reconstruct a corrupted input sequence.

Since not only the encoder but also the decoder is pretrained, BART is particularly effective when applied to text generation tasks.

Liu et al. (2020b) pretrained a multilingual BART (mBART) on 25 different languages. They showed that this multilingual pretraining brings significant performance gains on a variety of machine translation tasks. MASS (Song et al., 2019) is another multilingual pretrained sequence to sequence model, that learns to predict a masked span in the input sequence. The main difference between MASS and BART, is that the former only predicts the masked fragment of the sentence, while the latter learns to reconstruct the entire corrupted sentence. This difference makes MASS less effective in discriminative tasks, given that only the masked span is fed to the decoder (Lewis et al., 2019). ProphetNet (Yan et al., 2020) which also adopts the encoder-decoder structure, introduces a new learning objective called future n-gram prediction. This objective reduces overfitting on local correlations by learning to predict the next n-grams (instead of unigrams) at each time step given the previous context.

## 3.3 BARTHEZ AND ARABART

Our models are based on BART (Lewis et al., 2019), a denoising auto-encoder. It consists of a bidirectional encoder and a left-to-right auto-regressive decoder.

### 3.3.1 *Architecture*

We use the BASE architecture, with 6 encoder and 6 decoder layers. We did not opt for a LARGE architecture due to resource limitations. Our BASE architecture uses 768 hidden dimensions and 12 attention heads in both the encoder and the decoder. In total, our models have roughly 140M parameters. The architecture has two differences compared with the vanilla seq2seq Transformer (Vaswani et al., 2017). The first one is the use of GeLUs activation layers instead of ReLUs, and the second is the presence of a normalization layer on top of the encoder and the decoder, fol-

lowing Liu et al. (2020b). These additional layers help stabilizing the training when using FP16 precision.

### 3.3.2 *Vocabulary*

To generate the vocabulary of BARThez and AraBART, we use SentencePiece (Kudo and Richardson, 2018) that implements byte-pair-encoding (BPE) (Sennrich, Haddow, and Birch, 2015). We do not perform any type of pre-tokenization and we fix the size of the vocabulary to 50K sub-words. The SentencePiece model is trained on a 10GB random sample of the pretraining corpus. We fix the character coverage to 99.95%.

### 3.3.3 *Self-supervised learning*

We use the same pretraining as in BART. That is, BARThez and AraBART learn to reconstruct a corrupted input. More precisely, the input text is perturbed with a noise function, and the model has to predict it by minimizing the cross-entropy between the predicted and the original text. Formally, having a set of documents $\{X_1, X_2, ..., X_n\}$ and a noising function $n$, we aim at finding the parameters $\theta$ that minimize:

$$L_\theta = -\sum_i \log P(X_i | n(X_i); \theta)$$

Two different types of noise are applied in $n$. First, we use the *text infilling* scheme, where a number of text spans are sampled and replaced with one [MASK] special token. The length of the spans is sampled from a Poisson distribution with ($\lambda = 3.5$) and 30% of the text is masked. The second perturbation scheme is *sentence permutation*, where the input document, seen as a list of sentences, is shuffled.

Note that here, we follow Lewis et al. (2019), who showed that both text infilling and sentence shuffling were necessary to obtain best results.

### 3.3.4 *Pretraining corpus*

For the French model BARThez, we created a version of FlauBERT's corpus (Le et al., 2019) suitable for the two perturbation schemes described in subsection 4.3.2. Indeed, in the original FlauBERT corpus, each sentence is seen as an independent instance, while in our case, we need instances to correspond to complete documents.

Other than that, BARThez' corpus is similar to FlauBERT's. It primarily consists in the French part of CommonCrawl, NewsCrawl, Wikipedia and other smaller cor-

pora that are listed in Table 3.1. To clean the corpus from noisy examples, we used the script [11] provided by Le et al. (2019). Note that we disabled the Moses tokenizer, as we used SentencePiece which does not require any pre-tokenization. The total corpus size was 66/101GB before/after SentencePiece tokenization.

| **Corpus** | **Size** |
|---|---|
| CommonCrawl | 42.0 |
| NewsCrawl (Li et al., 2019) | 9.6 |
| Wikipedia | 4.0 |
| GIGA (Tiedemann, 2012) | 3.8 |
| ORTOLANG (ATILF and CLLE, 2020) | 2.7 |
| MultiUn (Eisele and Chen, 2010) | 2.2 |
| EU Bookshop (Skadiņš et al., 2014) | 2.1 |

Table 3.1 – BARThez pretraining corpus breakdown (sizes in GB, after cleaning).

For the Arabic model AraBART, we adopt the same corpus used to pretrain AraBERT (Antoun, Baly, and Hajj, 2020). While Antoun, Baly, and Hajj (2020) use a preprocessed version of the corpus, we opted to reverse the preprocessing by using a script that removes added spaces around non alphabetical characters, and also undo some words segmentation. The use of a corpus with no preprocessing, makes the text generation more natural. The size of the pretraining corpus before/after sentencepiece tokenization is 73/96 GB.

### 3.3.5 *Training details*

We pretrained BARThez and AraBART on 128 NVidia V100 GPUs. We fixed the batch size to 6000 tokens per GPU and the update frequency to 2, which gave a total number of roughly 2k documents per update. We used the Adam optimizer (Kingma and Ba, 2015) with a learning rate starting from $6.10^{-4}$ and decreasing linearly as a function of the training step. We used a warm up of 6% of the total number of training steps. Pretraining lasted for approximately 60 hours, allowing for 20 passes over the whole corpus in the case of BARThez and 25 passes in the case of AraBART. In the first $\frac{4}{5}$ of the epochs, we fixed the dropout to 0.1, then we decreased it to 0 in the last $\frac{1}{5}$ of the epochs. All the pretraining experiments were carried out using the `Fairseq` library (Ott et al., 2019b).

11. https://github.com/getalp/Flaubert

| Dataset | Train/val/test | Avg. Doc. length | | Avg. Summ. length | | Vocab size | |
|---|---|---|---|---|---|---|---|
| | | Words | Sentences | Words | Sentences | Docs | Summ. |
| CNN | 90.3/1.2/1.1 | 760.5 | 33.98 | 45.70 | 3.58 | 34 | 89 |
| DailyMail | 197/12.2/10.40 | 653.3 | 29.33 | 54.65 | 3.86 | 564 | 180 |
| NY Times | 590/32.7/32.7 | 800.0 | 35.55 | 45.54 | 2.44 | 1233 | 293 |
| XSum | 204/11.3/11.3 | 431.1 | 19.77 | 23.26 | 1.00 | 399 | 81 |
| OrangeSum Title | 30.6/1.5/1.5 | 315.3 | 10.87 | 11.42 | 1.00 | 483 | 43 |
| OrangeSum Abs. | 21.4/1.5/1.5 | 350 | 12.06 | 32.12 | 1.43 | 420 | 71 |

Table 3.2 – Sizes (column 2) are given in thousands of documents. Document and summary lengths are in words. Vocab sizes are in thousands of tokens.

| | | CNN | DailyMail | NY Times | XSum | OrangeSum Title | OrangeSum Abstract |
|---|---|---|---|---|---|---|---|
| % OF NOVEL N-GRAMS IN GOLD SUMMARY | Unigrams | 16.75 | 17.03 | 22.64 | 35.76 | 26.54 | 30.03 |
| | Bigrams | 54.33 | 53.78 | 55.59 | 83.45 | 66.70 | 67.15 |
| | Trigrams | 72.42 | 72.14 | 71.93 | 95.50 | 84.18 | 81.94 |
| | 4-grams | 80.37 | 80.28 | 80.16 | 98.49 | 91.12 | 88.3 |
| LEAD | ROUGE-1 | 29.15 | 40.68 | 31.85 | 16.30 | 19.84 | 22.21 |
| | ROUGE-2 | 11.13 | 18.36 | 15.86 | 1.61 | 08.11 | 07.00 |
| | ROUGE-L | 25.95 | 37.25 | 23.75 | 11.95 | 16.13 | 15.48 |
| EXT-ORACLE | ROUGE-1 | 50.38 | 55.12 | 52.08 | 29.79 | 31.62 | 38.36 |
| | ROUGE-2 | 28.55 | 30.55 | 31.59 | 8.81 | 17.06 | 20.87 |
| | ROUGE-L | 46.58 | 51.24 | 46.72 | 22.65 | 28.26 | 31.08 |

Table 3.3 – Degree of abstractivity of OrangeSum compared with that of other datasets, as reported in Narayan, Cohen, and Lapata (2018b). It can be observed that XSum and OrangeSum are more abstractive than traditional summarization datasets.

## 3.4 MBARTHEZ

mBART (Liu et al., 2020b) is a multilingual BART. It follows a LARGE architecture, with 12 layers in both the encoder and the decoder, hidden vectors of size 1024, and 16 attention heads. It was trained on a multilingual corpus containing 1369 GB of raw text, for over 2.5 weeks on 256 Nvidia V100 GPUs. The multilingual corpus covers 25 different languages, including 56 GB of French text. In the original paper, the authors evaluated mBART on machine translation. However, mBART can also be used to perform monolingual tasks.

We continued the pretraining of the pretrained mBART on BARThez' corpus (see subsection 3.3.4) for about 30 hours on 128 Nvidia V100 GPUs, which allowed for 4 passes over BARThez' corpus. This can be seen as an instance of *language-adaptive*

| | | |
|---|---|---|
| | Document | Le 18 octobre dernier, Jacline Mouraud se faisait connaître en publiant sur Facebook une vidéo dans laquelle elle poussait un "coup de gueule" contre le gouvernement. Aujourd'hui, la Bretonne a pris ses distances par rapport au mouvement, notamment face à d'autres figures plus radicales comme Éric Drouet. Jacline Mouraud réfléchit désormais à créer son propre parti, "la seule chose envisageable", comme elle l'explique au JDD. Nicolas Sarkozy, "le seul qui a des couilles". Cette figure des "gilets jaunes", accusée de faire le jeu de LREM estime que "le problème" d'Emmanuel Macron "c'est qu'il est jeune". "Il devrait y avoir un âge minimum pour être président : 50 ans", souligne Jacline Mouraud. Dans le JDD, elle raconte d'ailleurs avoir voté blanc lors de la dernière présidentielle. En 2007 et 2012, c'est Nicolas Sarkozy, "le seul qui a des couilles", que la figure des "gilets jaunes" avait soutenu. En attendant de se lancer, pas question pour elle en tous les cas d'être candidate aux européennes sur une liste de La République en marche. |
| Abstract | Gold | L'une des figures du mouvement ne sera toutefois pas candidate aux prochaines élections européennes. |
| | mBART | Jacline Mouraud, figure des "gilets jaunes", estime que le président d'Emmanuel Macron est trop jeune pour être président. |
| | mBARThez | Dans un entretien au JDD, la figure des "gilets jaunes" Jacline Mouraud révèle qu'elle réfléchit à créer son propre parti. |
| | BARThez | Dans les colonnes du JDD, la figure des "gilets jaunes" explique qu'elle envisage de se présenter aux européennes sur une liste La République en marche. |
| | C2C | Retirée de la vie politique depuis plusieurs mois, Bretone Mouraud envisage de se lancer en politique. Et elle réfléchit à quelque chose de plus, rapporte le JDD. |
| Title | Gold | "Gilets jaunes" : Jacline Mouraud réfléchit à créer son parti |
| | mBART | "Gilets jaunes" : Jacline Mouraud lance son propre parti |
| | mBARThez | "Gilets jaunes" : Jacline Mouraud prend ses distances |
| | BARThez | La figure des "gilets jaunes" Jacline Mouraud va créer son propre parti |
| | C2C | "Gilets jaunes" : Jacline Mouraud réfléchit à sa propre candidature |

Table 3.4 – Doc 19233 from OrangeSum's test set, and associated summaries. Incorrect information in orange. C2C stands for CamemBERT2CamemBERT.

*pretraining*, which goes a step further than *domain-adaptive pretraining* (Gururangan et al., 2020). The initial learning rate was set to 0.0001 and linearly decreased towards zero. We call the resulting model mBARThez.

Note that being multilingual, mBART uses a vocabulary containing tokens with non-latin characters. We eliminated these tokens from all embedding layers of mBARThez, reducing its number of parameters from 610M to 458M.

## 3.5 ORANGESUM

BART-based models are particularly well-suited to generative tasks, but unfortunately, FLUE (Le et al., 2019), the French equivalent of GLUE, only contains discriminative tasks [12] (Wang et al., 2018).

We therefore decided to create one such task. We opted for single-document abstractive summarization, as it is a generative task that also requires the model to encode its input very well. In other words, for a model to summarize well, it needs to both read, comprehend, and write well, making abstractive summarization one of the most central and challenging evaluation tasks in NLP.

**Motivation**. Our strategy here was to create a French equivalent of the recently introduced XSum dataset (Narayan, Cohen, and Lapata, 2018b). Unlike the historical summarization datasets, CNN, DailyMail, and NY Times, introduced by Hermann et al. (2015), which favor extractive strategies, XSum requires the models to display a high degree of abstractivity to perform well. XSum was created by scraping articles and their one-sentence summaries from the BBC website, where the one-sentence summaries are not catchy headlines, but rather capture the gist of the articles.

**Data collection**. We adopted an analogous strategy, and scraped the "Orange Actu" website [13]. Orange S.A. is a large French multinational telecommunications corporation, with 266M customers worldwide. Our scraped pages cover almost a decade from Feb 2011 to Sep 2020. They belong to five main categories: France, world, politics, automotive, and society [14]. The society category is itself divided into 8 subcategories: health, environment, people, culture, media, high-tech, unusual ("insolite" in French), and miscellaneous.

We extracted these two fields from each page, thus creating two summarization tasks: OrangeSum Title and OrangeSum Abstract. Gold summaries are respectively 11.42 and 32.12 words in length on average, for these two tasks (see Table 3.2). Note that like in XSum, titles in OrangeSum tend not to be catchy headlines but rather convey the essence of the article. The same can be said about the abstracts.

**Post-processing**. As a post-processing step, we removed all empty articles, and articles whose titles were shorter than 5 words. For OrangeSum Abstract, we removed the top 10% articles in terms of proportion of novel unigrams in the abstracts, as we observed that such abstracts tended to be introductions rather than real abstracts. This corresponded to a threshold of 57% novel unigrams.

12. There is no generative task in GLUE or superGLUE (Wang et al., 2019) either.
13. https://actu.orange.fr/, 'Actu' means News.
14. root URLs are https://actu.orange.fr/ for all categories except https://auto.orange.fr/news/ for automotive.

For both OrangeSum Title and OrangeSum Abstract, we set aside 1500 pairs for testing, 1500 for validation, and used all the remaining ones for training. We make the dataset publicly available [15].

An example document with its summaries is provided in Table 3.4. More examples are available in Appendix A.

**Analysis**. Table 3.2 compares OrangeSum with XSum and the well-known CNN, DailyMail, and NY Times datasets. We can see that the two OrangeSum datasets are very similar to XSum in terms of statistics, but is one order of magnitude smaller than XSum. However, the size of OrangeSum still allows for effective finetuning, as we later demonstrate in our experiments.

Table 3.3 provides empirical evidence showing that like XSum, OrangeSum is less biased towards extractive systems compared with the traditional datasets used for abstractive summarization. There are 30% novel unigrams in the OrangeSum Abstract reference summaries and 26.5% in OrangeSum Title, compared with 35.7% in Xsum, 17% in CNN, 17% in DailyMail, and 23% in NY Times. This indicates that XSum and OrangeSum summaries are more abstractive. These observations are also confirmed by the fact that the two extractive baselines LEAD and EXT-ORACLE perform much more poorly on XSum and OrangeSum than on the other datasets.

| | BASE | | LARGE | | |
|---|---|---|---|---|---|
| | **BART-random** | **BARThez (ours)** | **C2C** | **mBART** | **mBARThez (ours)** |
| **Layers** | 12 | 12 | 24 | 24 | 24 |
| **Params** | 165 | 165 | 274 | 610 | 458 |
| **Vocab. size** | 50 | 50 | 32 | 250 | 101 |
| **Pretraining hours** | 0 | 60 | 24 | 432 | 30 |
| **Pretraining GPUs** | NA | 128 | 256 | 256 | 256 + 128 |
| **Corpus size** | NA | 66 | 138 | 1369 | 1369 + 66 |

Table 3.5 – Summary of the models used in our experiments. Parameters are given in millions, vocab sizes in thousands, and corpus sizes in GB. C2C stands for CamemBERT2CamemBERT.

OrangeSum was later integrated into the prevalent benchmark GEMv2 [16] (Gehrmann et al., 2022), which is a multilingual NLG benchmark supporting 40 documented datasets in 51 languages.

15. https://github.com/Tixierae/OrangeSum
16. https://gem-benchmark.com/data_cards/OrangeSum

## 3.6 BARTHEZ EXPERIMENTS

We compare BARThez and mBARThez with the following models, summarized in Table 3.5.

• **mBART**. The multilingual BART LARGE described in section 3.4.

• **CamemBERT2CamemBERT (C2C)**. To apply CamemBERT to our generative task, we used the BERT2BERT approach proposed by Rothe, Narayan, and Severyn (2020b). More precisely, we fine-tuned a sequence-to-sequence model whose both encoder and decoder parameters were initialized with CamemBERT LARGE weights. The only weights that were initialized randomly were the encoder-decoder attention weights.

• **BART-random**. As an additional baseline, we train a model with the same architecture and vocabulary as BARThez from scratch on the downstream tasks.

| | | Abstract | | | | Title | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | R-1 | R-2 | R-L | BertScore | R-1 | R-2 | R-L | BertScore |
| | LEAD | 22.2 | 7.0 | 15.5 | 14.7/68.0 | 19.8 | 8.1 | 16.1 | 15.8/68.4 |
| | EXT-ORACLE | 38.4 | 20.9 | 31.1 | 29.0/73.4 | 31.6 | 17.1 | 28.3 | 25.1/72.0 |
| BASE | BART-random | 27.7 | 08.2 | 18.5 | 22.5/71.0 | 28.8 | 13.2 | 25.2 | 29.7/73.7 |
| BASE | BARThez (ours) | **31.44** | **12.8** | **22.2** | **27.5/72.8** | **40.9** | **23.7** | **36.0** | **40.6/77.7** |
| LARGE | C2C | 29.2 | 09.8 | 20.0 | 25.5/72.1 | 34.92 | 18.0 | 30.8 | 36.4/76.2 |
| LARGE | mBART | 31.9 | 13.1 | 22.4 | 27.8/72.9 | 40.7 | 23.7 | 36.0 | 40.4/77.7 |
| LARGE | mBARThez (ours) | **32.7** | **13.7** | **23.2** | **28.8/73.3** | **41.1** | **24.1** | **36.4** | **41.4/78.1** |

Table 3.6 – Results on OrangeSum. The two BertScore scores are with/without rescaling (Zhang et al., 2019).

### 3.6.1 *Summarization*

All pretrained models were finetuned for 30 epochs and we used a learning rate that warmed up to 0.0001 (6% of the training steps) and then decreased linearly to 0. BART-random was trained for 60 epochs. We selected the checkpoint associated with the best validation score to generate the test set summaries, using beam-search with a beam size of 4.

We classically report ROUGE-1, ROUGE-2 and ROUGE-L scores (Lin, 2004b) in Table 3.6. However, since ROUGE is limited to capturing n-gram overlap, which is poorly suited to the abstractive summarization setting, we also report BERTScore scores. BERTScore (Zhang et al., 2019) is a recently introduced metric that leverages the contextual representations of the candidate and gold sentences.

| | | Gold | BARThez (ours) | C2C | mBART | mBARThez (ours) |
|---|---|---|---|---|---|---|
| **OrangeSum Abstract** | 1-grams | 30.0 | 10.9 | 39.4 | 13.4 | 15.5 |
| | 2-grams | 67.2 | 34.0 | 79.1 | 38.9 | 43.3 |
| | 3-grams | 81.9 | 48.0 | 92.0 | 53.7 | 58.5 |
| | 4-grams | 88.3 | 56.8 | 96.2 | 62.6 | 67.3 |
| **OrangeSum Title** | 1-grams | 26.5 | 16.7 | 33.8 | 16.9 | 17.8 |
| | 2-grams | 66.7 | 51.7 | 75.7 | 52.3 | 53.4 |
| | 3-grams | 84.2 | 72.0 | 91.8 | 73.1 | 73.4 |
| | 4-grams | 91.1 | 82.5 | 96.7 | 82.7 | 82.9 |

Table 3.7 – Proportion of novel n-grams in the generated summaries. C2C stands for CamemBERT2CamemBERT. Note that C2C's high scores are misleading as many of the introduced words are irrelevant.

| | | Length | Repetitions (%) |
|---|---|---|---|
| ABSTRACT | Gold | 32.12 | 11.47 |
| | mBART | 28.20 | 7.47 |
| | mBARThez | 29.45 | 8.60 |
| | BARThez | 29.10 | 14.47 |
| | C2C | 30.68 | 23.00 |
| TITLE | Gold | 11.42 | 0.93 |
| | mBART | 10.79 | 1.73 |
| | mBARThez | 11.03 | 2.27 |
| | BARThez | 11.19 | 2.73 |
| | C2C | 11.23 | 19.53 |

Table 3.8 – Summary statistics.

Following Narayan, Cohen, and Lapata (2018b), we included two extractive baselines in our evaluation, LEAD and EXT-ORACLE. LEAD creates a summary by extracting the first n sentences from the document. In our case, we set $n = 1$. The second baseline, EXT-ORACLE, extracts from the document the set of sentences that maximizes a specific score. In our case, we extracted the one sentence maximizing ROUGE-L.

**Quantitative results**. Table 3.6 compares the performance of the models finetuned on the summarization task. While having four times less parameters, BARThez is on par with mBART, both in terms of ROUGE and BERTScore. mBARThez provides a significant boost over BARThez and mBART and reaches best performance everywhere. This highlights the importance of adapting a multilingual pretrained

| | System | Score |
|---|---|---|
| | Gold | 14.29 |
| BASE | BARThez (ours) | **21.43** |
| LARGE | CamemBERT2CamemBERT | -75.00 |
| LARGE | mBART | 11.90 |
| LARGE | mBARThez (ours) | **27.38** |

Table 3.9 – Human evaluation using Best-Worst Scaling.

| | | CLS-books | CLS-DVD | CLS-music | PAWSX | XNLI |
|---|---|---|---|---|---|---|
| BASE | mBERT† | 86.15 | 89.90 | 86.65 | 89.30 | 76.9 |
| BASE | $\text{CamemBERT}_{BASE}$† | 92.30 | 93.00 | 94.85 | **90.14** | **81.20** |
| BASE | $\text{FlauBERT}_{BASE}$† | 92.30 | 92.45 | 94.10 | 89.49 | 80.60 |
| BASE | BARThez (ours) | $\mathbf{94.47}_{0.17}$ | $\mathbf{93.17}_{0.40}$ | $\mathbf{94.97}_{0.25}$ | $88.90_{0.24}$ | $80.73_{0.40}$ |
| BASE | BART-random | $76.37_{0.34}$ | $73.20_{0.65}$ | $76.00_{1.28}$ | $55.27_{0.33}$ | $60.43_{0.87}$ |
| LARGE | $\text{Camembert}_{LARGE}$ | $\mathbf{95.47}_{0.33}$ | $\mathbf{95.37}_{0.07}$ | $\mathbf{96.00}_{0.29}$ | $\mathbf{91.83}_{0.54}$ | $\mathbf{85.33}_{0.05}$ |
| LARGE | $\text{Flaubert}^{\dagger}_{LARGE}$ | 95.00 | 94.10 | 95.85 | 89.34 | 83.40 |
| LARGE | mBART | $93.40_{0.22}$ | $93.10_{0.20}$ | $93.13_{0.79}$ | $89.70_{0.22}$ | $81.07_{0.38}$ |
| LARGE | mBARThez (ours) | $94.63_{0.05}$ | $94.03_{0.09}$ | $95.30_{0.16}$ | $90.90_{0.22}$ | $81.87_{0.50}$ |

Table 3.10 – Accuracy on discriminative tasks. We report the average accuracy over 3 runs, with standard deviation as subscript. † are taken from Le et al. (2019).

model to a specific language before finetuning (*language-adaptive pretraining*). This also suggests that, when proper adaptation is conducted, it can be advantageous to capitalize on a multilingual model to perform monolingual downstream tasks, probably because there are some translingual features and patterns to be learned. Finally, all BART-based models outperform CamemBERT2CamemBERT by a significant margin.

**Human evaluation**. To validate our positive quantitative results, we conducted a human evaluation study with 11 French native speakers. They were PhD students from the CS department of our university, working in NLP and other fields of AI. They volunteered after receiving an email announcement.

Following Narayan, Cohen, and Lapata (2018b), we used *Best-Worst Scaling* (Louviere, Flynn, and Marley, 2015). In this approach, two summaries from two different systems, along with their input document, are presented to a human annotator who has to decide which one is *better*. We asked evaluators to base their judgments on three criteria: *accuracy* (does the summary contain accurate facts?), *informativeness* (does the summary capture the important information in the document?) and *fluency* (is the summary written in well-formed French?).

We included the BARThez, mBARThez, mBART and C2C models in our analysis, along with the ground-truth summaries. We randomly sampled 14 documents from the test set of OrangeSum Abstract, and generated all possible summary pairs for each document, resulting in 140 pairs. Each pair was randomly assigned to three different annotators, resulting in 420 evaluation tasks in total. The final score of a model was given as the percentage of time its summary was chosen as *best* minus the percentage of time it was chosen as *worst*. Scores are reported in Table 3.9. mBARThez reaches first place, like for the quantitative results, but with an even wider margin. It is also interesting to note that BARThez, which was on par with mBART quantitatively, significantly outperforms it this time around, in terms of human evaluations. Note that the negative score of CamemBERT2CamemBERT should be analyzed *in comparison* with the other models. That is, C2C's summaries were judged to be worse more often than not.

Suprisingly, BARThez and mBARThez' summaries were often judged better than the ground truth ones. We hypothesize that since the GT summaries are short abstracts written by the authors of the articles, they may be well-written but contain information that is missing from the documents, such as dates. In such situations, the annotators may consider such information as inaccurate (e.g., due to model *hallucinations*) and favor the other model.

**Qualitative results**. As shown in Table 3.7, mBARThez is more abstractive than BARThez and mBART, as measured by the proportion of novel n-grams in the generated summaries. E.g., mBARThez introduces on average 15.48% of novel unigrams in its summaries for the Abstract task, compared with 10.93 and 13.40 for BARThez and mBART, respectively. It is interesting to note that despite this superior abstractivity, mBARThez still reaches first place everywhere in terms of the ROUGE metric, which measures n-gram overlap. We hypothesize that BARThez is less abstractive than mBART and mBARThez due to the fact that it is based on a BASE architecture instead of a LARGE one, and has thus four times less parameters.

Finally, it is also to be noted that CamemBERT2CamemBERT (C2C) introduces many new words, which could be considered a good thing at first. However, it also repeats itself a lot (see Table 3.8) and has low ROUGE, BERTSum, and human evaluation scores. A manual observation revealed that actually, many of the new words introduced by C2C are irrelevant (see Appendix A for summary examples).

Also, like Rothe, Narayan, and Severyn (2020b), we computed the length of the summaries, and the percentage of summaries with at least one non-stopword repetition. We used as stopwords the 500 most frequent words from the system and gold summaries, across all documents. As can be seen in Table 3.8, for both the Abstract and Title tasks, all models generated summaries of sizes very close to that of the Gold summaries.

In terms of repetitions, the less redundant models, closest to the ground truth, are mBART and mBARThez. This is especially apparent on the Abstract task, where potential for repetition is greater. On this task, mBART and mBARThez show less than 9% repetitions, compared with 14.5 and 23 for BARThez and C2C (resp.), and 11.5 in the references. C2C is also way more redundant than the other models and far from the reference on the Title task, with 19.5% repetitions.

### 3.6.2 *Discriminative tasks*

In addition to generative tasks, BART-like models can perform discriminative tasks (Lewis et al., 2019). In the case of sequence classification, the input sequence is fed to both the encoder and the decoder, and the representation of the last token in the sequence is used by adding a classification head on top of it. When the input consists of several sentences, these sentences are separated with a special token and pasted together. We evaluate the different models on five discriminative tasks from the FLUE benchmark [17] (Le et al., 2019), the French equivalent of GLUE (Wang et al., 2018).

• **CLS**. The Cross-lingual Sentiment analysis dataset (Prettenhofer and Stein, 2010) is made of Amazon reviews to be classified as positive or negative. It contains 3 product categories: books, DVD and music. The train and test sets are balanced and contain 2000 examples (each) per product category. Following Le et al. (2019), we used 20% of the train set as validation set.

• **PAWSX**. The Cross-lingual Adversarial Dataset for Paraphrase Identification (Yang et al., 2019) contains pairs of sentences, and the task is to predict whether they are semantically equivalent. There are 49401 examples for training, 1992 for development, and 1985 for testing.

• **XNLI**. The Cross-lingual NLI corpus (Conneau et al., 2018) contains pairs of sentences, and the task is to predict whether the first one (premise) entails the second one (hypothesis), contradicts it, or neither entails nor contradicts it (neutral relationship). 392702 pairs are used for training, 2490 for development, and 5010 for testing.

**Training details**. In all experiments, we finetuned the model for 10 epochs with a learning rate chosen from $\{10^{-4}, 5.10^{-5}, 10^{-5}\}$ based on the best validation score. We repeated each experiment 3 times with different seeds and report the mean and standard deviation.

**Results**. Table 3.10 reports the test set accuracies. For comparison purposes, we also copy that of other relevant BERT-based models as reported in Le et al. (2019).

17. https://github.com/getalp/Flaubert/tree/master/flue

These models are mBERT (Devlin et al., 2018), CamemBERT (Martin et al., 2019) and FlauBERT (Le et al., 2019).

Among the models having a BASE architecture, BARThez is best in the three sentiment analysis tasks, while being very close to CamemBERT and FlauBERT in the paraphrasing and inference tasks.

Among the LARGE models, mBARThez outperforms mBART in all tasks, showing again the importance of language-adaptive pretraining. On the other hand, CamemBERT and FlauBERT outperform mBARThez in most of the tasks, which could be attributed to the fact that CamemBERT and FlauBERT were trained for approximately 10 times more GPU hours on a monolingual French corpus. Nevertheless, given that huge difference in monolingual training time, it is remarkable that mBARThez is so close, and sometimes outperforms, FlauBERT, with e.g., a comfortable 1.56 margin on PAWSX.

We can conclude that the ability of BARThez and mBARThez to perform well on generative tasks does not appear to come at the expense of a decrease in performance on discriminative tasks, which is in line with the results presented in the BART paper (Lewis et al., 2019).

## 3.7 ARABART EXPERIMENTS

Although AraBART can be adapted to be finetuned on different NLP tasks as we showed in section 3.6, our main focus in this section is abstractive summarization. Our motivation is that other NLU tasks (e.g., text classification, named entity recognition, etc.) were extensively studied by previous efforts capitalizing on BERT-based architectures (Antoun, Baly, and Hajj, 2020; Safaya, Abdullatif, and Yuret, 2020; Abdul-Mageed, Elmadany, and Nagoudi, 2021; Inoue et al., 2021). However these models underperform in generative tasks, and on the other hand, Arabic abstractive summarization remains understudied.

### 3.7.1 *Datasets*

To evaluate AraBART, we use several datasets that consist mostly of news articles annotated with summaries with different level of abstractiveness. The first 7 datasets (*AAW*, *AFP*, *AHR*, *HYT*, *NHR*, *QDS* and *XIN*) are subsets of the Arabic Gigaword (Parker et al., 2011) corpus.[18] Each one is a different news source, composed of document-headline pairs. In all these datasets we use a train set of 50K

18. The datasets come from different Arabic newswire sources: AAW (Asharq Al-Awsat), AFP (Agence France Presse), AHR (Al-Ahram), HYT (Al Hayat), NHR (An Nahar), QDS (Al-Quds Al-Arabi), XIN (Xinhua News Agency).

examples, a validation set of size 5K examples and a test set of size 5K examples, selected randomly. The *MIX* dataset consists of 60K examples uniformly sampled from the union of the 7 different sources.

In addition to the Arabic Gigaword corpus, we use XL-Sum (Hasan et al., 2021). The news articles in XL-sum are annotated with summaries and titles, thus creating two tasks: summary generation, and title generation.

Table 3.11 shows that the different datasets used in our experiments cover a wide range of article/summary lengths and levels of abstractiveness. This variation can be explained by the fact that the target sentences in each dataset follow a different headline writing style. For example, the summaries of the *QDS* dataset which are the shortest and the less abstractive on average, are more like titles extracted from the first paragraph with minimal reformulation. On the other hand, the summaries of XL-Sum, which are the longest and the most abstractive, contain information interspersed in various parts of the input text.

| | **Average # of Tokens** | | **% Novel N-grams in Summary** | | |
|---|---|---|---|---|---|
| | document | summary | unigrams | bigrams | trigrams |
| **AAW** | 453.3 | 15.5 | 44.2 | 78.5 | 91.2 |
| **AHR** | 394.2 | 9.2 | 46.5 | 78.4 | 91.3 |
| **AFP** | 232.8 | 8.3 | 30.7 | 63.6 | 81.9 |
| **HYT** | 474.0 | 11.2 | 42.4 | 78.6 | 92.0 |
| **NHR** | 455.9 | 10.4 | 46.5 | 80.7 | 92.8 |
| **QDS** | 450.6 | 8.0 | 24.9 | 46.9 | 57.5 |
| **XIN** | 187.2 | 8.2 | 26.4 | 48.5 | 60.8 |
| **MIX** | 364.5 | 9.4 | 40.0 | 72.2 | 86.3 |
| **XL-S** | 428.7 | 25.6 | 53.5 | 85.8 | 95.2 |
| **XL-T** | 428.7 | 9.4 | 44.3 | 81.2 | 94.1 |

Table 3.11 – Statistics of Gigaword subsets, as well as XL-Sum summaries (XL-S) and titles (XL-T). The first two columns show the average document and summary lengths. The last three columns show the percentage of n-grams in the summary that do not occur in the input article, used here as a measure of abstractiveness (Narayan, Cohen, and Lapata, 2018a).

### 3.7.2 *Baselines*

We compare our model to four types of state-of-the-art sequence-to-sequence baselines. The first, called C2C, is a monolingual seq2seq model based on BERT2BERT (Rothe, Narayan, and Severyn, 2020a). The encoder and decoder are initialized us-

ing CAMELBERT (Inoue et al., 2021) weights while the cross-attention weights are randomly initialized. [19] C2C has 275M parameters in total.

The second baseline is mBART25 (Liu et al., 2020a) which is a multilingual BART pretrained on 25 different languages including Arabic. Although mBART25 was initially pretrained for neural machine translation, it was shown that it can be used in monolingual generative tasks such as abstractive summarization (Kamal Eddine, Tixier, and Vazirgiannis, 2021). mBART25 has 610M parameters in total.

Another multilingual model that we include as a baseline in our experiments is $\text{mT5}_{base}$ (Hasan et al., 2021). mT5 is a multilingual variant of T5 (Raffel et al., 2020) pretrained on the mC4 dataset - a large corpus comprising 27T of natural text in 101 different languages including Arabic. $\text{mT5}_{base}$ has 390M parameters in total. Another recently released T5-based model is AraT5, pretraind on 70GB of natural text written in modern standard Arabic. For a fair comparison, we use the *base* version of mT5 and AraT5. Table 3.5 summarizes the specifications of the different models used in our experiments.

### 3.7.3 *Training and Evaluation*

We finetuned each model for three epochs, using the Adam optimizer and $5 \times 10^{-5}$ maximum learning rate with linear decay scheduling. In the generation phase we use beam-search with beam size of 3.

For evaluation, we first normalize the output summaries as is common practice in Arabic: we removed Tatweel and diacritization, we normalized Alif/Ya, and we separated punctuation marks. We report ROUGE-1, ROUGE-2 and ROUGE-L f1-scores (Lin, 2004a). However, these metrics are solely based on surface-form matching and have limited sense of semantic similarity (Kamal Eddine et al., 2021). Thus we opted for using BERTScore (Zhang et al., 2020), a metric based on the similarity of the contextual embeddings of the reference and candidate summaries, produced by a BERT-like model. [20]

### 3.7.4 *Results*

We observe in Table 3.12 that AraBART outperforms C2C on all datasets with a clear margin. This is probably a direct consequence of pretraining the seq2seq architecture end-to-end.

19. We experimented with ARABERT (Antoun, Baly, and Hajj, 2020) which was slower to converge and didn't achieve better performance.

20. We use the official implementation (`https://github.com/Tiiiger/bert_score`) with the following options: `-m UBC-NLP/ARBERT -l 9` (Chiang, Huang, and Lee, 2020)

| Source | Model | R1 | R2 | RL | BS |
|---|---|---|---|---|---|
| ***AAW*** | AraBART | **30.7** | **15.3** | **27.4** | **62.5** |
| | mBART25 | 29.5 | 14.4 | 26.0 | 61.5 |
| | $mT5_{base}$ | 26.3 | 11.9 | 23.3 | 61.5 |
| | $AraT5_{base}$ | 24.1 | 9.8 | 21.3 | 56.7 |
| | C2C | 24.6 | 9.9 | 21.7 | 58.3 |
| ***AFP*** | AraBART | **55.0** | **37.9** | **53.4** | **77.5** |
| | mBART25 | 54.8 | 37.3 | 52.8 | 77.2 |
| | $mT5_{base}$ | 52.8 | 35.8 | 51.0 | 61.5 |
| | $AraT5_{base}$ | 47.8 | 29.6 | 46.3 | 73.6 |
| | C2C | 50.0 | 32.2 | 48.4 | 74.8 |
| ***AHR*** | AraBART | **39.1** | 25.4 | **37.7** | **68.2** |
| | mBART25 | **39.1** | **26.1** | 37.5 | 68.1 |
| | $mT5_{base}$ | 33.3 | 20.1 | 31.7 | 64.7 |
| | $AraT5_{base}$ | 25.6 | 12.9 | 24.4 | 59.4 |
| | C2C | 33.0 | 19.7 | 31.8 | 63.5 |
| ***HYT*** | AraBART | **33.1** | **17.5** | **30.7** | **63.8** |
| | mBART25 | 32.0 | 16.2 | 29.3 | 63.1 |
| | $mT5_{base}$ | 29.9 | 14.5 | 27.5 | 62.0 |
| | $AraT5_{base}$ | 26.3 | 10.7 | 24.2 | 58.0 |
| | C2C | 27.4 | 11.5 | 25.2 | 59.6 |
| ***NHR*** | AraBART | **32.0** | **17.2** | **30.3** | **61.2** |
| | mBART25 | 31.0 | 16.2 | 29.2 | 60.3 |
| | $mT5_{base}$ | 27.3 | 13.3 | 25.6 | 58.5 |
| | $AraT5_{base}$ | 19.5 | 7.5 | 18.3 | 51.1 |
| | C2C | 24.1 | 10.0 | 22.9 | 53.0 |

| Source | Model | R1 | R2 | RL | BS |
|---|---|---|---|---|---|
| ***QDS*** | AraBART | 62.1 | 53.9 | 61.4 | 80.3 |
| | mBART25 | **62.4** | **54.1** | **61.7** | **80.4** |
| | $mT5_{base}$ | 59.3 | 50.5 | 58.5 | 78.7 |
| | $AraT5_{base}$ | 56.3 | 47.1 | 55.6 | 76.4 |
| | C2C | 57.9 | 48.9 | 57.4 | 77.3 |
| ***XIN*** | AraBART | **66.0** | **53.9** | **65.1** | **84.4** |
| | mBART25 | 65.1 | 53.4 | 64.2 | 84.0 |
| | $mT5_{base}$ | 64.1 | 52.2 | 63.2 | 83.4 |
| | $AraT5_{base}$ | 61.5 | 48.5 | 60.6 | 82.3 |
| | C2C | 62.4 | 50.1 | 61.6 | 82.5 |
| ***MIX*** | AraBART | **39.2** | 25.5 | **37.6** | **67.6** |
| | mBART25 | 39.0 | **25.6** | 37.1 | 67.2 |
| | $mT5_{base}$ | 33.1 | 20.0 | 31.5 | 64.0 |
| | $AraT5_{base}$ | 32.2 | 18.8 | 30.8 | 62.2 |
| | C2C | 32.8 | 19.1 | 31.4 | 62.5 |
| ***XL-S*** | AraBART | **34.5** | **14.6** | **30.5** | **67.0** |
| | mBART25 | 32.1 | 12.5 | 27.6 | 65.3 |
| | $mT5_{base}$ | 32.8 | 12.7 | 28.7 | 65.8 |
| | $AraT5_{base}$ | 25.2 | 7.6 | 21.6 | 58.1 |
| | C2C | 26.9 | 8.7 | 23.1 | 61.6 |
| ***XL-T*** | AraBART | **32.0** | **13.7** | **29.4** | **65.8** |
| | mBART25 | 29.8 | 11.7 | 26.9 | 64.3 |
| | $mT5_{base}$ | 25.7 | 9.3 | 23.5 | 61.6 |
| | $AraT5_{base}$ | 24.0 | 7.1 | 21.8 | 57.3 |
| | C2C | 25.2 | 7.9 | 22.9 | 61.1 |

| Source | Model | R1 | R2 | RL | BS |
|---|---|---|---|---|---|
| ***Macro Averages*** | AraBART | **42.4** | **28.8** | **40.3** | **69.8** |
| | mBART25 | 41.5 | 28.1 | 39.2 | 69.1 |
| | $mT5_{base}$ | 38.5 | 24.0 | 36.5 | 66.2 |
| | $AraT5_{base}$ | 34.2 | 20.0 | 32.5 | 63.5 |
| | C2C | 36.4 | 23.1 | 34.6 | 65.4 |

Table 3.12 – The performance of AraBART, mBART25, $mT5_{base}$, $AraT5_{base}$, and C2C (CAMeLBERT2CAMeLBERT) on all datasets in terms of ROUGE-1 (R1), ROUGE-2 (R2), ROUGE-L (RL) and BERTScore (BS). Macro averages are computed over all datasets.

AraBART also outperforms mBART25 on XL-Sum which is the most abstractive dataset. On Gigawords, AraBART is best everywhere except on AHR with mitigated results. On QDS, the set with the least abstractive summaries (see Table 3.11), however, it falls clearly behind mBART25 on all metrics. In fact, we notice that the gap between AraBART and the baselines is greater on the XL-Sum dataset than Gigaword. For instance, our model's ROUGE-L score is 2.9 absolute points higher that mBART25 on XL-S while the maximum margin obtained on a Gigaword subset is 1.4 points on AAW and HYT. We observe a tendency for AraBART to outperform mBART on more abstractive datasets. In fact, the margin between their BERTScores is positively correlated with abstractiveness as measured by the percentage of novel trigrams. [21]

| System | Reference | AraBART | C2C | mBART | mT5 | BWS Score |
|---|---|---|---|---|---|---|
| **Reference** | - | 44.7 | 79.0 | 53.0 | 56.5 | 16.65 |
| **AraBART** | 55.3 | - | 82.85 | 54.75 | 58.5 | **25.6** |
| **C2C** | 21.0 | 17.15 | - | 14.5 | 15.5 | -65.9 |
| **mBART** | 47.0 | 45.25 | 85.5 | - | 50.5 | 14.2 |
| **mT5**$_{base}$ | 43.5 | 41.5 | 84.5 | 49.5 | - | 9.55 |

Table 3.13 – Human evaluation using Best-Worst Scaling (BWS). The numbers in the first five columns represent the percentage of the times the *row* model was chosen as better than the *column* model. The BWS score is the percentage of time the model's summary was chosen as best minus the percentage of time it was chosen as worst.

## 3.8 HUMAN EVALUATION

To validate the automatic evaluation results, we conducted a detailed manual evaluation that covers two aspects: **quality** and **faithfulness**. We considered 100 documents randomly sampled from the test set along with their respective candidate summaries. The systems included in the manual evaluation are: AraBART, mBART25, mT$_{base}$ and CAMeLBERT2CAMeLBERT. [22] In addition to the generated summaries, we include the reference summaries following Narayan, Cohen, and Lapata (2018a) and Kamal Eddine, Tixier, and Vazirgiannis (2021). The annotations were carried out by 14 Arabic native speaker volunteers. To guarantee a better quality assessments, each example was annotated by two volunteers separately.

21. With a Pearson R score of 0.6625 and $p$-value<0.05.

22. We separately evaluate the AraT5 model (Al-Maleh and Desouki, 2020), which was not yet published at the time of this human evaluation, in Section 3.8.3.

### 3.8.1 *Quality Evaluation*

To assess the overall quality of system summaries we use the *Best-Worst Scaling* (BWS) method (Narayan, Cohen, and Lapata, 2018a). For each document, the annotators were provided with the list of all possible combinations of summary pairs. They were asked to choose the best summary of each of the pairs. To help them in their decisions the annotators were asked to focus on three aspects: *factuality* (does the summary contain factual information?), *relevance* (does the summary capture the important information in the document?) and *fluency* (is the summary written in well-formed Arabic?).

Table 3.13 shows a pairwise comparison between the models with regard to their overall quality. The scores represent the percentage of the times the *row* model was chosen as better than the *column* model. The last column in the table represent the BWS score, which is, for each model the percentage of time the model's summary was chosen as best minus the percentage of time it was chosen as worst (Narayan, Cohen, and Lapata, 2018a).

The manual quality assessment showed the same ranking as the automatic evaluation presented in Table 3.12. However, in the current assessment, the differences between the models' performances vary. For example, AraBART, which is the top performing model, has a wider margin compared to mBART25. On the other hand, mBART25 lost its significant margin compared to the mT5 model. These findings highlight the importance of carrying out manual evaluation in the context of abstractive summarization generation. Finally, AraBART summaries were even judged as being of better quality than some references by the annotators. While this finding could seem problematic, it is in line with previous efforts (Narayan, Cohen, and Lapata, 2018a; Kamal Eddine, Tixier, and Vazirgiannis, 2021). The lower scores of the reference summaries are related to the nature of the task itself. The news headline generation task considers headlines written by journalists as summaries. However these headlines, while being relevant and fluent, may contain some information that is not presented by the input document such as names and dates. These bits of information are considered by the human annotators as inaccurate or non-factual. This assumption is confirmed in the next section.

### 3.8.2 *Faithfulness Evaluation*

Recent efforts have shown that automatic systems are highly prone to generate content that is unfaithful to the source document (Maynez et al., 2020; Chen et al., 2021). Thus, we opted for a manual evaluation that focuses on the summaries faithfulness. In this evaluation task, we asked the annotators to detect *unfaithful spans*. A

| System | Unfaithful Spans # | Faithful Words % |
|---|---|---|
| **Reference** | 2.31 | 77.91 |
| **AraBART (ours)** | **1.36** | **84.47** |
| **C2C** | 3.18 | 61.80 |
| **mBART** | 1.68 | 81.31 |
| **mT**$_{base}$ | 1.49 | 81.62 |

Table 3.14 – Faithfulness results in terms of the average number of unfaithful spans of text in summaries (less is more faithful), and the percentage of faithful words in summaries (higher is more faithful).

span is considered as unfaithful if it contains information that is not covered by the input document even if the information is factual (Maynez et al., 2020).

Automatic metrics based on surface token (e.g., Rouge) or distributional semantic (e.g., BERTScore) overlap between the reference and the generated summaries are not sufficient for abstractive summarization evaluation. This is mainly because they are not able to capture the faithfulness of the summary with respect to the input document. This is why, manually assessing the faithfulness of the summary could be very useful for evaluating the summarization systems. Table 3.14 shows the degree of faithfulness of each model to the input document.

Here again, AraBART outperforms all the other systems, obtaining a lower number of unfaithful spans and a higher percentage of faithful summary words. On the other hand, the reference summaries are outperformed by AraBART and two other baselines which confirms our assumption in Section 3.8.1 about the underperformance of the reference summaries compared to AraBART. The difference in the system rankings and the improvement margins between the automatic, the quality and the faithfulness evaluations, highlights the importance of conducting a detailed evaluation considering various aspects and dimensions.

### 3.8.3 *AraBART vs AraT5*

At the time we carried out the manual evaluation, the AraT5 model (Al-Maleh and Desouki, 2020) was not yet published. For this reason we performed a separate quality assessment evaluation comparing AraT5 to AraBART only. We used the same 100 documents as previously, and the annotators had to choose the better summary among those of AraT5 and AraBART following the same guidelines of the overall quality assessment. Three annotators participated in this evaluation task,

and each document was annotated by only one participant. The final score shows that 91.5% of the time AraBART summaries were chosen as best, which again shows the superiority of AraBART in the abstactive summarization task.

## 3.9 CONCLUSION

We released BARThez and mBARThez, the first large-scale pretrained seq2seq models for the French language, as well as a novel summarization dataset for French, inspired by the XSum dataset. By evaluating our models on the summarization dataset, we showed that: (1) BARThez is on par with mBART while having four times fewer parameters and that (2) mBARThez provides a significant boost over mBART by simply adding a relatively affordable language-adaptive phase to the pretraining. In addition, we evaluated BARThez and mBARThez on 5 sentiment analysis, paraphrasing, and natural language inference tasks against cutting-edge BERT-based French language models (FlauBERT and CamemBERT), and obtained very competitive results.

On the other hand, we released AraBART, the first sequence-to-sequence pretrained Arabic model. We evaluated our model on a set of abstractive summarization tasks with different levels of abstractiveness. We compared AraBART to a number of state-of-the-art models, and we showed that it outperforms them almost everywhere despite being smaller in terms of parameters.

## ACKNOWLEDGMENTS

We are grateful to the three anonymous reviewers for their constructive feedback. We thank the National Center for Scientific Research (CNRS) for giving us access to the Jean Zay supercomputer, under allocation 2020-AD011011499.

# 4

# FRUGALSCORE

Fast and reliable evaluation metrics are key to R&D progress. While traditional natural language generation (NLG) metrics are fast, they are not very reliable. Conversely, new metrics based on large pretrained language models are much more reliable, but require significant computational resources. In this paper, we propose FrugalScore, an approach to learn a fixed, low cost version of any expensive NLG metric, while retaining most of its original performance. Experiments with BERTScore (Zhang et al., 2020) and MoverScore (Zhao et al., 2019b) on summarization and translation show that FrugalScore is on par with the original metrics (and sometimes better), while having several orders of magnitude less parameters and running several times faster. On average over all learned metrics, tasks, and variants, FrugalScore retains 96.8% of the performance, runs 24 times faster, and has 35 times less parameters than the original metrics. We make our trained metrics publicly available [1] and easily accessible online, to benefit the entire NLP community and in particular researchers and practitioners with limited resources.

## 4.1 INTRODUCTION

Automatic evaluation metrics are the only way to monitor the training of, evaluate, and compare across models in a systematic, large-scale way, and are thus a critical component of the research and development ecosystem in machine learning. To get adopted in practice, evaluation metrics need to be both reliable and affordable, i.e., fast and easy to compute.

While some metrics meet these criteria, such as precision and recall in information retrieval, root mean square error in regression, etc., finding suitable metrics is still an open problem in the field of Natural Language Generation (NLG) (Novikova et al., 2017).

Indeed, historical $n$-gram matching metrics such as ROUGE (Lin, 2004b) for summarization, BLEU (Papineni et al., 2002) and METEOR (Banerjee and Lavie, 2005) for translation, while affordable, are not very reliable, as they are based on surface-form matching only, i.e., lexical similarity, and have thus no sense of semantic similarity. For instance, it makes little sense to use ROUGE for the evaluation of abstractive summarization systems (which are becoming the norm), or whenever the generated text paraphrases the original text.

1. https://github.com/moussaKam/FrugalScore

Following the advent of transfer learning in NLP, new NLG metrics based on large pretrained language models have recently been proposed, such as BERTScore (Zhang et al., 2019) and MoverScore (Zhao et al., 2019b). By relying on contextual embeddings, these metrics capture semantics and are therefore much more reliable. However, due to the sheer size of the underlying models, these metrics pose environmental issues (Strubell, Ganesh, and McCallum, 2019), take time to compute, and require access to significant computational resources, so they are not accessible by everyone in the NLP community.

For example, we were not able to run some of the best variants of BERTScore [2], based on DeBERTa-Large and DeBERTa-XLarge (He et al., 2020) on a 12GB GPU. Even when enough GPU memory is available, relying on such large models is still associated with extended runtimes, which can impede the progress of experiments when used once or more per epoch for validation and monitoring purposes.

To address this problem, we propose in this paper FrugalScore, an approach to learn a lightweight version of BERTScore, MoverScore, and more generally any metric based on a large pretrained language model.

Our contributions can be summarized as follows:
1) Our compact models have several orders of magnitude less parameters than the original metrics and run several times faster, while retaining most of their original performance. We even outperform the original metrics in some cases [3].
2) Our metrics are not only faster because of the much smaller amount of parameters, but also because they do not rely on any similarity function.
3) Regardless of how expensive the original metric is, querying our trained metrics always has the same low, fixed cost. This decoupling is a major advantage as the size of the pretrained language models has recently been growing tremendously (e.g., Brown et al. (2020)).

## 4.2 BACKGROUND

Related work falls into two categories: unsupervised and supervised metrics.

### 4.2.1 *Unsupervised metrics*

To address the limitations of ROUGE and BLEU, variants based on static word embeddings (Mikolov et al., 2013a) were developed, e.g., ROUGE-WE (Ng and Abrecht, 2015a), BLEU2VEC (Tättar and Fishel, 2017), and MEANT 2.0 (Lo, 2017). While using word vectors is a progress over strict *n*-gram matching, static embed-

2. From BERTScore's authors: https://tinyurl.com/8cwyter2
3. Hence the name FrugalScore, as frugal engineering is defined as "achieving more with fewer resources".

dings are still very limited as they do not capture polysemy, i.e., the fact that words have different meanings in different contexts.

More recently, the focus has shifted to harnessing the power of the contextualized embeddings produced by large pretrained language models. For instance, the Sentence Mover's Similarity (Clark, Celikyilmaz, and Smith, 2019b) represents sentences as the average of their ELMo word embeddings (Peters et al., 2018a) and measures the minimum cost of transforming one summary into the other, using a modified version of the Word Mover's Distance (Kusner et al., 2015). BERTR (Mathur, Baldwin, and Cohn, 2019b) computes approximate recall based on the pairwise cosine similarity between the BERT embeddings (Devlin et al., 2018) of the words in automatic and reference translations. Mark-Evaluate (Mordido and Meinel, 2020) is a family of metrics that consider contextualized word or sentence embeddings derived from BERT as population samples, to evaluate language generation with population estimation methods used in ecology.

Finally, the recently introduced BERTScore (Zhang et al., 2019) and MoverScore (Zhao et al., 2019b) are general-purpose NLG evaluation metrics that are becoming widely used. The main difference between BERTScore and MoverScore lies in the function used to compute the similarity between the representations of the two sequences $\mathbf{x} = \langle \mathbf{x_1}, ..., \mathbf{x_k} \rangle$ and $\mathbf{y} = \langle \mathbf{y_1}, ..., \mathbf{y_l} \rangle$. We experimented with these two metrics, so we provide more details about them in what follows.

**BERTScore** first computes the pairwise cosine similarity between the representations of the tokens in each sequence, and uses greedy matching to match each token to the most similar one in the other sequence. Given two pre-normalized vector sequences $\mathbf{x}$ and $\mathbf{y}$, BERTScore computes:

$$R_{BERT} = \frac{1}{|\mathbf{x}|} \sum_{\mathbf{x_i} \in x} \max_{\mathbf{y}_j \in \mathbf{y}} \mathbf{x_i^T y_j} \tag{4.1}$$

and:

$$P_{BERT} = \frac{1}{|\mathbf{y}|} \sum_{\mathbf{y_i} \in y} \max_{\mathbf{x}_j \in \mathbf{x}} \mathbf{y_i^T x_j} \tag{4.2}$$

The F1-score is classically obtained as:

$$F_{BERT} = 2\frac{P_{BERT} R_{BERT}}{P_{BERT} + R_{BERT}} \tag{4.3}$$

**MoverScore** uses an $n$-gram generalization of the Word Mover's Distance (WMD) (Kusner et al., 2015) as their (dis)similarity function. More specifically, they solve for the optimal transportation flow matrix $F \in \mathbb{R}^{|\mathbf{x}| \times |\mathbf{y}|}$ between the two weighted sequences of $n$-grams:

$$WMD(\mathbf{x}, \mathbf{y}) = min_F \langle C, F \rangle \tag{4.4}$$

$$s.t. \quad F\mathbf{1} = f_{\mathbf{x}}, \quad F^T\mathbf{1} = f_{\mathbf{y}}$$

Where $C$ is the transportation cost matrix ($C_{ij}$ is the Euclidean distance between $x_i$ and $y_j$) and $f_x \in \mathbb{R}_+^{|\mathbf{x}|}$ and $f_y \in \mathbb{R}_+^{|\mathbf{y}|}$ are the $n$-gram weight vectors.

Note that by directly learning BERTScore's and MoverScore's full internal mapping (from sequence pairs to final scalar scores), FrugalScore internalizes their similarity functions. This does not only provide a speedup at inference time, but also improves performance, as shown in section 4.5.

### 4.2.2 *Supervised metrics*

Related to our work are also supervised metrics, which are directly trained on human evaluations. ROSE (Conroy and Dang, 2008) is a linear combination model of different variants of ROUGE using canonical correlation. BEER (Stanojević and Sima'an, 2014) is a learning-to-rank approach using word and character n-gram matching, and token ordering, as features to maximize correlation with human rankings of machine translation systems. $S^3$ (Peyrard, Botschen, and Gurevych, 2017) trains a regression model that takes the evaluation scores of several existing metrics and many hand-crafted features as input, and learns the best combination of them to approximate human summary judgments. DPMFcomb (Yu et al., 2015) and Blend (Ma et al., 2017) are combined metrics incorporating a vast amount of lexical, syntactic and semantic based translation evaluation metrics using ranking and regression SVMs respectively. RUSE (Shimanaka, Kajiwara, and Komachi, 2018) evaluates machine translation with a neural regressor based on universal sentence embeddings (e.g., InferSent (Conneau et al., 2017)). NUBIA (Kane et al., 2020) consists of three modules: a feature extractor based on RoBERTa (Liu et al., 2019) and GPT-2 (Radford et al., 2019) fine-tuned on language evaluation tasks, an aggregator trained to predict the quality of the hypothesis given the reference using the extracted features, and a calibrator mapping all predictions between 0 and 1.

**Differences**. Like the aforementioned efforts, FrugalScore is a learned metric. However, it does not rely on any intermediate or handcrafted features, and, most importantly, it does not require training on human annotations. Supervision in FrugalScore is conducted on a synthetic dataset, as a trick to expose and learn the internal mapping of the unsupervised metrics to be learned. Last but not least, unlike all aforementioned methods, compression is central to FrugalScore, which is based on miniature versions of the models used by the original metrics.

### 4.2.3 *Knowledge distillation*

Knowledge distillation (KD) (Hinton, Vinyals, and Dean, 2015) is the process of transferring knowledge from a large teacher model to a smaller student model to accomplish model compression (Buciluǎ, Caruana, and Niculescu-Mizil, 2006). It was originally proposed in the domain of computer vision and speech recognition, then successfully adapted to NLP (Sanh et al., 2019). Distillation can be accomplished in three ways: (1) offline, where a teacher is first pre-trained, then a student is trained under the guidance of the teacher (Hinton, Vinyals, and Dean, 2015); (2) online, where the student and the teacher are trained simultaneously (Zhang et al., 2018); and (3) self, where the same model plays the role of student and teacher, e.g., transferring the knowledge of a later exit layer into an earlier one of the same multi-exit network (Phuong and Lampert, 2019). Previous studies on KD mainly focused on classification problems (Gou et al., 2021). A few attempts have been made on regression problems (Chen et al., 2017; Saputra et al., 2019; Takamoto, Morishita, and Imaoka, 2020), in which special losses were proposed to train the student with respect to both the teacher's regression outputs and ground truth scores. Different from conventional distillation, our work is more similar to *data-free* KD (Kang and Kang, 2021), where the student is trained in the absence of the dataset used to train the teacher. To transfer knowledge, we first create a synthetic dataset by annotating sequence pairs with a large model (teacher), and then train a miniature model (student) on that dataset, in an offline and regression setting.

### 4.2.4 *Differences with BLEURT*

A work closely related to ours is BLEURT (Sellam, Das, and Parikh, 2020a). However, there are a number of significant differences with our approach. First, BLEURT continues the pretraining of an already pretrained BERT-based model on a synthetic dataset in a self-supervised way, whereas FrugalScore is directly trained to learn the scores of the metric of interest, in a supervised fashion.

Also, BLEURT's synthetic dataset is made by perturbing Wikipedia sentences with mask-filling, backtranslation, and word dropping, whereas we use other data sources than Wikipedia such as summarization and translation datasets, and only NLG models to induce perturbations.

When creating its synthetic dataset, BLEURT automatically annotates the (original, perturbed) sequence pairs with numerical and categorical "signals": BLEU, ROUGE, BERTscore, backtranslation likelihood, textual entailment (probability of three labels: entail, contradict, and neutral, given by BERT fine-tuned on MNLI), and backtranslation flag. On the other hand, FrugalScore simply and directly annotates the sequence pairs with the metric to be learned.

After pretraining, BLEURT is fine-tuned on human judgments, in a way similar to the supervised metrics described in subsection 4.2.2. BLEURT does not learn to generate a scalar until that final fine-tuning phase, so it cannot be used as a metric before that. Conversely, FrugalScore is trained from the start to be a metric, and the fine-tuning phase is optional.

Also, BLEURT was designed for the evaluation of translation. The authors only test whether it can be applied to a different task by experimenting on the WebNLG (data-to-text) dataset (Gardent et al., 2017). Conversely, we focus on learning general text similarity metrics (e.g., BERTscore and MoverScore), so FrugalScore is task-agnostic by design.

Finally, and above all, the objective of FrugalScore is model compression, whereas that of BLEURT is metric learning.

## 4.3 OUR APPROACH

Developing FrugalScore requires three phases, one of which is optional.

**Phase 1**. We create a synthetic dataset (see subsection 4.3.1) by sampling pairs of more or less related sequences and annotating them with the expensive metrics to be learned. This is a one-time operation that does not need to be repeated regardless of the model used in Phase 2.

**Phase 2**. We continue the pretraining (see subsection 4.3.2) of a miniature pretrained language model on the synthetic dataset built by Phase 1. Here, the miniature model learns the internal mapping of the expensive metric, including any similarity function applied to the representations. Note that a different miniature is trained for each metric to be learned (we leave learning metric combinations as future work).

The miniature can then be used in inference mode to generate scores for any never-seen pair of sequences.

**Phase 3** (optional). We fine-tune the miniature on human annotations, which, as shown in section 4.6, can boost performance.

### 4.3.1 *Synthetic dataset*

The objective here was to generate pairs of sequences mimicking the (reference, candidate) pairs found in NLG datasets, which are usually semantically related and in many cases paraphrasing one another. We sampled our sequences from a variety of data sources, listed next.

**Summarization**. For each document in the well-known CNN/DailyMail dataset (Nallapati et al., 2016), our goal was to generate several summaries differing in terms of structure and quality. To this purpose, we used different pretrained seq2seq summarization models: BART-base and BART-large (Lewis et al., 2019), mBART (Liu et al., 2020b), and BARThez (Kamal Eddine, Tixier, and Vazirgiannis, 2021). BART is a seq2seq autoencoder with a Transformer architecture.

The four models were fine-tuned for one epoch on 50k examples randomly sampled from the training set of CNN/DM, and were used to generate summaries for the whole training set of 287,112 documents, using greedy decoding.

Note that we kept the 50K documents used for fine-tuning in the final generation pool, in order to create quality differences among summaries. Indeed, models are expected to better summarize the documents used for training than never-seen documents.

We also used the human reference summaries, so that in the end, each document was associated with 5 summaries, resulting in 10 pairs of summaries per document.

**Backtranslation**. We also generated paraphrases with backtranslation, by sampling sentences from the OpenSubtitle English monolingual corpus (Lison and Tiedemann, 2016), and translating them to French, Arabic and German with OPUS-MT (Tiedemann and Thottingal, 2020), before translating them back to English. We used OPUS-MT because of its ready-to-use checkpoints available for many language pairs. We ended up with 4 variations for each sentence (including the original one), resulting in 6 paraphrase pairs per sentence.

**Denoising**. To avoid bias towards summarization and translation, we also generated pairs of related sequences such that the first element in the pair was a Wikipedia segment and the second element was a BART-denoised version of it (Lewis et al., 2019).

More precisely, we sampled 2M segments from Wikipedia such that the number of unigrams in these segments was uniformly distributed between 1 and 200. Our assumption was that enforcing variations in sequence length would help the learned metric to generalize.

We then applied BART's *text infilling* and *sentence permutation* perturbation strategies to each segment. That is, multiple text spans were sampled and replaced with a `[MASK]` special token. The lengths of the spans were sampled from a Poisson distribution ($\lambda = 3$). 50% of the tokens within the input segment were masked and 20% of the masked text was replaced with random tokens (creating pathological examples to increase the robustness of the learned metric). The sentences in the input segment were then shuffled.

We finally used a BART-Base checkpoint [4] from the Fairseq library (Ott et al., 2019b) to try to reconstruct the perturbed versions of the original sequences, hence creating variants of them.

**Annotating pairs**. We sampled 4.5M sequence pairs uniformly from each aforelisted source. These pairs were then annotated with the metrics to be learned. Note that this is a one-time operation that does not need to be repeated regardless of which models are trained downstream.

In this work, we experimented with two recent expensive NLG metrics that rely on large pretrained language models, BERTScore (Zhang et al., 2019) and MoverScore (Zhao et al., 2019b), presented in section 4.2. However, it is important to note that our method can be used with any other NLG metric.

Note that for BERTScore, we used the F-1 score $F_{BERT}$, as recommended by the authors (Zhang et al., 2019). For MoverScore, still following the authors (Zhao et al., 2019b), we used the variant operating on unigrams and the IDF to compute the vectors of weights.

### 4.3.2 *Metric learning*

We continue the pretraining of three BERT miniatures [5] on our synthetic dataset: BERT-Tiny ($L = 2$, $H = 128$), BERT-Small ($L = 4$, $H = 512$) and BERT-Medium ($L = 8$, $H = 512$), where $L$ is the number of layers and $H$ is the dimension of the embedding space. These models have respectively 25 times, 3.78 times, and 2.64 times less parameters than BERT-base. The concept of BERT miniatures was introduced by Turc et al. (2019) to test whether pretraining small models from scratch was competitive to distilling very large models. The miniature models have already been pretrained on masked language model and next sentence prediction objectives.

We continue pretraining using the standard method introduced by Devlin et al. (2018). We concatenate the two sequences $x = \langle x_1, ..., x_k \rangle$ and $y = \langle y_1, ..., y_l \rangle$ in a given pair, separating them with a special `[SEP]` token. A special `[CLS]` token is also added at the beginning of the resulting sequence. The sequence of contextualised embeddings $\langle \mathbf{z}_{[\mathbf{CLS}]}, \mathbf{x_1}, ...\mathbf{x_k}, \mathbf{z}_{[\mathbf{SEP}]}, \mathbf{y_1}, ..., \mathbf{y_l} \rangle$ is then obtained. We finally add a fully connected layer on top, that linearly projects the $\mathbf{z}_{[\mathbf{CLS}]}$ vector to a scalar $s$.

The model is trained to minimize the mean square error (MSE) loss between the learned metric $s_i$ and the metric to be learned $\hat{s_i}$ (i.e., the annotation of the pair):

$$l = \frac{1}{N} \sum_{i=1}^{N} ||s_i - \hat{s_i}||^2 \tag{4.5}$$

4. https://dl.fbaipublicfiles.com/fairseq/models/bart.base.tar.gz
5. https://huggingface.co/google

| | Metric | Model | Scores (TAC) | Runtime (TAC) | Scores (WMT) | Runtime (WMT) | Params |
|---|---|---|---|---|---|---|---|
| a | BERTScore | BERT-Tiny | 55.4/47.5 | 1m 27s | 37.6 | 1m 22s | 4.4M |
| b | BERTScore | BERT-Small | 61.6/51.5 | 2m 20s | 39.1 | 1m 42s | 29.1M |
| c | BERTScore | BERT-Medium | 62.7/52.4 | 2m 28s | 39.8 | 2m 04s | 41.7M |
| d | BERTScore | BERT-Base | 64.7/54.7 | 3m 28s | 41.9 | 2m 09s | 110M |
| e | BERTScore | RoBERTa-Large | 64.2/55.4 | 5m 17s | 43.2 | 3m 03s | 355M |
| f | BERTScore | DeBERTa-XLarge | 64.5/**56.0** | 6m 20s | **44.5** | 3m 49s | 900M |
| g | MoverScore | BERT-Base | **66.5**/55.4 | 301m 29s | 44.0 | 64m 32s | 110M |
| i | FrugalScore$_d$ | BERT-Tiny | 64.9/53.5 | 1m 28s | 38.4 | 1m 18s | 4.4M |
| ii | FrugalScore$_d$ | BERT-Small | 64.7/53.7 | 2m 29s | 41.3 | 1m 35s | 29.1M |
| iii | FrugalScore$_d$ | BERT-Medium | 64.8/54.2 | 3m 41s | 41.9 | 1m 55s | 41.7M |
| iv | FrugalScore$_e$ | BERT-Tiny | 60.0/50.1 | 1m 28s | 37.5 | 1m 18s | 4.4M |
| v | FrugalScore$_e$ | BERT-Small | 64.1/53.8 | 2m 29s | 40.5 | 1m 35s | 29.1M |
| vi | FrugalScore$_e$ | BERT-Medium | 63.9/52.1 | 3m 41s | 41.7 | 1m 55s | 41.7M |
| vii | FrugalScore$_f$ | BERT-Tiny | 61.7/51.0 | 1m 28s | 38.0 | 1m 18s | 4.4M |
| viii | FrugalScore$_f$ | BERT-Small | 66.0/54.9 | 2m 29s | 41.5 | 1m 35s | 29.1M |
| ix | FrugalScore$_f$ | BERT-Medium | 65.5/54.9 | 3m 41s | 43.0 | 1m 55s | 41.7M |
| x | FrugalScore$_g$ | BERT-Tiny | **67.3**/**55.1** | 1m 28s | 39.8 | 1m 18s | 4.4M |
| xi | FrugalScore$_g$ | BERT-Small | 65.9/54.7 | 2m 29s | 42.8 | 1m 35s | 29.1M |
| xii | FrugalScore$_g$ | BERT-Medium | 66.2/**55.1** | 3m 41s | **43.6** | 1m 55s | 41.7M |

Table 4.1 – Scores are summary-level (TAC) and segment-level (WMT) Pearson correlations averaged over 2008 to 2011 for TAC (pyramid score/responsiveness) and over all source languages for WMT-2019. Runtimes include preprocessing. Subscripts refer to row labels and indicate which metric-model combination was used to annotate pairs (e.g., for FrugalScore$_d$, it is row $d$, i.e., BERTScore-BERT-Base).

When pretraining is over, the models can be further fine-tuned on smaller human-annotated datasets as shown in section 4.6, or directly used to generate scores for unseen examples as shown in section 4.4.

**Setup**. We use a batch size of 32 and the Adam optimizer (Kingma and Ba, 2015) with a learning rate of $3 \times 10^{-5}$, linear decay, and a warm-up for 6% of the total training steps, and we train the model for three epochs. We conducted the pretraining on a single TITAN RTX GPU (24GB). It took 10, 24 and 33 hours, respectively for the tiny, small, and medium miniatures. We rely on the TRANSFORMERS library (Wolf et al., 2019) for all pretraining and fine-tuning experiments.

## 4.4 EXPERIMENTS

In this section, FrugalScore is used in inference mode to generate scores directly after pretraining, i.e., no fine-tuning is performed (see section 4.6 for fine-tuning results).

We evaluate on two text generation tasks: summarization and translation. We use evaluation datasets containing (reference, candidate) sequence pairs annotated with human scores assessing the quality of the candidates given the references. We measure the effectiveness of FrugalScore by measuring the Pearson correlation of its scores with the human judgments and comparing it to that of the original metrics. We also take the number of parameters and the runtime into account.

**Text summarization**. We use 4 different multi-document summarization datasets from the Text Analysis Conference (TAC) [6]: TAC-2008, TAC-2009, TAC-2010 and TAC-2011.

These datasets respectively contain 48, 44, 46 and 44 clusters of documents and 58, 55, 43 and 51 systems are used to generate summaries. Each cluster forms a topic to be summarized and has 4 reference summaries. There are approximately 10k pairs in each dataset. Each pair is annotated with two human judgment scores: the *Pyramid Score* (Harnly et al., 2005) and the *Responsiveness* (Dang, Owczarzak, et al., 2008). The former measures the proportion of important semantic units (SCUs) in the reference summaries captured by the system summary, while the latter reflects the content coverage and the readability of each summary.

**Machine translation**. Our evaluation corpus is from the WMT-2019 [7] shared task (Li et al., 2019). We consider all the to-English pairs: Chinese, Czech, German, Finnish, Russian, Lithuanian and Kazakh to English. For each language, we use the test set that contains several thousands of reference-candidate pairs annotated with human ratings that assess the translation quality.

## 4.5 RESULTS

Table 4.1 reports the results averaged over the 4 TAC datasets and the 7 WMT to-English language pairs. Details are provided in Appendices B.1 and B.2.

We benchmarked the metrics in terms of Pearson correlations with human scores, runtimes, and numbers of parameters. We used two approaches to compute the Pearson correlations: summary-level (or segment-level) and system-level.

6. https://tac.nist.gov/
7. http://www.statmt.org/wmt19/

In the former approach, a score is attributed to each of the output candidates, while in the latter approach, one single overall score is attributed to the system (by averaging its individual scores).

Rows `a` to `c` correspond to BERTScore with BERT miniatures as the underlying model. They are simple baselines added for the sake of comparison, to see what we get when BERTScore is used with the same number of parameters as FrugalScore.

Rows `d` to `g` correspond to the expensive metrics that are learned by FrugalScore (in the respective sections of the bottom half of the table). They are BERTScore and MoverScore metrics where the underlying model is a large pretrained language model: BERT-Base ($L = 12$, $H = 512$), RoBERTa-Large ($L = 24$, $H = 1024$) (Liu et al., 2019), and DeBERTa-XLarge ($L = 24$, $H = 1536$) (He et al., 2020).

Finally, rows `i` to `xii` correspond to FrugalScore. Subscripts refer to row labels and indicate which metric-model combination was used to annotate pairs. I.e., FrugalScore$_d$ learned the metric of row $d$, i.e., BERTScore with BERT-Base.

First, results show that all FrugalScores, regardless of which metric they learned, significantly outperform the BERTScores with miniature models. These results suggest that FrugalScore is a better approach than using an existing metric with a lightweight underlying model. The reason for this is probably that in FrugalScore, the knowledge of the original unsupervised metric (based on a large model) is explicitly transferred to the miniature via the continuation of its pretraining on the synthetic dataset. That is, the miniature is actually learning a metric. Whereas, on the other hand, plugging a compressed version of a general-purpose language model into the original unsupervised metric just makes it lose expressiveness and capacity.

Second, we can clearly see that FrugalScore retains most of the performance of the original metric, while running several times faster and reducing the number of parameters by several orders of magnitude. On average over all metrics, tasks, and miniatures, FrugalScore retains 96.8% of the original performance, runs 24 times faster, and has 35 times less parameters.

More precisely, on average across all metrics, FrugalScore-Tiny retains 97.7/94.7% of the original performance on TAC (pyramid score/responsiveness), while running 54 times faster and having 84 times less parameters. Its small and medium versions retain near full performance in terms of responsiveness (98 and 97.7%) and even slightly outperform the original metrics in terms of pyramid score, while at the same time reducing the runtime and the number of parameters by 32 (resp. 21) and 13 (resp. 9) times.

On WMT, FrugalScore-Tiny retains 88.58% of the performance of the original metrics, while running 14 times faster (and still having 84 times less parameters), while the small and medium versions of FrugalScore retain 95.71 % and 98.06% of the original performance while still offering a 32 times (resp. 21) speedup and having 13 times (resp. 9) less parameters, on average.

Interestingly, FrugalScore even improves the performance of the original metrics in some cases. For example, on TAC, FrugalScore$_g$ with BERT-Tiny (row x) improves the performance of the original MoverScore metric based on BERT-Base (row g) from 66.5 to 67.3 in terms of pyramid score, while reducing the number of parameters by 25 and running 50 times faster. Other examples, also for TAC with the pyramid score, include FrugalScore$_f$ with BERT-Small (row viii, +1.5 point) and FrugalScore$_f$ with BERT-medium (row ix, +1 point).

Finally, the results of FrugalScore for different miniature sizes show that, on WMT, using larger models always improves performance (e.g., row x → xi → xii). But interestingly, on TAC, this observation does not hold (e.g., row vi → viii → ix), and sometimes, FrugalScore with the smallest miniature (BERT-Tiny) is superior (e.g. rows i and x). This finding suggests that the impact of the pretrained language model size is task-dependent.

To sum up, results clearly show the effectiveness of FrugalScore in learning a cheaper, lighter, and faster version of the original metrics, while retaining most of their original performance. The system-level correlations, provided in Appendices B.3 and B.4, corroborate these positive results.

We also provide the correlations between the original and the learned metrics in Appendices B.5 and B.6. It is interesting to note that a greater correlation with the original metric is not always associated with a better performance. E.g., the tiny version of FrugalScore$_g$ is the best (row x), while it is the less correlated with the original metric.

| | Pretraining Continued | TAC-2008 | TAC-2009 | TAC-2010 | TAC-2011 | Average |
|---|---|---|---|---|---|---|
| TAC-2008 | no | - | $67.7_{0.57}$ | $66.1_{0.18}$ | $63.6_{0.36}$ | 65.8 |
| | yes | | $74.4_{0.13}$ | $71.3_{0.04}$ | $67.3_{0.13}$ | 71.0 |
| TAC-2009 | no | $61.4_{0.41}$ | - | $66.9_{0.24}$ | $62.7_{0.55}$ | 63.7 |
| | yes | $65.8_{0.25}$ | | $70.7_{0.32}$ | $66.0_{0.18}$ | 67.5 |
| TAC-2010 | no | $59.7_{0.47}$ | $67.3_{0.7}$ | - | $62.4_{0.47}$ | 63.1 |
| | yes | $64.7_{0.19}$ | $74.3_{0.24}$ | | $67.2_{0.11}$ | 68.7 |
| TAC-2011 | no | $57.6_{1.39}$ | $64.7_{1.03}$ | $66.5_{0.66}$ | - | 62.9 |
| | yes | $63.9_{0.31}$ | $72.0_{0.44}$ | $71.6_{0.44}$ | | 69.2 |

Table 4.2 – Summary-level Pearson correlations with human judgments (Pyramid scores), averaged over 3 runs (standard deviation as subscript). Rows correspond to the training sets and columns to the test sets.

## 4.6 FINE-TUNING ON HUMAN ANNOTATIONS

We test two hypotheses in this section: (1) whether fine-tuning on a human-annotated dataset is beneficial, and (2) when fine-tuning on human annotations, whether continuing pretraining on our synthetic dataset is useful.

Because we cannot use the same dataset for fine-tuning and evaluation, we fine-tune a BERT-Small on each year of TAC 2008-2011 for 4 epochs, using two other years as the validation set, and the remaining year as the test set. The best epoch is selected based on validation performance. We use a batch size of 32 and a learning rate of 2e-5 that linearly decreases to zero. Finally, we experiment with two scenarios: fine-tuning the miniature directly without continuing its pretraining on our synthetic dataset, and fine-tuning it after the pretraining continuation (with annotations generated by BERTScore-BERT-Base).

**Results**. Results are reported in Table 4.2 in terms of summary-level Pearson correlations with human evaluations (Pyramid), averaged over 3 runs with different random seeds.

First, it is obvious that everywhere, continuing the pretraining on our synthetic dataset leads to a significant boost in performance. This is in accordance with Sellam, Das, and Parikh (2020a), who found that pretraining was beneficial even in a supervised setting.

Second, even if a direct comparison is not possible, we can remark when looking at the TAC Pyramid score of row ii) in Table 4.1 (FrugalScore$_d$-BERT-Small) that fine-tuning after pretraining seems very beneficial too. Indeed, after fine-tuning, we reach on average 71, 67.5, 68.7, and 69.2 (depending on the split), which represents overall a gain of 4.4 points over the non-fine-tuned model (score of 64.7).

## 4.7 IMPACT OF DATA SOURCES

To test the importance of each data source introduced in subsection 4.3.1, we created a training set containing sequence pairs uniformly and equally sampled from each source. We annotated these pairs with the BERTScore-BERT-Base metric and we used them to continue the pretraining of a BERT-Small miniature.

We also considered pairs drawn at random from the pairs generated with the other strategies. The motivation for random pairs was to sample "negative examples", as seeing only "positive examples" (pairs of related sequences) could bias the learned metric towards considering any two unrelated sequences as similar.

We then continued the pretraining of the BERT-Small miniature four times, excluding each time the pairs coming from a specific data source. We evaluated the learned metric on TAC-2008 to 2011 and on WMT-2019. Figure 4.1 shows the average improvements in the Pearson correlation with human judgments relative to training

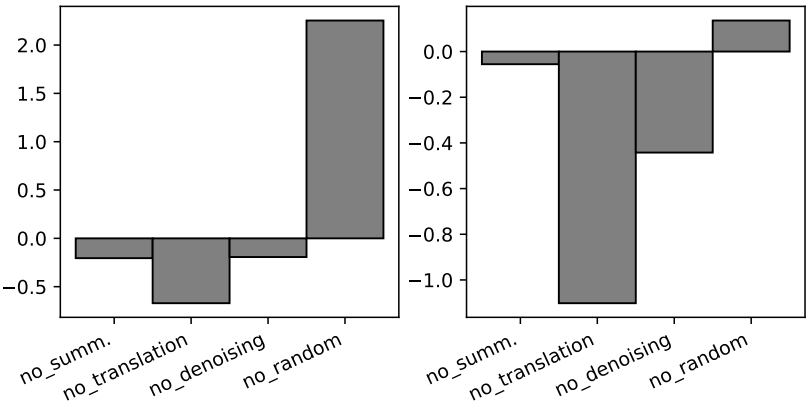


Figure 4.1 – Relative improvement in Pearson correlation compared to a dataset covering all sources. Left: TAC. Right: WMT.

a model on all sources. Note that when training on all four sources, we sampled 30k pairs from each source (120k total), and when excluding a source, we sampled 40k pairs from each source (120k total).

We can clearly see that excluding the random pairs improves performance while excluding any of the other data sources decreases performance. In other words, all our data sources are beneficial, and it is not necessary to add "negative examples". We hypothesise that this is due to the fact that NLG datasets typically do not contain completely unrelated pairs of sentences. Interestingly, the pairs generated with the backtranslation strategy have the greatest impact on performance.

| Dataset: | Web | Asv | MUS | Pas | Fli | mSu | Rea | SumE | OpQA | OkVQA |
|---|---|---|---|---|---|---|---|---|---|---|
| Task: | DTG | Sim | Sim | ImCa | ImCa | Sum | Sum | Sum | QA | VQA |
| ROUGE-1 | 69.7 | 47.9 | 41.6 | 43.4 | 37.1 | 50.7 | 47.4 | **17.9** | **35.5** | 19.5 |
| ROUGE-L | 61.2 | 43.0 | 40.9 | 41.4 | 38.2 | 52.9 | 42.6 | 15.7 | 35.4 | 19.5 |
| BLEU | 53.6 | 29.9 | 32.7 | 29.5 | 32.2 | 48.0 | 37.6 | 7.0 | 10.8 | 19.0 |
| METEOR | 67.9 | 52.2 | 40.6 | 42.9 | 41.6 | 46.3 | **53.7** | 16.2 | 33.7 | 5.7 |
| BLEURT | 77.1 | **68.1** | 37.7 | 51.6 | 53.2 | 46.3 | 34.1 | 9.8 | 22.6 | 15.2 |
| Nubia | **78.7** | 62.2 | 43.5 | **52.9** | **58.6** | 37.7 | 12.5 | 6.0 | 33.5 | 13.8 |
| BERTScore (row e) | 56.1 | 64.5 | 41.7 | 33.1 | 45.0 | **57.0** | 40.8 | 11.5 | 9.5 | 12.0 |
| FrugalScore$_{e,tiny}$ (iv) | 71.5 | 43.9 | **51.9** | 37.3 | 49.8 | 44.1 | 43.0 | 16.9 | 29.2 | 23.3 |
| FrugalScore$_{e,small}$ (v) | 72.1 | 52.9 | 47.9 | 37.7 | 54.1 | 48.4 | 43.6 | 12.1 | 28.2 | 19.3 |
| FrugalScore$_{e,medium}$ (vi) | 73.3 | 58.4 | 45.7 | 38.7 | 54.2 | 52.4 | 43.9 | 13.4 | 24.7 | **20.1** |

Table 4.3 – **Correctness dimension**: Pearson coefficient (computed given a single human reference) between automatic metrics and human judgement for Correctness on the 10 human evaluation datasets. Results in top bloc were taken from the BEA-Metrics paper.

| Dataset: | Web | Asv | Asv | MUS | mSu | SumE | SumE | SumE | OpQA | OkVQA | OkVQA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Task: | DTG | Sim | Sim | Sim | Sum | Sum | Sum | Sum | QA | VQA | VQA |
| Dim: | Flu | Flu | Sim | Flu | Rel | Rel | Coh | Flu | Obv | Pos | Obv |
| ROUGE-1 | 55.4 | 33.7 | 31.2 | 26.1 | 50.9 | 33.3 | 18.8 | 13.6 | **41.7** | 13.8 | **28.9** |
| ROUGE-L | 52.0 | 31.8 | 28.5 | 25.4 | 52.8 | 26.8 | 18.2 | 12.1 | 41.6 | 13.8 | **28.9** |
| BLEU | 43.8 | 25.6 | 23.5 | 22.4 | 49.6 | 20.8 | 13.0 | 6.9 | 12.0 | 9.7 | 23.5 |
| METEOR | 53.8 | 35.3 | 31.7 | 26.6 | 54.9 | 30.4 | 15.3 | 11.8 | 40.1 | 1.0 | 4.3 |
| BLEURT | **64.0** | **55.3** | 48.7 | 31.8 | 43.5 | 28.1 | 14.4 | 14.8 | 21.2 | **26.7** | 23.6 |
| Nubia | 50.4 | 43.6 | 39.3 | 24.3 | 29.4 | 14.4 | 7.2 | 7.5 | 38.2 | 8.9 | 20.2 |
| BERTScore (row e) | 54.7 | 50.2 | **46.6** | 26.8 | **56.9** | **37.2** | **34.2** | 15.6 | 6.9 | 13.1 | 16.2 |
| FrugalScore$_{e,tiny}$ (iv) | 54.1 | 32.0 | 30.9 | **32.4** | 43.6 | 30.5 | 18.5 | **16.2** | 29.4 | 23.7 | 28.4 |
| FrugalScore$_{e,small}$ (iv) | 55.8 | 36.5 | 36.0 | 29.7 | 46.6 | 26.5 | 15.5 | 13.7 | 27.4 | 22.2 | 26.6 |
| FrugalScore$_{e,medium}$ (iv) | 58.2 | 41.6 | 40.9 | 28.7 | 51.4 | 27.3 | 16.4 | 14.6 | 23.0 | 20.3 | 25.9 |

Table 4.4 – **Non Correctness dimensions**: Pearson coefficient (computed given a single human reference) between automatic metrics and human judgement for the dimensions other than Correctness. Flu, Sim, Rel, Coh, Obv, and Pos denote fluency, simplicity, relevance, obviousness, and possibility, respectively. Results in top bloc were taken from the BEAMetrics paper.

## 4.8 BEAMETRICS

In the end, we wanted to evaluate the generalizability and reliability of our FrugalScore. To this purpose, we use BEAMetrics (Scialom and Hill, 2021): a multi-task, multi-lingual, and multi-dimensional benchmark to evaluate automatic NLG metrics, covering a variety of datasets for the tasks: data-to-text (DTG), text simplification (Sim), image captioning (ImCa), summarization (Sum), and (visual) question answering (VQA/QA). We compare our FrugalScore$_e$ metrics (row iv, v, and vi in Table 4.1) with many state-of-the-art unsupervised and supervised alternatives. Results for correctness dimension and the other non-correctness dimensions in Table 4.3 and 4.4 show that our metrics are competitive. Notably, in terms of factual correctness, FrugalScore shows a stable and a higher performance compared to the other metrics, which demonstrates its independence and robustness to various tasks and datasets. For the sake of completeness, we also report the results of BERTScore with RoBERTa-Large (row e in Table 4.1), which confirms the effectiveness of our metric distillation approach, FrugalScore$_e$ is on par (and sometimes outperforms) its teacher metric.

## 4.9 CONCLUSION

We proposed FrugalScore, an approach to learn a fixed, low-cost version of any expensive NLG evaluation metric. Experiments on summarization and translation

tasks show that our FrugalScore versions of BERTScore and MoverScore retain most of the original performance in terms of the correlation with human judgments, while running several times faster and having several orders of magnitude less parameters. On average over all learned metrics, tasks, and variants, FrugalScore retains 96.8% of the performance, runs 24 times faster, and has 35 times less parameters than the original metrics.

## 4.10 ACKNOWLEDGMENTS

This work was supported by the SUMM-RE project (ANR-20-CE23-0017). We thank the anonymous reviewers for their feedback.

# 5

# DATSCORE

The rapid development of large pretrained language models not only has revolutionized the field of Natural Language Generation (NLG), but also its evaluation. Inspired by the recent work of BARTScore: a metric leveraging the BART language model to evaluate the quality of generated text from various aspects, we introduce DATScore. DATScore uses data augmentation techniques to improve the evaluation of machine translation. Our main finding is that introducing data augmented translations of the source and reference texts is greatly helpful in evaluating the quality of the generated translation. We also propose two novel score averaging and term weighting strategies to improve the original score computing process of BARTScore. Experimental results on WMT show that DATScore correlates better with human meta-evaluations than the other recent state-of-the-art metrics, especially for low resource languages. Ablation studies demonstrate the value added by our new scoring strategies. Moreover, we report the performance of DATScore on 3 other NLG tasks than translation in our extended experiments. Code is publicly available [1].

## 5.1 INTRODUCTION

Massive pretrained language models have brought significant improvement to NLG tasks (Lewis et al., 2020). Recent systems can even generate texts of higher quality than human annotated ones (Peyrard, 2019). At the same time, standard metrics, such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004a), for translation and summarization respectively, have not evolved for the past two decades (Bhandari et al., 2020). These metrics rely on surface lexicographic matches, making them particularly unsuitable for evaluating modern systems operating with embeddings at the semantic level, that often generate paraphrases (Ng and Abrecht, 2015b). To address this issue, many metrics have been proposed (Sai, Mohankumar, and Khapra, 2022), but none of them were widely adopted until the release of BERTSore (Zhang et al., 2019) and MoverScore (Zhao et al., 2019a). These metrics take advantage of large pretrained language models like BERT (Devlin et al., 2019), which are now being used in nearly all NLP tasks (Qiu et al., 2020; Min et al., 2021).

In this work, we focus on the task of evaluating machine translation. We propose an extension of BARTScore (Yuan, Neubig, and Liu, 2021), a recent metric exploit-

1. link will be provided upon acceptance.

ing the BART seq2seq language model (Lewis et al., 2020) to evaluate the quality of generated text from various aspects. BARTScore covers four evaluation facets: Faithfulness, Precision, Recall, and F-score, derivated from different generation directions between the *source* text, the *hypothesis* (the text generated by a system given the source), and the *reference* (the reference text for the generation, often provided by human annotators). The scores are obtained by pairing the 3 entities differently at the input or the output side of a trained seq2seq model for fetching conditional generation probabilities.

On the basis of BARTScore, and motivated by the general idea and positive effect of data augmentation techniques, we found that adding augmented, translated copies of the source and reference texts in BARTScore, can greatly help in evaluating the quality of the hypothesis translation. We also propose two novel score averaging and term weighting strategies to improve the original score computing process of BARTScore. Results and ablation studies show that our metric DATScore (Data Augmented Translation Score) outperforms the other recent state-of-the-art metrics, and our new scoring strategies are effective. Moreover, the performance of DATScore is also reported on 3 other NLG tasks than translation: data-to-text, summarization, and image captioning.

To the best of our knowledge, no prior work has been done on leveraging data augmentation techniques for untrained NLG evaluation metrics. Our work will help filling this gap. Our contributions include:
1) Inspired by BARTScore, we developed DATScore that incorporates augmented data translated from the source and reference texts. DATScore is an untrained and unsupervised translation evaluation metric, that offers a larger performance boost in evaluating low resource language generation. In contrast to other widely adopted metrics, DATScore can efficiently incorporate both the source and reference texts in the evaluation.
2) We introduced a novel one-vs-rest method to average the scores for different generation directions with different weights, which improves over the simple arithmetic averaging method used in BARTScore.
3) We proposed a novel entropy-based scheme for weighting the target generated terms, so that higher informative tokens receive more importance in accounting the score, which outperforms the naive uniform weighting employed in BARTScore.

## 5.2 RELATED WORK

### 5.2.1 *Translation evaluation metrics*

BLEU (Papineni et al., 2002) is the de facto metric for evaluating machine translation. It simply calculates $n$-gram matching between the reference and the hypothesis

using precision scores with a brevity penalty. METEOR (Banerjee and Lavie, 2005) was developed to address two drawbacks of BLEU. It is F-score based (thus taking recall into account) and allows for a more relaxed matching, based on three forms: extract unigram, stemmed word, and synonym with WordNet (Miller, 1994). Apart from the above word-based metrics, some approaches operate at the character level. For example, chrF (Popović, 2015) computes the overall precision and recall over the character $n$-grams with various values of $n$. More recently, static word embeddings (Mikolov et al., 2013a) have enabled capturing the semantic similarity between two texts possible, of what the historical metrics are incapable. Several metrics have been proposed to incorporate word vectors. For example, MEANT 2.0 (Lo, 2017) evaluates translation adequacy by measuring the similarity of the semantic frames and their role fillers between the human and machine translations.

Lately, pretrained language models have become popular, because they provide context-dependent embeddings. This proved beneficial to all NLP tasks, but also to evaluation metrics. For example, using a modified version of the Word Mover's Distance (Kusner et al., 2015), the Sentence Mover's Similarity (Clark, Celikyilmaz, and Smith, 2019a) measures the minimum cost of transforming one text into the other as the evaluation score, where sentences are represented as the average of their ELMo word embeddings (Peters et al., 2018a). BERTr (Mathur, Baldwin, and Cohn, 2019a) computes approximate recall based on the pairwise cosine similarity between the BERT word embeddings (Devlin et al., 2019) of two translations. UniTE (Wan et al., 2022) proposes a unified framework for modeling three evaluation prototypes: estimating the quality of the translation hypothesis by comparing it with reference-only, source-only, or source-reference-combined data. UniTE is built upon XLM-R multilingual language model (Conneau et al., 2020).

Among several alternatives, BERTSore (Zhang et al., 2019) and MoverScore (Zhao et al., 2019a) have received more attention, and been adopted for reporting results in recent NLG publications (Lin et al., 2022; Weston et al., 2022). They both are unsupervised, general-purpose metrics and leverage BERT-like language models, however, with one difference lying in the similarity function for matching the two sequence representations. BERTScore greedily matches each token from one sequence to the single most similar token in the other sequence, in terms of cosine similarity of their token embeddings. While MoverScore conducts soft one-to-many matching using an $n$-gram generalization of the Word Mover's Distance (Kusner et al., 2015).

Finally, the work closely related to ours is BARTScore (Yuan, Neubig, and Liu, 2021). Different with all the above metrics trying to match tokens or their embeddings, BARTScore proposes a novel conceptual view. It treats the evaluation of generated text as a text generation problem, with the help of a pretrained seq2seq model BART (Lewis et al., 2020). At the time of writing, this metric represents the state-

of-the-art in the NLG evaluation. We will provide more details about it in Section 5.3.

### 5.2.2 *Data augmentation*

As deep learning models are often heavily reliant on large amounts of training data, a common attempt to get around the data scarcity problem is by applying data augmentation techniques (Shorten and Khoshgoftaar, 2019). These techniques increase the size of the training set by making slightly modified copies of already-existing instances or by creating new, synthetic ones. Such augmented data have proven to be beneficial to the training of models in a wide variety of contexts, from computer vision (Shorten and Khoshgoftaar, 2019) to speech recognition (Bird et al., 2020), to NLP (Feng et al., 2021), as it acts as a regularizer and helps reduce overfitting (Krizhevsky, Sutskever, and Hinton, 2012). For dealing with textual data, a suite of augmentation techniques exists. To name only a few, backtranslation (Sennrich, Haddow, and Birch, 2016) translates a text into an intermediate language and then back into the original language, as a way of paraphrasing the initial text. Contextual augmentation (Kobayashi, 2018) generates augmented samples by randomly replacing words with others drawn following the in-context word distribution of a recurrent language model. SeqMix method (Guo, Kim, and Rush, 2020) creates synthetic examples by softly mixing parts of two sentences via a convex combination.

Data augmentation has also been applied to the field of NLG evaluation metrics. BLEURT (Sellam, Das, and Parikh, 2020b) is a supervised metric, i.e., it requires to be finetuned on human meta-evaluations. Before finetuning, BLEURT creates an augmented synthetic dataset made by perturbing Wikipedia sentences with BERT mask-filling, backtranslation, and random word dropping techniques. The data are then annotated with some automatic numerical and categorical signals as pretraining labels. FrugalScore (Kamal Eddine et al., 2022a) proposes the first knowledge distillation approach for NLG evaluation metrics, to alleviate the significant requirement of computational resources by the heavy metrics based on large pretrained language models (e.g., BERTScore and MoverScore). Unlike BLEURT, it is purely trained on a synthetic dataset consisting of pairs of more or less related sentences, created via various data augmentation techniques (e.g., paraphrasing with backtranslation, perturbation then denoising, etc.). The sentence pairs for training the student model are annotated with scores given by the metrics to be learned.

**Differences.** Note that BLEURT and FrugalScore use augmented data for the purpose of training their parameterized metric models, while our DATScore is an untrained and unsupervised metric not requiring human judgments for training and using augmented translation for the sole purpose of scoring.

## 5.3 DATSCORE

As mentioned in Subsection 5.2.1, BARTScore is not based on matching tokens nor their embeddings as the other evaluation metrics. Rather, it uses a novel approach by framing the evaluation of generated text as a text generation problem. Assuming first a pretrained seq2seq model is "perfect" (e.g., BART), BARTScore uses directly the model's conditional probability of generating a provided target text $Y$ given a provided input text $X$, as the evaluation score of the generation direction $X \rightarrow Y$. For example, $Y$ corresponds to a translation hypothesis generated by any system, and $X$ is the reference. If $Y$ is of high quality, then by providing the pair to the pretrained BART model, the estimated conditional generation probability (evaluation score) $P(Y|X)$ should be high.

Therefore, with placing differently the *source* (Src), the *reference* (Ref), and the *hypothesis* (Hypo) in pair at the input or the output side of the trained seq2seq model for fetching conditional generation probabilities, BARTScore deviates three different generation directions illustrated as dashed arrows in Figure 5.1. The conditional probabilities associated with the directions are denoted as: Precision (*Ref→Hypo*), Recall (*Hypo→Ref*) and Faithfulness [2] (*Src→Hypo*). Additionally, a F-score, the arithmetic average of Precision and Recall.

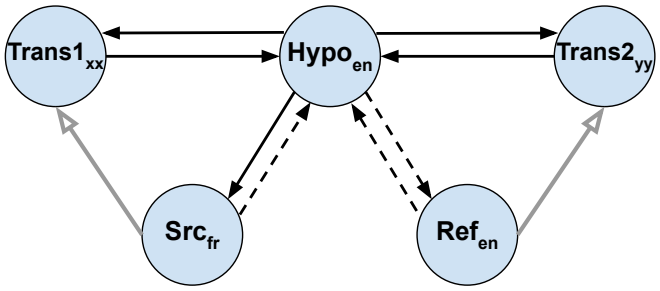


Figure 5.1 – Dashed arrows denote the generation directions covered by BARTScore. Solid black arrows indicate our newly introduced directions for calculating DATScore of the example *hypothesis* in English ($\text{Hypo}_{en}$). $\text{Trans1}_{xx}$ and $\text{Trans2}_{yy}$ represent data *augmented translations* in any languages xx and yy, obtained by applying a translation model (grey arrows) to the example *source* in French ($\text{Src}_{fr}$) and example *reference* in English ($\text{Ref}_{en}$), respectively.

The score (conditional probability) for the generation direction from a source sequence $X = \{x_t\}_{t=1}^{n}$ to a target sequence $Y = \{y_t\}_{t=1}^{m}$ is calculated as the factorized, weighted log probability over all generation steps:

$$\text{Score}_{X \rightarrow Y} = \sum_{t=1}^{m} w_t \log P(y_t|X, \{y_{t'}\}_{t'=1}^{t-1}; \theta) \tag{5.1}$$

2. BART being a monolingual model, faithfulness is only relevant in the context of abstractive summarization, and its corresponding direction cannot be applied to machine translation evaluation.

where $w_t$ denotes the term importance score to put different emphasis on different target tokens $y_t$. BARTScore simply employs a uniform weighting scheme (all equal to 1). $\theta$ denotes the parameterized seq2seq model.

Our contributions consist of three modifications tailored to machine translation:

**Data augmented translations.** Unlike BARTScore, we employ M2M-100 (Fan et al., 2021), a non-English-centric multilingual machine translation system as our backbone seq2seq model, due to its superior performance. As our main contribution, we translate the source (e.g., $\text{Src}_{fr}$ in Figure 5.1) and the reference ($\text{Ref}_{en}$) into any languages as our augmented data ($\text{Trans1}_{xx}$ and $\text{Trans2}_{yy}$) for evaluating the hypothesis ($\text{Hypo}_{en}$). In addition to the three directions covered by BARTScore, our metric takes into consideration all generation directions centered on the hypothesis connecting the source, the reference, and the two data augmented translations, i.e., in total 8 directions as the black (dashed and solid) arrows depicted in Figure 5.1. DATScore is calculated as the weighted average of the scores associated with all the directions:

$$\text{DATScore} = \sum_{X,Y} w_{X\to Y}\text{Score}_{X\to Y}; X \neq Y \tag{5.2}$$

where $w_{X\to Y}$ denotes the weight of the direction $X \to Y$, as detailed below.

**One-vs-rest score averaging method.** We observed empirically that sometimes, one direction score might strongly disagree with the others, likely being an outlier (failed evaluation). This may significantly affect the final DATScore correlations with the human meta-evaluations, if a simple arithmetic averaging method is applied (like BARTScore in computing F-score). To reduce this effect, we weight each direction with the sum of the Pearson correlations of its scores with the scores of all the other directions:

$$w_{X\to Y} = \sum_{X',Y'} Corr(\text{Score}_{X\to Y}, \text{Score}_{X'\to Y'}) \tag{5.3}$$
$$\text{s.t.} \quad (X,Y) \neq (X',Y')$$

This one-vs-rest method will assign a low weight to the direction score correlated badly with the rest scores, thus reduce its negative effect to the averaging result.

**Entropy-based term weighting scheme.** BARTScore gives an equal weight $w_t$ to every token in Equation 5.1 (uniform weighting). Instead, we introduce a novel scheme to give different importance to different target tokens $y_t$, based on the entropy:

$$w_t = -\sum_{i=1}^{v} \text{P}_\text{t}(z_i)\text{log}\text{P}_\text{t}(z_i) \tag{5.4}$$

where $v$ denotes the size of output generation vocabulary. $\text{P}_\text{t}(z_i)$ represents the probability of the $i$-th token in the vocabulary at time step $t$. Our assumption is that, when

the model is very confident in generating the target token (low entropy), then this token is non-informative (e.g., stopword). While when the model is less confident (higher entropy), the target word is more informative, and then a higher weight should be assigned.

The effectiveness of all our choices regarding the above contributions is shown by our ablation studies (see Section 5.6).

## 5.4 EXPERIMENTS

### 5.4.1 *Experimental settings*

We benchmark DATScore on two commonly used meta-evaluation datasets for machine translation metrics: WMT17 (Bojar, Graham, and Kamran, 2017) and WMT18 (Ma, Bojar, and Graham, 2018) consisting of multiple `to_English` and `from_English` language pairs. For each pair, a few thousand examples are available, each being made of a *source*, a *reference*, a *hypothesis* and a *label* produced by human annotators, assessing the quality of the system generated *hypothesis*. Depending on the *label* type, we use Kendall's Tau $\tau$ correlations or absolute Pearson $|r|$ correlations. The former is used when relative ranking is provided, and the latter in the case of direct assessment. We adopt the Kendall's Tau-like formulation proposed in (Bojar, Graham, and Kamran, 2017):

$$\tau = \frac{|Concordant| - |Discordant|}{|Concordant| + |Discordant|} \tag{5.5}$$

where $|Concordant|$ is the number of examples on which the metric agrees with the human relative ranking, and $|Discordant|$ is the number of examples when they disagree.

To compute DATScore, two M2M-100 models: M2M-100_418M [3] and M2M-100_1.2B [4] are adopted (418M and 1.2B refer to the model sizes). They are finetuned to translate a source text to a target text by providing the source language code (e.g. "fr") at the beginning of the encoder input sequence, and a target language code at the beginning of the decoder input sequence. In our experiments, when English is the target language (`to-English`), we choose English for Trans1 and Spanish for Trans2 (see Figure 5.1). Otherwise, whenever English is the source language (`from-English`), we choose Spanish for Trans1 and English for Trans2. This choice is motivated by the fact that English and Spanish are the top two represented languages in the training set of M2M-100 (Fan et al., 2021).

3. https://huggingface.co/facebook/m2m100_418M
4. https://huggingface.co/facebook/m2m100_1.2B

| Metric | | Model | \|r\|:cs→en / τ:en→cs | \|r\|:de→en / τ:en→de | \|r\|:fi→en / τ:en→fi | \|r\|:lv→en / τ:en→lv | \|r\|:ru→en / - | \|r\|:tr→en / τ:en→tr | \|r\|:zh→en / - | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|
| BLEU | 1a) | N/A | 34.4/22.0 | 36.6/23.6 | 44.4/42.1 | 32.1/21.5 | 41.3/- | 44.1/33.6 | 44.0/- | 37.8/27.3 |
| BERTScore | 1b) | RL/mBERT | 71.0/43.8 | **74.5**/40.4 | 83.3/58.8 | 75.6/46.6 | 74.6/- | 75.1/57.1 | 77.5/- | 75.9/49.3 |
| MoverScore | 1c) | BB/mBERT | 66.6/38.3 | 70.6/35.9 | 82.2/54.2 | 71.7/37.8 | 73.7/- | 76.1/49.8 | 74.3/- | 73.6/43.2 |
| BARTScore | 1d) | BL+para/mBART | 68.4/39.0 | 70.8/33.4 | 79.4/50.4 | 74.9/50.4 | 71.8/- | 73.9/53.8 | 76.0/- | 73.6/45.4 |
| | 1e) | M2M-100_418M | 65.9/45.0 | 66.1/44.5 | 79.9/59.2 | 71.7/40.3 | 69.0/- | 71.8/70.9 | 71.6/- | 70.9/52.0 |
| | 1f) | M2M-100_1.2B | 67.4/49.6 | 69.3/49.2 | 80.7/63.5 | 73.7/46.9 | 70.4/- | 71.6/**72.5** | 73.0/- | 72.3/56.3 |
| DATScore | 1g) | M2M-100_418M | 68.6/51.1 | 68.5/48.1 | 82.0/63.7 | 74.7/48.3 | 73.0/- | 77.6/70.9 | 76.5/- | 74.4/56.4 |
| | 1h) | M2M-100_1.2B | **71.3**/**53.9** | 72.9/**52.2** | **83.5**/**66.3** | **76.8**/**52.0** | **75.9**/- | **78.1**/70.9 | **77.7**/- | **76.6**/**59.1** |

Table 5.1 – Absolute Pearson correlation (|r|) for `to-English` and Kendall correlations (τ) for `from-English` with segment-level human scores on WMT17. BB stands of Bert-Base, RL for RoBERTa-Large and BL for BART-Large.

| Metric | | Model | $\tau$:cs→en / $\tau$:en→cs | $\tau$:de→en / $\tau$:en→de | $\tau$:et→en / $\tau$:en→et | $\tau$:fi→en / $\tau$:en→fi | $\tau$:ru→en / $\tau$:en→ru | $\tau$:tr→en / $\tau$:en→tr | $\tau$:zh→en / $\tau$:en→zh | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|
| BLEU | 2a) | N/A | 23.3/38.9 | 41.5/62.0 | 38.5/41.4 | 15.4/35.5 | 22.8/33.0 | 14.5/26.1 | 17.8/31.1 | 24.8/38.3 |
| BERTScore | 2b) | RL/mBERT | 40.4/55.9 | **55.0**/72.7 | 39.7/58.4 | 29.6/53.9 | 35.3/42.4 | 29.2/38.9 | **26.4**/36.1 | 36.5/51.2 |
| MoverScore | 2c) | BB/mBERT | 36.8/44.6 | 53.9/68.4 | 39.4/52.7 | 28.7/50.9 | 27.9/40.1 | **33.6**/32.5 | 25.6/35.2 | 35.1/46.3 |
| BARTScore | 2d) | BL+para/mBART | 39.6/50.2 | 54.7/65.0 | 39.4/53.3 | 28.9/57.2 | 34.6/37.0 | 27.4/37.7 | 24.9/32.4 | 35.6/47.5 |
| | 2e) | M2M-100_418M | 36.3/55.4 | 53.5/72.2 | 37.6/58.4 | 26.3/60.2 | 33.4/44.4 | 26.8/45.1 | 23.4/31.3 | 33.9/52.4 |
| | 2f) | M2M-100_1.2B | 38.4/**63.5** | 54.6/**76.2** | 39.2/63.2 | 27.9/64.5 | 35.7/45.6 | 28.5/50.2 | 24.3/34.7 | 35.5/56.8 |
| DATScore | 2g) | M2M-100_418M | 38.6/53.5 | 53.5/71.3 | 39.3/64.0 | 28.4/62.2 | 34.9/44.4 | 28.5/47.9 | 25.3/34.0 | 35.5/53.9 |
| | 2h) | M2M-100_1.2B | **40.7**/61.9 | 54.9/**76.2** | **40.5**/**68.2** | **30.4**/**67.9** | **36.4**/**46.2** | 31.0/**52.7** | 26.3/**36.6** | **37.2**/**58.5** |

Table 5.2 – Kendall correlations ($\tau$) for `to-English` and `from-English` with segment-level human scores on WMT18. BB stands of Bert-Base, RL for RoBERTa-Large and BL for BART-Large.

### 5.4.2 *Main results*

We compare the performance of our metric against BLEU and three other reference-based unsupervised metrics: BERTScore [5], MoverScore [6] and BARTScore [7] (detailed in Subsection 5.2.1 and Section 5.3), using their official implementations. Experimental results are reported in Table 5.1 and 5.2. Following their original settings, we use different underlying language models for each baseline metric. For BERTScore and MoverScore, RoBERTa-Large (RL; Liu et al., 2019) and Bert-Base (BB) are used respectively when we evaluate `to-English` translations, and mBERT (Devlin et al., 2019) for `from-English` translations. In the case of BARTScore, we use a BART-Large (BL) checkpoint (finetuned on CNNDM (See, Liu, and Manning, 2017) and ParaBank2 (Hu et al., 2019) datasets) for evaluating `to-English` translations, and an mBART-50 model (Escolano et al., 2021) for `from-English` translations.

Overall, results show that on average across all language pairs, DATScore significantly outperforms all 4 baseline metrics under their original model settings (rows 1a-1d and 2a-2d). Specifically, with respect to the best performed baseline BERTScore (row 1b and 2b), our metric provides a performance boost of 0.7 for `to-English` case and of 9.8 for `from-English` case on WMT17 dataset in Table 5.1, and achieves a gain of 0.7 and of 7.3 respectively on WMT18 dataset in Table 5.2. These averaging results demonstrate the superiority and applicability of DATScore in evaluating general machine translations of many languages. Moreover, it's interesting to note that our improvement is much more significant in `from-English` case, which makes DATScore particularly well-suited to evaluate hypothesis translations in non-English languages, often with low resource. We hypothesize that this is due to the inconsistency of underlying language models. The baselines adopt a monolingual model for evaluating English, but a multilingual one for non-English languages. However, DATScore uses a single multilingual M2M-100 model for both cases. It is known that, in general, monolingual models outperforms multilingual competitors. Thus, it is reasonable that when comparing multilingual-based DATScore against monolingual baselines in the `to-English` case, DATScore achieves a smaller improvement than in the other `from-English` case, where the comparison is fairer (multilingual vs. multilingual).

By looking across specific language pairs and directions, we observe DATScore constantly performs better than 4 baseline metrics with a few exceptions, i.e., de ⇢ en (-1.6) in Table 5.1, and de ⇢ en (-0.1), tr ⇢ en (-2.6), and zh ⇢ en (-0.1) in Table 5.2. Despite these small drops in the performance, DATScore brings a larger margin of

5. https://github.com/Tiiiger/bert_score
6. https://github.com/AIPHES/emnlp19-moverscore
7. https://github.com/neulab/BARTScore

| Metric | Model | WebNLG | | |
|---|---|---|---|---|
| | | SEMA | GRAM | FLU |
| BLEU | N/A | 45.5 | 36.0 | 34.9 |
| BERTScore | RoBERTa-Large | 56.1 | 60.8 | 54.8 |
| MoverScore | BERT-Base | -9.9 | -27.8 | -20.6 |
| BARTScore | BART-Large+para | **71.9** | 61.3 | 57.4 |
| | M2M-100_418M | 64.9 | 62.8 | 56.0 |
| | M2M-100_1.2B | 66.1 | 63.9 | 57.2 |
| DATScore | M2M-100_418M | 69.9 | 62.9 | 57.2 |
| | M2M-100_1.2B | 70.4 | **63.7** | **57.9** |

Table 5.3 – Pearson correlation results on WebNLG dataset.

improvement in most cases, such as en ⇢ tr up to 13.8 both on WMT17 and WMT18 datasets.

In the end, for the sake of having a complete comparison, we additionally evaluate BARTScore[8] with M2M-100_418M and M2M-100_1.2B models (row 1e, 1f, 2e, and 2f) that are used as DATScore's underlying models. Results show that, only in the `from-English` case, while they bring improvement compared to the vanilla BARTScore (row 1d and 2d), they are not able to yield as big of a gain as our metric, indicating that our achieved improvement is not solely due to the underlying language model, but also to taking additional generation directions into account, including those related to data augmented translations.

## 5.5 OTHER NLG TASKS

In addition to machine translation, our main focus, we evaluate DATScore on other NLG tasks, including data-to-text generation, abstractive summarization, and image captioning. To work around with the different modalities of source inputs represented in these tasks (e.g., not able to create a data augmented translation with an image), we adapt DATScore to only consider 4 generation directions: *Hypo↔Ref* and *Hypo↔Trans2*.

**Data-to-text.** Table 5.3 shows the performance of DATScore compared to the other baselines on the WebNLG data-to-text dataset (Shimorina et al., 2018), which con-

8. The official implementation of BARTScore is slightly modified to take into account the languages tokens when using a multilingual model.

| Metric | Model | REALSumm | SummEval | | | |
|---|---|---|---|---|---|---|
| | | COV | COH | CONS | FLU | REL |
| BLEU | N/A | 37.9 | 11.8 | 6.3 | 7.7 | 18.6 |
| BERTScore | RoRERTa-Large | 41.2 | **33.9** | 10.5 | **15.0** | **35.9** |
| MoverScore | BERT-Base | 44.1 | 14.4 | **14.7** | 13.8 | 29.1 |
| BARTScore | BART-Large+para | 31.7 | 20.8 | -3.5 | 6.7 | 22.2 |
| | M2M-100_418M | 30.1 | 14.8 | -2.3 | 3.0 | 19.8 |
| | M2M-100_1.2B | 32.0 | 17.1 | 1.1 | 6.7 | 22.8 |
| DATScore | M2M-100_418M | 44.7 | 17.1 | 4.4 | 4.6 | 26.3 |
| | M2M-100_1.2B | **45.5** | 19.5 | 6.8 | 8.2 | 30.2 |

Table 5.4 – Pearson correlation results on two summarization datasets: REALSumm and SummEval.

tains 2000 descriptions of structured tables along with their corresponding references. In addition, human assessments covering three dimensions are provided (*semantics*, *grammer*, and *fluency*). The results show that DATScore significantly outperforms all the other metrics in two settings (grammer and fluency) out of three, while being very competitive in the third setting (semantics). Surprisingly, BERTScore is largely behind DATScore, and MoverScore failed to correlate positively with human judgments in all dimensions.

**Summarization.** Table 5.4 shows the evaluation of the different metircs on two summarization meta-evaluation datasets: REALSumm (Bhandari et al., 2020) and SummEval (Fabbri et al., 2021). Both datasets contain a few thousand examples of system-generated summaries and their references. The generated summaries are annotated with *lightweight pyramids* (Shapira et al., 2019) method in the case of REALSumm, while the annotations in SummEval cover four dimensions: *coherence*, *consistency*, *fluency*, and *relevance*. On REALSumm, DATScore has the best performance compared to all the other baselines even when using its smaller version (M2M-100_418M). However, despite its higher correlations compared to BARTScore and MoverScore, DATScore fails to outperform BERTScore on the different dimensions of SummEval.

**Image captioning.** We consider Flickr8K (Hodosh, Young, and Hockenmaier, 2013) and PASCAL-50S (Vedantam, Lawrence Zitnick, and Parikh, 2015), two image captioning datasets. The former is annotated with scores from 1 to 4 assessing the relevance of the captions, and the latter is annotated with relative ranking (i.e., given

| Metric | Model | Flickr8K | PASCAL-50S |
|---|---|---|---|
| | | RELE | RR |
| BLEU | N/A | 13.8 | 8.1 |
| BERTScore | RoBERTa-Large | 46.1 | **33.8** |
| MoverScore | BERT-Base | **52.5** | 33.2 |
| BARTScore | BART-Large+para | 44.8 | 33.1 |
| | M2M-100_418M | 34.3 | 29.6 |
| | M2M-100_1.2B | 34.6 | 26.3 |
| DATScore | M2M-100_418M | 42.6 | 29.6 |
| | M2M-100_1.2B | 45.3 | 31.4 |

Table 5.5 – Pearson correlation Results on two Image Captioning datasets: Flickr8K and PASCAL-50S.

two descriptions which one is better). Table 5.5 shows that in this task, DATScore is competitive to BARTScore and BERTScore. Surprisingly, MoverScore outperforms significantly all the other metrics despite its poor performance on the other datasets.

Finally, although not being the top performing metric across all tasks, DATScore showed an overall stable and competitive performance. Each of the other metrics fails in evaluating generations at least in one of the tasks. For example, BERTScore and MoverScore have a poor performance on the WebNLG dataset. On the other hand, BARTScore fails to correlate well on REALSumm and SummEval, even it is finetuned on an abstractive summarization dataset. This finding suggests that, regardless of being initially designed for machine translation evaluation, DATScore can be safely used to evaluate NLG systems in other tasks for different evaluation dimensions.

## 5.6 ABLATION STUDY

To validate our different choices with regards to DATScore, we conducted ablation studies on:
1) the contributions of all 8 direction scores, results are illustrated in Figure 5.2.
2) the effectiveness of our *one-vs-rest* score averaging and *entropy-based* term weighting strategies (See Section 5.3), results are reported in Table 5.6.

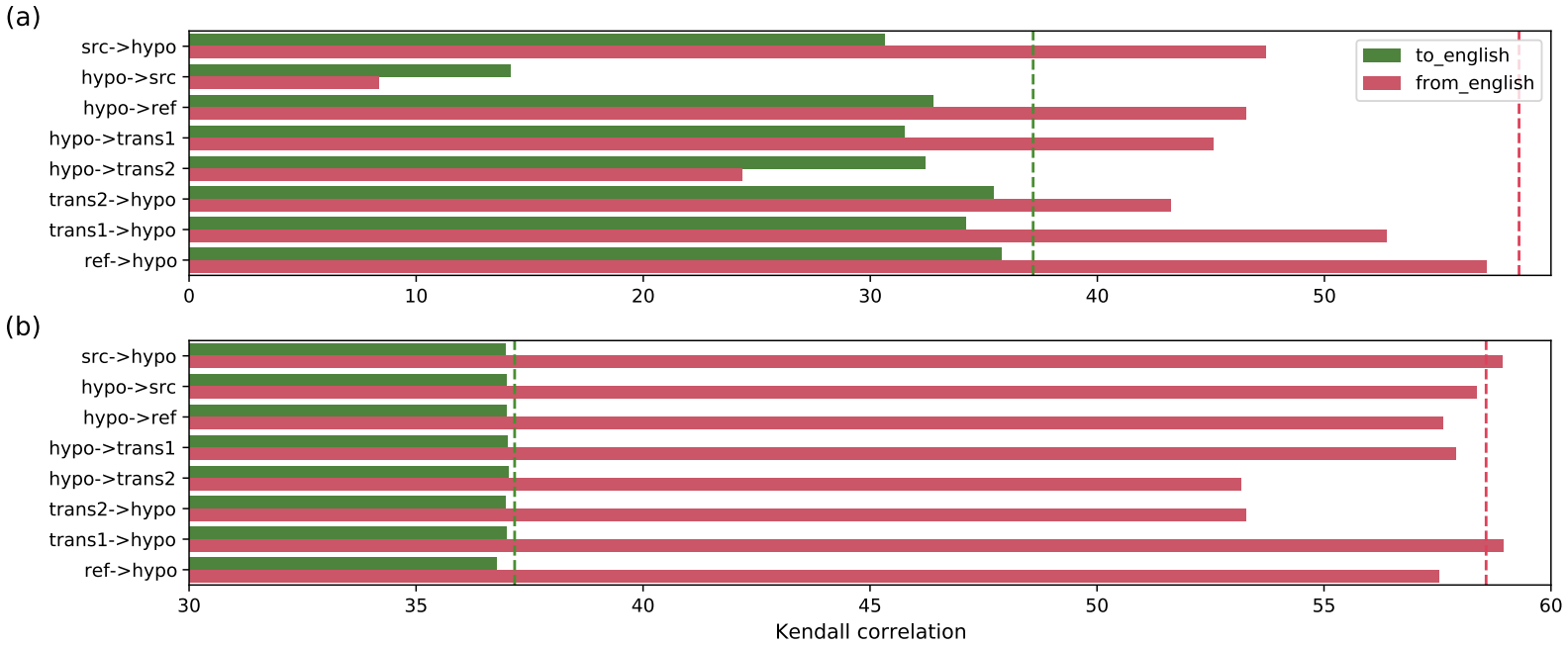


Figure 5.2 – (a): The horizontal bars represent the Kendall correlations of **each individual generation direction**. (b): The horizontal bar represents the Kendall correlation of **a variant of DATScore with excluding the single generation direction** of the line. Both in (a) and (b), the dashed vertical lines represent the Kendall correlation of the vanilla and **complete DATScore**. Correlation results of `to-English` (in green) and `from-English` (in red) cases are calculated w.r.p human judgments, and averaged over all languages pairs. Experiments are conducted on WMT18.

| Entropy-based weighting | One-vs-rest weighting | to_English | from_English |
|---|---|---|---|
| ✓ | ✓ | **37.2** | **58.5** |
| ✓ | ✗ | 37.1 | 58.1 |
| ✗ | ✓ | 36.4 | 55.9 |
| ✗ | ✗ | 36.4 | 56.0 |

Table 5.6 – The average Kendall correlation (to/from)-English when the entropy-based and one-vs-rest weighting are included or excluded. Experiments are conducted on WMT18.

**Contributions of all direction scores.** From Figure 5.2(a), we observe that none of the individual directions (horizontal bars) has a better correlation with human judgements than DATScore (dashed vertical lines), which confirms the importance of our ensemble approach. In Figure 5.2(b), we can see that all variants excluding one direction will lead, in almost all cases, to a drop in the performance, compared to the complete DATScore in which all directions are included. Besides, in the case of `to-English` translations, we can see that the drop in the performance is almost the same for all exclusions of direction. While for `from-English` translations, the largest drop in performance is observed when *Hypo*→*Trans2* and *Trans2*→*Hypo* are excluded. This finding highlights the important contribution of our augmented data, especially in the low resource language settings (`from-English`). In the end, we can see that excluding *Src*→*Hypo* or *Trans1*→*Hypo* directions can lead to a slightly better final score. We leave the investigation of the potential negative impact of the two directions to a future work.

**One-vs-rest and entropy-based weighting strategies.** Table 5.6 shows the performance of DATScore variants with respect to different combinations of applying or not our proposed weighting strategies. Note that, when *one-vs-rest* and *entropy-based* weightings are not applied, they are replaced with a simple uniform averaging approach (as used in BARTScore). A performance drop is observed when excluding one of the two weighting strategies, especially for the entropy-based method, whose inclusion leads to an improvement of 2.5 compared to the uniform weighting. This experiment confirms the positive impact of our proposed weighting methods and motivates future work to further investigate a more elaborated approach in this direction.

## 5.7 CONCLUSION

In this work, we proposed one of the first applications of data augmentation techniques to NLG evaluation. To obtain an evaluation score of the translation hypothesis, our developed metric DATScore additionally leverages newly translated copies augmented from the source and reference texts. We also proposed two novel strategies for score averaging and term weighting to improve the original, naive score computing process of BARTScore, on the basis of which our work is built. Experimental results show that DATScore achieved a higher correlation with human meta-evaluations, in comparison with the other recent state-of-the-art metrics, especially for those less represented languages other than English. Moreover, ablation studies show the effectiveness of our newly proposed score computing approaches, and extended experiments showed an overall stable and competitive performance of DATScore on more NLG tasks.

### LIMITATIONS

In this section, we list some limitations that are worth further investigation in future works:

1) DATScore requires generating additional data augmented translations to perform evaluation. This process might be time consuming depending on the adopted backbone seq2seq model, especially if the original text is long. Thus, the performance scalability can be investigated in future complementary experiments.

2) We made a choice of using English and Spanish to create data augmented translations, for the reason that they are the most two represented languages in training M2M-100 model (see Subsection 5.4.1). This leaves a question to be answered about the performance of DATScore with augmentations varying in different other languages (e.g., Chinese). Moreover, for the sake of simplicity, we decided to only include a single translated copy of the source text and of the reference text each. However, this can be easily extended, and more augmented translations can be created in more languages. We expect to see an improvement in performance with diminishing returns.

3) BARTScore only considers the 8 generation directions centered on the hypothesis connecting with the source, the reference, and the two data augmented translations (see Section 5.3). However, other connections exist between these entities, such as *Src*→*Ref* and *Trans1*→*Src* (see Figure 5.1). Therefore, future research could be dedicated to discovering the effect of these other directions and potentially leveraging them to improve the performance of DATScore.

4) Since our focus was on evaluating machine translation, we naturally chose translation for augmenting the data. However, other data augmentation techniques could be seamlessly integrated into DATScore, such as using a text paraphrasing model (Bandel et al., 2022).

# 6

# CONCLUSION

In this chapter, we wrap up the dissertation by briefly outlining our main contributions described in detail in the preceding chapters and then outlining several intriguing future directions that still need to be investigated.

## 6.1 SUMMARY OF CONTRIBUTIONS

*Pretrained seq2seq models*

We contributed two pretrained seq2seq models, BARThez and AraBART, to French and Arabic languages, respectively. These two models set a new state-of-the-art on the abstractive summarization task. In addition, we contributed a novel abstractive summarization dataset, OrangeSum, that was integrated into the well know GEMv2 benchmark. Moreover, we introduced the notion of *language adaptive pretraining* and applied it to the mBART model, which we adapted to the French language. Finally, we showed that a significant performance boost could be achieved with a cheap further pretraining step.

Currently, language adaptive pretraining is a subject of ongoing work in which we extensively study the pretraining time required to adapt a multilingual model to a monolingual corpus such that it matches the performance of its monolingual counterpart. Some early findings suggest that this research direction is promising and can help alleviate the burden of the high computational requirements of pretraining new language models.

*FrugalScore*

We proposed FrugalScore, a distillation approach to speed up the evaluation of NLG systems when using expensive metrics. Our findings show that training a compact model on a synthetic dataset annotated with an expensive metric can achieve a comparable performance while running several times faster and having several orders of magnitude fewer parameters. In the supervised settings, this step can be seen as a pretraining step that significantly boosts the performance of the model when finetuned on an annotated dataset.

In the future, we would like to explore the possibility of incorporating information from several expensive models into the compact model. Early-stage experiments showed that NLG metrics performance significantly varies w.r.t to the examples on which they are running. While some metrics might fail on some examples, others excel. For example, given $n$ metrics, an oracle metric that considers the best metric among them w.r.t each example can have a significant boost with a wide margin in the performance. Our goal is then to train a model that can predict for each example (e.g., pair of sentences) the best performing metric. This model can be used to annotate our synthetic dataset on which our compact model will be trained.

*DATScore*

We proposed DATScore; an evaluation metric particularly adapted to evaluate machine translation models. Our experiments on the WMT meta-evaluation datasets showed that our model is at least on par with the state-of-the-art models, while it outperforms them in many settings, especially when the target language is different from English.

One of the limitations of the DATScore is the significant runtime. As a future direction, we would like to study the possibility of applying the FrugalScore approach to DATScore. FrugalScore, in its current setup, is designed to distill the metrics that only consider the reference and the generated sentences. However, DATScore takes as input the source sentence in addition to the reference and generated sentences. Adapting FrugalScore to this requirement would alleviate the problem of significant runtime.

## 6.2 EPILOGUE

In recent years, NLP has become a part of our daily life. Whether we are talking to our smart home assistant, using social media, or translating some texts online, large pretrained models are running in the background performing incredible tasks, improving our experience, and making our life easier. However, along with this ease, great challenges and risks are emerging. In this dissertation, we took our first steps on the road to mitigating these challenges hoping that in the future, we could have more significant contributions and higher impact solutions.

# BIBLIOGRAPHY


Miller, George A. (1994). « WordNet: A Lexical Database for English. » In: *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*. URL: https://aclanthology.org/H94-1111 (cit. on p. 65).

Reiter, Ehud and Robert Dale (1997). « Building applied natural language generation systems. » In: *Natural Language Engineering* 3.1, pp. 57–87 (cit. on p. 1).

Joachims, Thorsten (1998). « Text categorization with support vector machines: Learning with many relevant features. » In: *European conference on machine learning*. Springer, pp. 137–142 (cit. on p. 3).

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu (July 2002). « Bleu: a Method for Automatic Evaluation of Machine Translation. » In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, pp. 311–318. DOI: 10.3115/1073083.1073135. URL: https://aclanthology.org/P02-1040 (cit. on pp. 5, 47, 63, 64).

Eyheramendy, Susana, David D. Lewis, and David Madigan (2003). « On the Naive Bayes Model for Text Categorization. » In: *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*. Ed. by Christopher M. Bishop and Brendan J. Frey. Vol. R4. Proceedings of Machine Learning Research. Reissued by PMLR on 01 April 2021. PMLR, pp. 93–100. URL: https://proceedings.mlr.press/r4/eyheramendy03a.html (cit. on p. 3).

Lin, Chin-Yew (July 2004a). « ROUGE: A Package for Automatic Evaluation of Summaries. » In: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, pp. 74–81. URL: https://www.aclweb.org/anthology/W04-1013 (cit. on pp. 5, 19, 41, 63).

– (2004b). « ROUGE: A Package for Automatic Evaluation of Summaries. » In: *Text Summarization Branches Out*. URL: http://aclweb.org/anthology/W04-1013 (cit. on pp. 34, 47).

Banerjee, Satanjeev and Alon Lavie (June 2005). « METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. » In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Ann Arbor, Michigan: Association for Computational Linguistics, pp. 65–72. URL: https://aclanthology.org/W05-0909 (cit. on pp. 5, 47, 65).

Harnly, Aaron, Ani Nenkova, Rebecca Passonneau, and Owen Rambow (2005). « Automation of summary evaluation by the pyramid method. » In: *Recent Advances in Natural Language Processing (RANLP)*, pp. 226–232 (cit. on pp. xvi, 7, 56).

Buciluǎ, Cristian, Rich Caruana, and Alexandru Niculescu-Mizil (2006). « Model compression. » In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 535–541 (cit. on p. 51).

Oliphant, Travis E (2006). *A guide to NumPy*. Vol. 1. Trelgol Publishing USA (cit. on p. 10).

Genkin, Alexander, David D Lewis, and David Madigan (2007). « Large-scale Bayesian logistic regression for text categorization. » In: *technometrics* 49.3, pp. 291–304 (cit. on p. 3).

Hunter, J. D. (2007). « Matplotlib: A 2D graphics environment. » In: *Computing in Science & Engineering* 9.3, pp. 90–95. DOI: 10.1109/MCSE.2007.55 (cit. on p. 10).

Callison-Burch, Chris, Philipp Koehn, Christof Monz, Josh Schroeder, and Cameron Shaw Fordyce, eds. (June 2008). *Proceedings of the Third Workshop on Statistical Machine Translation*. Columbus, Ohio: Association for Computational Linguistics. URL: https://aclanthology.org/W08-0300 (cit. on p. 6).

Collobert, Ronan and Jason Weston (2008). « A unified architecture for natural language processing: Deep neural networks with multitask learning. » In: *Proceedings of the 25th international conference on Machine learning*, pp. 160–167 (cit. on p. 14).

Conroy, John M. and Hoa Trang Dang (Aug. 2008). « Mind the Gap: Dangers of Divorcing Evaluations of Summary Content from Linguistic Quality. » In: *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*. Manchester, UK: Coling 2008 Organizing Committee, pp. 145–152. URL: https://aclanthology.org/C08-1019 (cit. on p. 50).

Dang, Hoa Trang, Karolina Owczarzak, et al. (2008). « Overview of the TAC 2008 update summarization task. » In: *TAC* (cit. on p. 56).

Eisele, Andreas and Yu Chen (2010). « MultiUN: A Multilingual Corpus from United Nation Documents. » In: *LREC* (cit. on p. 29).

Prettenhofer, Peter and Benno Stein (2010). « Cross-language text classification using structural correspondence learning. » In: *Proceedings of the 48th annual meeting of the association for computational linguistics*, pp. 1118–1127 (cit. on p. 38).

McKinney, Wes et al. (2011). « pandas: a foundational Python library for data analysis and statistics. » In: *Python for high performance and scientific computing* 14.9, pp. 1–9 (cit. on p. 10).

Parker, Robert, David Graff, Ke Chen, Junbo Kong, and Kazuaki Maeda (2011). *Arabic Gigaword Fifth Edition*. https://doi.org/10.35111/p02g-rw14. DOI: 10.35111/p02g-rw14 (cit. on p. 39).

Van Der Walt, Stefan, S Chris Colbert, and Gael Varoquaux (2011). « The NumPy array: a structure for efficient numerical computation. » In: *Computing in Science & Engineering* 13.2, p. 22 (cit. on p. 10).

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). « Imagenet classification with deep convolutional neural networks. » In: *Advances in neural information processing systems* 25 (cit. on pp. 23, 66).

Tiedemann, Jörg (2012). « Parallel Data, Tools and Interfaces in OPUS. » In: *Lrec*. Vol. 2012, pp. 2214–2218 (cit. on p. 29).

Hodosh, Micah, Peter Young, and Julia Hockenmaier (2013). « Framing image description as a ranking task: Data, models and evaluation metrics. » In: *Journal of Artificial Intelligence Research* 47, pp. 853–899 (cit. on p. 74).

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean (2013a). « Efficient estimation of word representations in vector space. » In: *arXiv preprint arXiv:1301.3781* (cit. on pp. 14, 48, 65).

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean (2013b). « Distributed representations of words and phrases and their compositionality. » In: *Advances in neural information processing systems*, pp. 3111–3119 (cit. on pp. 4, 6, 14, 25).

Pennington, Jeffrey, Richard Socher, and Christopher D Manning (2014a). « Glove: Global vectors for word representation. » In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543 (cit. on pp. 15, 25).

Pennington, Jeffrey, Richard Socher, and Christopher Manning (Oct. 2014b). « GloVe: Global Vectors for Word Representation. » In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: https://aclanthology.org/D14-1162 (cit. on p. 4).

Skadiņš, Raivis, Jörg Tiedemann, Roberts Rozis, and Daiga Deksne (2014). « Billions of parallel words for free: Building and using the eu bookshop corpus. » In: *Proceedings of LREC* (cit. on p. 29).

Stanojević, Miloš and Khalil Sima'an (2014). « Beer: Better evaluation as ranking. » In: *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pp. 414–419 (cit. on p. 50).

Hermann, Karl Moritz, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom (2015). « Teaching machines to read and comprehend. » In: *Advances in neural information processing systems*, pp. 1693–1701 (cit. on p. 32).

Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean (2015). « Distilling the knowledge in a neural network. » In: *arXiv preprint arXiv:1503.02531* (cit. on p. 51).

Kingma, Diederik P. and Jimmy Ba (2015). « Adam: A Method for Stochastic Optimization. » In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. URL: http://arxiv.org/abs/1412.6980 (cit. on pp. 29, 55).

Kusner, Matt J., Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger (2015). « From Word Embeddings to Document Distances. » In: *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*. ICML'15. Lille, France: JMLR.org, pp. 957–966 (cit. on pp. 6, 20, 49, 65).

Louviere, Jordan J, Terry N Flynn, and Anthony Alfred John Marley (2015). *Best-worst scaling: Theory, methods and applications*. Cambridge University Press (cit. on p. 36).

Ng, Jun-Ping and Viktoria Abrecht (Sept. 2015a). « Better Summarization Evaluation with Word Embeddings for ROUGE. » In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 1925–1930. DOI: 10.18653/v1/D15-1222. URL: https://www.aclweb.org/anthology/D15-1222 (cit. on p. 48).

– (Sept. 2015b). « Better Summarization Evaluation with Word Embeddings for ROUGE. » In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 1925–1930. DOI: 10.18653/v1/D15-1222. URL: https://aclanthology.org/D15-1222 (cit. on p. 63).

Popović, Maja (Sept. 2015). « chrF: character n-gram F-score for automatic MT evaluation. » In: *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal: Association for Computational Linguistics, pp. 392–395. DOI: 10.18653/v1/W15-3049. URL: https://aclanthology.org/W15-3049 (cit. on p. 65).

Sennrich, Rico, Barry Haddow, and Alexandra Birch (2015). « Neural machine translation of rare words with subword units. » In: *arXiv preprint arXiv:1508.07909* (cit. on p. 28).

Vedantam, Ramakrishna, C Lawrence Zitnick, and Devi Parikh (2015). « Cider: Consensus-based image description evaluation. » In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4566–4575 (cit. on p. 74).

Yu, Hui, Qingsong Ma, Xiaofeng Wu, and Qun Liu (Sept. 2015). « CASICT-DCU Participation in WMT2015 Metrics Task. » In: *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal: Association for Computational Linguistics, pp. 417–421. DOI: 10.18653/v1/W15-3053. URL: https://aclanthology.org/W15-3053 (cit. on p. 50).

Lison, Pierre and Jörg Tiedemann (2016). « OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. » In: *Proceedings of the Tenth Interna-*

*tional Conference on Language Resources and Evaluation (LREC'16)*, pp. 923–929 (cit. on p. 53).

Nallapati, Ramesh, Bowen Zhou, Cicero dos Santos, Çağlar GuÌ‡lçehre, and Bing Xiang (Aug. 2016). « Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. » In: *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*. Berlin, Germany: Association for Computational Linguistics, pp. 280–290. DOI: 10.18653/v1/K16-1028. URL: https://www.aclweb.org/anthology/K16-1028 (cit. on p. 53).

Rajpurkar, Pranav, Jian Zhang, Konstantin Lopyrev, and Percy Liang (Nov. 2016). « SQuAD: 100,000+ Questions for Machine Comprehension of Text. » In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, pp. 2383–2392. DOI: 10.18653/v1/D16-1264. URL: https://aclanthology.org/D16-1264 (cit. on p. 2).

Sennrich, Rico, Barry Haddow, and Alexandra Birch (Aug. 2016). « Improving Neural Machine Translation Models with Monolingual Data. » In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 86–96. DOI: 10.18653/v1/P16-1009. URL: https://aclanthology.org/P16-1009 (cit. on p. 66).

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov (2017). « Enriching word vectors with subword information. » In: *Transactions of the Association for Computational Linguistics* 5, pp. 135–146 (cit. on pp. 15, 25).

Bojar, Ondřej, Yvette Graham, and Amir Kamran (Sept. 2017). « Results of the WMT17 Metrics Shared Task. » In: *Proceedings of the Second Conference on Machine Translation*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 489–513. DOI: 10.18653/v1/W17-4755. URL: https://aclanthology.org/W17-4755 (cit. on p. 69).

Chen, Guobin, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker (2017). « Learning efficient object detection models with knowledge distillation. » In: *Advances in neural information processing systems* 30 (cit. on p. 51).

Conneau, Alexis, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes (Sept. 2017). « Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. » In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 670–680. DOI: 10.18653/v1/D17-1070. URL: https://aclanthology.org/D17-1070 (cit. on p. 50).

Gardent, Claire, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini (Sept. 2017). « The WebNLG Challenge: Generating Text from RDF Data. » In: *Proceedings of the 10th International Conference on Natural Language Generation*. Santiago de Compostela, Spain: Association for Computational Linguistics, pp. 124–133.

DOI: 10.18653/v1/W17-3518. URL: https://aclanthology.org/W17-3518 (cit. on p. 52).

Lo, Chi-kiu (Sept. 2017). « MEANT 2.0: Accurate semantic MT evaluation for any output language. » In: *Proceedings of the Second Conference on Machine Translation*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 589–597. DOI: 10.18653/v1/W17-4767. URL: https://aclanthology.org/W17-4767 (cit. on pp. 48, 65).

Ma, Qingsong, Yvette Graham, Shugen Wang, and Qun Liu (2017). « Blend: a novel combined MT metric based on direct assessment—CASICT-DCU submission to WMT17 metrics task. » In: *Proceedings of the second conference on machine translation*, pp. 598–603 (cit. on p. 50).

Novikova, Jekaterina, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser (2017). « Why we need new evaluation metrics for NLG. » In: *arXiv preprint arXiv:1707.06875* (cit. on p. 47).

Peyrard, Maxime, Teresa Botschen, and Iryna Gurevych (Sept. 2017). « Learning to Score System Summaries for Better Content Selection Evaluation. » In: *Proceedings of the Workshop on New Frontiers in Summarization*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 74–84. DOI: 10.18653/v1/W17-4510. URL: https://aclanthology.org/W17-4510 (cit. on p. 50).

See, Abigail, Peter J. Liu, and Christopher D. Manning (July 2017). « Get To The Point: Summarization with Pointer-Generator Networks. » In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 1073–1083. DOI: 10.18653/v1/P17-1099. URL: https://www.aclweb.org/anthology/P17-1099 (cit. on p. 72).

Tättar, Andre and Mark Fishel (2017). « bleu2vec: the painfully familiar metric on continuous vector space steroids. » In: *Proceedings of the Second Conference on Machine Translation*, pp. 619–622 (cit. on p. 48).

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin (2017). « Attention is All you Need. » In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, pp. 6000–6010. URL: http://papers.nips.cc/paper/7181-attention-is-all-you-need (cit. on pp. 1, 11, 23, 25, 27).

Conneau, Alexis, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R Bowman, Holger Schwenk, and Veselin Stoyanov (2018). « XNLI: Evaluating cross-lingual sentence representations. » In: *arXiv preprint arXiv:1809.05053* (cit. on p. 38).

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2018). « Bert: Pre-training of deep bidirectional transformers for language understanding. » In: *arXiv preprint arXiv:1810.04805* (cit. on pp. 4, 6, 23, 26, 39, 49, 54).

Gatt, Albert and Emiel Krahmer (2018). « Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. » In: *Journal of Artificial Intelligence Research* 61, pp. 65–170 (cit. on p. 1).

Howard, Jeremy and Sebastian Ruder (2018). « Universal language model fine-tuning for text classification. » In: *arXiv preprint arXiv:1801.06146* (cit. on p. 25).

Kobayashi, Sosuke (June 2018). « Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations. » In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 452–457. DOI: 10.18653/v1/N18-2072. URL: https://aclanthology.org/N18-2072 (cit. on p. 66).

Kudo, Taku and John Richardson (2018). « Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. » In: *arXiv preprint arXiv:1808.06226* (cit. on p. 28).

Ma, Qingsong, Ondřej Bojar, and Yvette Graham (Oct. 2018). « Results of the WMT18 Metrics Shared Task: Both characters and embeddings achieve good performance. » In: *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*. Belgium, Brussels: Association for Computational Linguistics, pp. 671–688. DOI: 10.18653/v1/W18-6450. URL: https://aclanthology.org/W18-6450 (cit. on p. 69).

Narayan, Shashi, Shay B. Cohen, and Mirella Lapata (2018a). « Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization. » In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, pp. 1797–1807. DOI: 10.18653/v1/D18-1206. URL: https://aclanthology.org/D18-1206 (cit. on pp. vi, xvii, 8, 40, 43, 44).

Narayan, Shashi, Shay B Cohen, and Mirella Lapata (2018b). « Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. » In: *arXiv preprint arXiv:1808.08745* (cit. on pp. xvii, 24, 30, 32, 35, 36).

Peters, Matthew E., Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer (June 2018a). « Deep Contextualized Word Representations. » In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 2227–2237. DOI: 10.18653/v1/N18-1202. URL: https://aclanthology.org/N18-1202 (cit. on pp. 4, 15, 49, 65).

Peters, Matthew E, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer (2018b). « Deep contextualized word representations. » In: *arXiv preprint arXiv:1802.05365* (cit. on p. 25).

Radford, Alec, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever (2018). « Improving language understanding by generative pre-training. » In: *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf* (cit. on pp. 4, 15, 23, 25).

Shimanaka, Hiroki, Tomoyuki Kajiwara, and Mamoru Komachi (Oct. 2018). « RUSE: Regressor Using Sentence Embeddings for Automatic Machine Translation Evaluation. » In: *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*. Belgium, Brussels: Association for Computational Linguistics, pp. 751–758. DOI: 10.18653/v1/W18-6456. URL: https://aclanthology.org/W18-6456 (cit. on p. 50).

Shimorina, Anastasia, Claire Gardent, Shashi Narayan, and Laura Perez-Beltrachini (2018). « WebNLG challenge: Human evaluation results. » PhD thesis. Loria & Inria Grand Est (cit. on p. 73).

Wang, Alex, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman (2018). « Glue: A multi-task benchmark and analysis platform for natural language understanding. » In: *arXiv preprint arXiv:1804.07461* (cit. on pp. 32, 38).

Zhang, Ying, Tao Xiang, Timothy M Hospedales, and Huchuan Lu (2018). « Deep mutual learning. » In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4320–4328 (cit. on p. 51).

Clark, Elizabeth, Asli Celikyilmaz, and Noah A. Smith (July 2019a). « Sentence Mover's Similarity: Automatic Evaluation for Multi-Sentence Texts. » In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 2748–2760. DOI: 10.18653/v1/P19-1264. URL: https://aclanthology.org/P19-1264 (cit. on p. 65).

Clark, Elizabeth, Asli Celikyilmaz, and Noah A Smith (2019b). « Sentence mover's similarity: Automatic evaluation for multi-sentence texts. » In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2748–2760 (cit. on p. 49).

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov (2019). « Unsupervised cross-lingual representation learning at scale. » In: *arXiv preprint arXiv:1911.02116* (cit. on pp. 4, 26).

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (June 2019). « BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. » In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and*

*Short Papers*). Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423 (cit. on pp. 4, 15, 63, 65, 72).

Hu, J. Edward, Abhinav Singh, Nils Holzenberger, Matt Post, and Benjamin Van Durme (Nov. 2019). « Large-Scale, Diverse, Paraphrastic Bitexts via Sampling and Clustering. » In: *Proceedings of the 23rd Conference on Computational Natural Language Learning* (*CoNLL*). Hong Kong, China: Association for Computational Linguistics, pp. 44–54. DOI: 10.18653/v1/K19-1005. URL: https://aclanthology.org/K19-1005 (cit. on p. 72).

Kuratov, Yuri and Mikhail Arkhipov (2019). « Adaptation of deep bidirectional multilingual transformers for russian language. » In: *arXiv preprint arXiv:1905.07213* (cit. on p. 26).

Lample, Guillaume and Alexis Conneau (2019). « Cross-lingual language model pretraining. » In: *arXiv preprint arXiv:1901.07291* (cit. on pp. 4, 26).

Le, Hang, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, and Didier Schwab (2019). « Flaubert: Unsupervised language model pre-training for french. » In: *arXiv preprint arXiv:1912.05372* (cit. on pp. xvii, 24, 26, 28, 29, 32, 36, 38, 39).

Lewis, Mike, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer (2019). *BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension*. arXiv: 1910.13461 [cs.CL] (cit. on pp. 4, 6, 15, 24, 27, 28, 38, 39, 53).

Li, Xian, Paul Michel, Antonios Anastasopoulos, Yonatan Belinkov, Nadir Durrani, Orhan Firat, Philipp Koehn, Graham Neubig, Juan Pino, and Hassan Sajjad (2019). « Findings of the first shared task on machine translation robustness. » In: *arXiv preprint arXiv:1906.11943* (cit. on pp. 29, 56).

Liu, Yang and Mirella Lapata (Nov. 2019). « Text Summarization with Pretrained Encoders. » In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing* (*EMNLP-IJCNLP*). Hong Kong, China: Association for Computational Linguistics, pp. 3730–3740. DOI: 10.18653/v1/D19-1387. URL: https://www.aclweb.org/anthology/D19-1387 (cit. on p. 26).

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov (2019). « Roberta: A robustly optimized bert pretraining approach. » In: *arXiv preprint arXiv:1907.11692* (cit. on pp. 1, 4, 6, 17, 26, 50, 57, 72).

Martin, Louis, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot (2019).

« Camembert: a tasty french language model. » In: *arXiv preprint arXiv:1911.03894* (cit. on pp. 24, 26, 39).

Mathur, Nitika, Timothy Baldwin, and Trevor Cohn (July 2019a). « Putting Evaluation in Context: Contextual Embeddings Improve Machine Translation Evaluation. » In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 2799–2808. DOI: 10.18653/v1/P19-1269. URL: https://aclanthology.org/P19-1269 (cit. on p. 65).

– (2019b). « Putting evaluation in context: Contextual embeddings improve machine translation evaluation. » In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2799–2808 (cit. on p. 49).

Ott, Myle, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli (June 2019a). « fairseq: A Fast, Extensible Toolkit for Sequence Modeling. » In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 48–53. DOI: 10.18653/v1/N19-4009. URL: https://aclanthology.org/N19-4009 (cit. on p. 10).

– (2019b). « fairseq: A fast, extensible toolkit for sequence modeling. » In: *arXiv preprint arXiv:1904.01038* (cit. on pp. 29, 54).

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. (2019). « Pytorch: An imperative style, high-performance deep learning library. » In: *Advances in neural information processing systems* 32 (cit. on p. 10).

Peyrard, Maxime (July 2019). « Studying Summarization Evaluation Metrics in the Appropriate Scoring Range. » In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 5093–5100. DOI: 10.18653/v1/P19-1502. URL: https://aclanthology.org/P19-1502 (cit. on pp. 6, 63).

Phuong, Mary and Christoph H Lampert (2019). « Distillation-based training for multi-exit architectures. » In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1355–1364 (cit. on p. 51).

Polignano, Marco, Pierpaolo Basile, Marco de Gemmis, Giovanni Semeraro, and Valerio Basile (2019). « AlBERTo: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets. » In: *CLiC-it* (cit. on p. 26).

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever (2019). « Language models are unsupervised multitask learners. » In: *OpenAI blog* 1.8, p. 9 (cit. on pp. 26, 50).

Ruder, Sebastian, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf (June 2019). « Transfer Learning in Natural Language Processing. » In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational*

*Linguistics: Tutorials*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 15–18. DOI: 10.18653/v1/N19-5004. URL: https://www.aclweb.org/anthology/N19-5004 (cit. on p. 14).

Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf (2019). « DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. » In: *arXiv preprint arXiv:1910.01108* (cit. on p. 51).

Saputra, Muhamad Risqi U, Pedro PB De Gusmao, Yasin Almalioglu, Andrew Markham, and Niki Trigoni (2019). « Distilling knowledge from a deep pose regressor network. » In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 263–272 (cit. on p. 51).

Shang, Guokan, Antoine Jean-Pierre Tixier, Michalis Vazirgiannis, and Jean-Pierre Lorré (2019). « Energy-based self-attentive learning of abstractive communities for spoken language understanding. » In: *arXiv preprint arXiv:1904.09491* (cit. on p. 3).

Shapira, Ori, David Gabay, Yang Gao, Hadar Ronen, Ramakanth Pasunuru, Mohit Bansal, Yael Amsterdamer, and Ido Dagan (June 2019). « Crowdsourcing Lightweight Pyramids for Manual Summary Evaluation. » In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 682–687. DOI: 10.18653/v1/N19-1072. URL: https://aclanthology.org/N19-1072 (cit. on p. 74).

Shorten, Connor and Taghi M Khoshgoftaar (2019). « A survey on image data augmentation for deep learning. » In: *Journal of big data* 6.1, pp. 1–48 (cit. on p. 66).

Song, Kaitao, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu (2019). « Mass: Masked sequence to sequence pre-training for language generation. » In: *arXiv preprint arXiv:1905.02450* (cit. on p. 27).

Souza, Fábio, Rodrigo Nogueira, and Roberto Lotufo (2019). « Portuguese named entity recognition using BERT-CRF. » In: *arXiv preprint arXiv:1909.10649* (cit. on p. 26).

Strubell, Emma, Ananya Ganesh, and Andrew McCallum (2019). « Energy and policy considerations for deep learning in NLP. » In: *arXiv preprint arXiv:1906.02243* (cit. on pp. 3, 48).

Turc, Iulia, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019). « Well-read students learn better: On the importance of pre-training compact models. » In: *arXiv preprint arXiv:1908.08962* (cit. on p. 54).

Vries, Wietse de, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim (2019). « Bertje: A dutch bert model. » In: *arXiv preprint arXiv:1912.09582* (cit. on p. 26).

Wang, Alex, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman (2019). « Superglue: A stickier bench-

mark for general-purpose language understanding systems. » In: *Advances in Neural Information Processing Systems*, pp. 3266–3280 (cit. on p. 32).

Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. (2019). « Huggingface's transformers: State-of-the-art natural language processing. » In: *arXiv preprint arXiv:1910.03771* (cit. on pp. 10, 55).

Yang, Yinfei, Yuan Zhang, Chris Tar, and Jason Baldridge (2019). « PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. » In: *arXiv preprint arXiv:1908.11828* (cit. on p. 38).

Zhang, Tianyi, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi (2019). « Bertscore: Evaluating text generation with bert. » In: *arXiv preprint arXiv:1904.09675* (cit. on pp. xvii, 19, 34, 48, 49, 54, 63, 65).

Zhao, Wei, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger (Nov. 2019a). « MoverScore: Text Generation Evaluating with Contextualized Embeddings and Earth Mover Distance. » In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, pp. 563–578. DOI: 10.18653/v1/D19-1053. URL: https://aclanthology.org/D19-1053 (cit. on pp. 63, 65).

Zhao, Wei, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger (2019b). « Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. » In: *arXiv preprint arXiv:1909.02622* (cit. on pp. 6, 19, 47–49, 54).

ATILF and CLLE (2020). *Corpus journalistique issu de l'Est Républicain*. ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr. URL: https://hdl.handle.net/11403/est_republicain/v4 (cit. on p. 29).

Al-Maleh, Molham and Said Desouki (2020). « Arabic text summarization using deep learning approach. » In: *Journal of Big Data* 7, pp. 1–17 (cit. on pp. 43, 45).

Antoun, Wissam, Fady Baly, and Hazem Hajj (May 2020). « AraBERT: Transformer-based Model for Arabic Language Understanding. » English. In: *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*. Marseille, France: European Language Resource Association, pp. 9–15. ISBN: 979-10-95546-51-1. URL: https://aclanthology.org/2020.osact-1.2 (cit. on pp. 24, 29, 39, 41).

Bhandari, Manik, Pranav Narayan Gour, Atabak Ashfaq, Pengfei Liu, and Graham Neubig (Nov. 2020). « Re-evaluating Evaluation in Text Summarization. » In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 9347–9359. DOI: 10.18653/v1/2020.emnlp-main.751. URL: https://aclanthology.org/2020.emnlp-main.751 (cit. on pp. 63, 74).

Bird, Jordan J, Diego R Faria, Cristiano Premebida, Anikó Ekárt, and Pedro PS Ayrosa (2020). « Overcoming data scarcity in speaker identification: Dataset augmentation with synthetic mfccs via character-level rnn. » In: *2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*. IEEE, pp. 146–151 (cit. on p. 66).

Brown, Tom B, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. (2020). « Language models are few-shot learners. » In: *arXiv preprint arXiv:2005.14165* (cit. on pp. 1, 48).

Cañete, José, Gabriel Chaperon, Rodrigo Fuentes, and Jorge Pérez (2020). « Spanish Pre-Trained BERT Model and Evaluation Data. » In: *to appear in PML4DC at ICLR 2020* (cit. on p. 26).

Chiang, Cheng-Han, Sung-Feng Huang, and Hung-yi Lee (Nov. 2020). « Pretrained Language Model Embryology: The Birth of ALBERT. » In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 6813–6828. DOI: 10.18653/v1/2020.emnlp-main.553. URL: https://aclanthology.org/2020.emnlp-main.553 (cit. on p. 41).

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov (July 2020). « Unsupervised Cross-lingual Representation Learning at Scale. » In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 8440–8451. DOI: 10.18653/v1/2020.acl-main.747. URL: https://aclanthology.org/2020.acl-main.747 (cit. on p. 65).

Delobelle, Pieter, Thomas Winters, and Bettina Berendt (2020). « RobBERT: a dutch RoBERTa-based language model. » In: *arXiv preprint arXiv:2001.06286* (cit. on p. 26).

Guo, Demi, Yoon Kim, and Alexander Rush (Nov. 2020). « Sequence-Level Mixed Sample Data Augmentation. » In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 5547–5552. DOI: 10.18653/v1/2020.emnlp-main.447. URL: https://aclanthology.org/2020.emnlp-main.447 (cit. on p. 66).

Gururangan, Suchin, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith (2020). « Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. » In: *arXiv preprint arXiv:2004.10964* (cit. on p. 31).

He, Pengcheng, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen (2020). « Deberta: Decoding-enhanced bert with disentangled attention. » In: *arXiv preprint arXiv:2006.03654* (cit. on pp. 48, 57).

Joshi, Pratik, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury (July 2020). « The State and Fate of Linguistic Diversity and Inclusion in the NLP World. » In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 6282–6293. DOI: 10.18653/v1/2020.acl-main.560. URL: https://aclanthology.org/2020.acl-main.560 (cit. on p. 4).

Kane, Hassan, Muhammed Yusuf Kocyigit, Ali Abdalla, Pelkins Ajanoh, and Mohamed Coulibali (Dec. 2020). « NUBIA: NeUral Based Interchangeability Assessor for Text Generation. » In: *Proceedings of the 1st Workshop on Evaluating NLG Evaluation*. Online (Dublin, Ireland): Association for Computational Linguistics, pp. 28–37. URL: https://aclanthology.org/2020.evalnlgeval-1.4 (cit. on p. 50).

Lewis, Mike, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer (July 2020). « BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. » In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 7871–7880. DOI: 10.18653/v1/2020.acl-main.703. URL: https://aclanthology.org/2020.acl-main.703 (cit. on pp. 63–65).

Li, Chuan (2020). « Openai's gpt-3 language model: A technical overview. » In: *Blog Post* (cit. on p. 1).

Liu, Yinhan, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer (2020a). « Multilingual Denoising Pre-training for Neural Machine Translation. » In: *Transactions of the Association for Computational Linguistics* 8, pp. 726–742. DOI: 10.1162/tacl_a_00343. URL: https://aclanthology.org/2020.tacl-1.47 (cit. on pp. 4, 24, 41).

– (2020b). « Multilingual denoising pre-training for neural machine translation. » In: *arXiv preprint arXiv:2001.08210* (cit. on pp. 4, 27, 28, 30, 53).

Maynez, Joshua, Shashi Narayan, Bernd Bohnet, and Ryan McDonald (July 2020). « On Faithfulness and Factuality in Abstractive Summarization. » In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 1906–1919. DOI: 10.18653/v1/2020.acl-main.173. URL: https://aclanthology.org/2020.acl-main.173 (cit. on pp. 44, 45).

Mordido, Gonçalo and Christoph Meinel (Dec. 2020). « Mark-Evaluate: Assessing Language Generation using Population Estimation Methods. » In: *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, pp. 1963–1977. DOI: 10.18653/v1/2020.coling-main.178. URL: https://aclanthology.org/2020.coling-main.178 (cit. on p. 49).

Qiu, Xipeng, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang (2020). « Pre-trained models for natural language processing: A survey. » In: *Science China Technological Sciences* 63.10, pp. 1872–1897 (cit. on p. 63).

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu (2020). « Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. » In: *Journal of Machine Learning Research* 21.140, pp. 1–67. URL: http://jmlr.org/papers/v21/20-074.html (cit. on p. 41).

Rogers, Anna, Olga Kovaleva, and Anna Rumshisky (2020). « A primer in bertology: What we know about how bert works. » In: *arXiv preprint arXiv:2002.12327* (cit. on p. 26).

Rothe, Sascha, Shashi Narayan, and Aliaksei Severyn (2020a). « Leveraging Pre-trained Checkpoints for Sequence Generation Tasks. » In: *Transactions of the Association for Computational Linguistics* 8, pp. 264–280. DOI: 10.1162/tacl_a_00313. URL: https://aclanthology.org/2020.tacl-1.18 (cit. on pp. 1, 40).

– (2020b). « Leveraging pre-trained checkpoints for sequence generation tasks. » In: *Transactions of the Association for Computational Linguistics* 8, pp. 264–280 (cit. on pp. 34, 37).

Safaya, Ali, Moutasem Abdullatif, and Deniz Yuret (Dec. 2020). « KUISAIL at SemEval-2020 Task 12: BERT-CNN for Offensive Speech Identification in Social Media. » In: *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. Barcelona (online): International Committee for Computational Linguistics, pp. 2054–2059. DOI: 10.18653/v1/2020.semeval-1.271. URL: https://aclanthology.org/2020.semeval-1.271 (cit. on p. 39).

Schwartz, Roy, Jesse Dodge, Noah A Smith, and Oren Etzioni (2020). « Green ai. » In: *Communications of the ACM* 63.12, pp. 54–63 (cit. on p. 3).

Sellam, Thibault, Dipanjan Das, and Ankur P Parikh (2020a). « BLEURT: Learning robust metrics for text generation. » In: *arXiv preprint arXiv:2004.04696* (cit. on pp. 51, 59).

Sellam, Thibault, Dipanjan Das, and Ankur Parikh (July 2020b). « BLEURT: Learning Robust Metrics for Text Generation. » In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 7881–7892. DOI: 10.18653/v1/2020.acl-main.704. URL: https://aclanthology.org/2020.acl-main.704 (cit. on p. 66).

Takamoto, Makoto, Yusuke Morishita, and Hitoshi Imaoka (2020). « An efficient method of training small models for regression problems with knowledge distillation. » In: *2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, pp. 67–72 (cit. on p. 51).

Thompson, Neil C, Kristjan Greenewald, Keeheon Lee, and Gabriel F Manso (2020). « The computational limits of deep learning. » In: *arXiv preprint arXiv:2007.05558* (cit. on p. 2).

Tiedemann, Jörg and Santhosh Thottingal (2020). « OPUS-MT — Building open translation services for the World. » In: *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*. Lisbon, Portugal (cit. on p. 53).

Yan, Yu, Weizhen Qi, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou (2020). « Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. » In: *arXiv preprint arXiv:2001.04063* (cit. on p. 27).

Zhang, Tianyi, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi (2020). « BERTScore: Evaluating Text Generation with BERT. » In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=SkeHuCVFDr (cit. on pp. 6, 41, 47).

Abdul-Mageed, Muhammad, AbdelRahim Elmadany, and El Moatez Billah Nagoudi (Aug. 2021). « ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic. » In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, pp. 7088–7105. DOI: 10.18653/v1/2021.acl-long.551. URL: https://aclanthology.org/2021.acl-long.551 (cit. on p. 39).

Chen, Sihao, Fan Zhang, Kazoo Sone, and Dan Roth (June 2021). « Improving Faithfulness in Abstractive Summarization with Contrast Candidate Generation and Selection. » In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, pp. 5935–5941. DOI: 10.18653/v1/2021.naacl-main.475. URL: https://aclanthology.org/2021.naacl-main.475 (cit. on p. 44).

Escolano, Carlos, Marta R. Costa-jussà, José A. R. Fonollosa, and Mikel Artetxe (Apr. 2021). « Multilingual Machine Translation: Closing the Gap between Shared and Language-specific Encoder-Decoders. » In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Online: Association for Computational Linguistics, pp. 944–948. DOI: 10.18653/v1/2021.eacl-main.80. URL: https://aclanthology.org/2021.eacl-main.80 (cit. on p. 72).

Fabbri, Alexander R., Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev (2021). « SummEval: Re-evaluating Summarization Evaluation. » In: *Transactions of the Association for Computational Linguistics* 9, pp. 391–409. DOI: 10.1162/tacl_a_00373. URL: https://aclanthology.org/2021.tacl-1.24 (cit. on p. 74).

Fan, Angela, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. (2021). « Beyond English-Centric Multilingual Machine Translation. » In: *J. Mach. Learn. Res.* 22.107, pp. 1–48 (cit. on pp. vii, 9, 68, 69).

Feng, Steven Y., Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy (Aug. 2021). « A Survey of Data Augmentation Approaches for NLP. » In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics, pp. 968–988. DOI: 10.18653/v1/2021.findings-acl.84. URL: https://aclanthology.org/2021.findings-acl.84 (cit. on p. 66).

Gou, Jianping, Baosheng Yu, Stephen J Maybank, and Dacheng Tao (2021). « Knowledge distillation: A survey. » In: *International Journal of Computer Vision* 129.6, pp. 1789–1819 (cit. on p. 51).

Hasan, Tahmid, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar (Aug. 2021). « XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages. » In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics, pp. 4693–4703. DOI: 10.18653/v1/2021.findings-acl.413. URL: https://aclanthology.org/2021.findings-acl.413 (cit. on pp. 40, 41).

Inoue, Go, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash (Apr. 2021). « The Interplay of Variant, Size, and Task Type in Arabic Pre-trained Language Models. » In: *Proceedings of the Sixth Arabic Natural Language Processing Workshop*. Kyiv, Ukraine (Virtual): Association for Computational Linguistics, pp. 92–104. URL: https://aclanthology.org/2021.wanlp-1.10 (cit. on pp. 39, 41).

Kamal Eddine, Moussa, Guokan Shang, Antoine J-P Tixier, and Michalis Vazirgiannis (2021). « FrugalScore: Learning Cheaper, Lighter and Faster Evaluation Metricsfor Automatic Text Generation. » In: *arXiv preprint arXiv:2110.08559* (cit. on p. 41).

Kamal Eddine, Moussa, Antoine Tixier, and Michalis Vazirgiannis (Nov. 2021). « BARThez: a Skilled Pretrained French Sequence-to-Sequence Model. » In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 9369–9390. URL: https://aclanthology.org/2021.emnlp-main.740 (cit. on pp. 4, 41, 43, 44, 53).

Kang, Myeonginn and Seokho Kang (2021). « Data-free knowledge distillation in neural networks for regression. » In: *Expert Systems with Applications* 175, p. 114813 (cit. on p. 51).

Min, Bonan, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heinz, and Dan Roth (2021). « Recent advances in natural language processing via large pre-trained language models: A survey. » In: *arXiv preprint arXiv:2111.01243* (cit. on p. 63).

Patterson, David, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean (2021). « Carbon emissions and large neural network training. » In: *arXiv preprint arXiv:2104.10350* (cit. on p. 3).

Scialom, Thomas and Felix Hill (2021). « BEAMetrics: A Benchmark for Language Generation Evaluation Evaluation. » In: *arXiv preprint arXiv:2110.09147* (cit. on p. 61).

Yuan, Weizhe, Graham Neubig, and Pengfei Liu (2021). « Bartscore: Evaluating generated text as text generation. » In: *Advances in Neural Information Processing Systems* 34, pp. 27263–27277 (cit. on pp. 20, 63, 65).

Bandel, Elron, Ranit Aharonov, Michal Shmueli-Scheuer, Ilya Shnayderman, Noam Slonim, and Liat Ein-Dor (May 2022). « Quality Controlled Paraphrase Generation. » In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics* (*Volume 1: Long Papers*). Dublin, Ireland: Association for Computational Linguistics, pp. 596–609. DOI: 10.18653/v1/2022.acl-long.45. URL: https://aclanthology.org/2022.acl-long.45 (cit. on p. 79).

Gehrmann, Sebastian, Abhik Bhattacharjee, Abinaya Mahendiran, Alex Wang, Alexandros Papangelis, Aman Madaan, Angelina McMillan-Major, Anna Shvets, Ashish Upadhyay, Bingsheng Yao, et al. (2022). « GEMv2: Multilingual NLG Benchmarking in a Single Line of Code. » In: *arXiv preprint arXiv:2206.11249* (cit. on p. 33).

Kamal Eddine, Moussa, Guokan Shang, Antoine Tixier, and Michalis Vazirgiannis (May 2022a). « FrugalScore: Learning Cheaper, Lighter and Faster Evaluation Metrics for Automatic Text Generation. » In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics* (*Volume 1: Long Papers*). Dublin, Ireland: Association for Computational Linguistics, pp. 1305–1318. DOI: 10.18653/v1/2022.acl-long.93. URL: https://aclanthology.org/2022.acl-long.93 (cit. on p. 66).

Kamal Eddine, Moussa, Nadi Tomeh, Nizar Habash, Joseph Le Roux, and Michalis Vazirgiannis (2022b). « AraBART: a Pretrained Arabic Sequence-to-Sequence Model for Abstractive Summarization. » In: *WANLP 2022* (cit. on p. 4).

Lin, Haitao, Junnan Zhu, Lu Xiang, Yu Zhou, Jiajun Zhang, and Chengqing Zong (May 2022). « Other Roles Matter! Enhancing Role-Oriented Dialogue Summarization via Role Interactions. » In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics* (*Volume 1: Long Papers*). Dublin, Ireland: Association for Computational Linguistics, pp. 2545–2558. DOI: 10.18653/v1/

2022.acl-long.182. URL: https://aclanthology.org/2022.acl-long.182 (cit. on p. 65).

Sai, Ananya B, Akash Kumar Mohankumar, and Mitesh M Khapra (2022). « A survey of evaluation metrics used for NLG systems. » In: *ACM Computing Surveys (CSUR)* 55.2, pp. 1–39 (cit. on pp. 5, 6, 63).

Sevilla, Jaime, Lennart Heim, Anson Ho, Tamay Besiroglu, Marius Hobbhahn, and Pablo Villalobos (2022). « Compute trends across three eras of machine learning. » In: *arXiv preprint arXiv:2202.05924* (cit. on p. 1).

Wan, Yu, Dayiheng Liu, Baosong Yang, Haibo Zhang, Boxing Chen, Derek Wong, and Lidia Chao (May 2022). « UniTE: Unified Translation Evaluation. » In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 8117–8127. DOI: 10.18653/v1/2022.acl-long.558. URL: https://aclanthology.org/2022.acl-long.558 (cit. on p. 65).

Weston, Jack, Raphael Lenain, Udeepa Meepegama, and Emil Fristed (May 2022). « Generative Pretraining for Paraphrase Evaluation. » In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 4052–4073. DOI: 10.18653/v1/2022.acl-long.280. URL: https://aclanthology.org/2022.acl-long.280 (cit. on p. 65).

Antoun, Wissam, Fady Baly, and Hazem Hajj (n.d.). « AraBERT: Transformer-based Model for Arabic Language Understanding. » In: *LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020*, p. 9 (cit. on p. 26).

## COLOPHON

This document was typeset using the typographical look-and-feel `classicthesis`. Most of the graphics in this dissertation are generated using the `Matplotlib` library for the Python programming language. The bibliography is typeset using the `biblatex`.

Part I

# APPENDIX

# A

# APPENDIX APRESENTS SOME EXAMPLES OF THE OUTPUT OF THE VARIOUS

## A.1 ORANGESUM EXAMPLES

In what follows, we provide, for 10 documents randomly selected from OrangeSum's test set, the reference and model summaries for each task (Abstract and Title)

| | | |
|---|---|---|
| | Document | ”Nous pouvons confirmer à ce stade que cinq personnes ont péri. Au moins 70 personnes sont dans un état inconscient dans un hôpital non loin, et dans l’ensemble, entre 200 et 500 personnes reçoivent des soins”, a déclaré Swaroop Rani, une responsable de la police de Visakhapatnam, dans l’État de l’Andhra Pradesh.Opérée par LG Polymers, l’usine est située en bordure de la ville industrielle et portuaire de Visakhapatnam. L’agglomération compte une population d’environ 5 millions de personnes.Le gaz ”avait été laissé là à cause du confinement. Cela a mené à une réaction chimique et de la chaleur est apparue dans les réservoirs, et le gaz a fui à cause de cela”, a expliqué Mme Rani. L’Inde est actuellement confinée depuis fin mars pour lutter contre la pandémie de coronavirus.”Nous avons reçu un appel d’urgence de villageois autour de 03H30 du matin aujourd’hui (mercredi 22H00 GMT, ndlr). Ils disaient qu’il y avait du gaz dans l’air”, a-t-elle poursuivi.”Nous sommes arrivés immédiatement sur place. On pouvait sentir le gaz dans l’air et il ne n’était possible à aucun d’entre nous de rester là plus de quelques minutes”, a-t-elle décrit.L’Inde a été le théâtre en décembre 1984 d’un des pires accidents industriels de l’Histoire, lorsque 40 tonnes de gaz s’étaient échappées d’une usine de pesticides de la ville de Bhopal (centre).Quelque 3.500 personnes avaient péri en quelques jours, principalement dans des bidonvilles situés autour de cette usine d’Union Carbide, et des milliers d’autres dans les années qui ont suivi. |
| Abstract | Gold | Au moins cinq personnes ont péri et plus d’un millier ont été hospitalisées à la suite d’une fuite de gaz dans une usine chimique du sud-est de l’Inde, ont annoncé jeudi la police et une source officielle. |
| | mBART | Cinq personnes sont mortes mercredi dans l’accident d’une usine de pesticides qui s’est produite en Inde, à la suite du confinement lié à l’épidémie de Covid-19, ont indiqué des responsables. |
| | mBARThez | Une explosion a fait cinq morts mercredi dans une usine de pesticides du centre de l’Inde, après que le gaz a fui dans les réservoirs après une réaction chimique, ont indiqué les autorités. |
| | BARThez | Une dizaine de personnes ont péri et des centaines d’autres ont été blessées mercredi dans une usine de pesticides près de Visakhapatnam, dans le sud de l’Inde, a annoncé la police. |
| | C2C | Au moins vingt personnes sont mortes, dont cinq sont mortes et cinq sont portées disparues, selon un bilan officiel lundi après-midi en Inde, faisant craindre une fuite de gaz meurtrière dans le pays, selon une source gouvernementale à l’AFP. |
| Title | Gold | Fuite de gaz dans une usine en Inde: 5 morts, au moins 1.000 personnes hospitalisées |
| | mBART | Inde: cinq morts dans un accident de la usine de pesticides |
| | mBARThez | Inde: au moins cinq morts dans le crash d’une usine de pesticides |
| | BARThez | Inde: cinq morts dans un glissement de terrain à Visakhapatnam |
| | C2C | Inde: cinq morts dans un gaz mortel dans un usine de recyclage |

Table A.1 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 12158.

| | | |
|---|---|---|
| | Document | De nombreux scientifiques occidentaux ont fait part de leurs doutes quant à la rapidité avec laquelle ce vaccin aurait été mis au point.Le ministre américain de la Santé Alex Azar s'est fait l'écho mercredi de leurs points de vue, à l'issue d'une visite de trois jours à Taïwan."Il est important que nous fournissions des vaccins sans danger et efficaces et que les données soient transparentes... Ce n'est pas une course pour être le premier", a-t-il déclaré à la presse lors d'une conférence téléphonique."Je dois souligner que deux des six vaccins américains dans lesquels nous avons investi sont entrés dans la phase des essais cliniques il y a trois semaines, alors que le vaccin russe ne fait que commencer", a-t-il ajouté."Les données des premiers essais en Russie n'ont pas été divulguées, ce n'est pas transparent", a estimé le ministre américain.Mardi, le président russe Vladimir Poutine a annoncé le développement par son pays du "premier" vaccin sans danger contre le Covid-19, affirmant que l'une de ses filles se l'est fait inoculer.Ce vaccin a été baptisé "Spoutnik V" (V comme vaccin, ndlr), en référence à la victoire politico-scientifique russe qu'était la mise en orbite en 1957 du satellite du même nom en pleine Guerre froide.Peu après la déclaration du Kremlin, l'Organisation mondiale de la santé (OMS) a réagi en appelant à la prudence, rappelant que la "pré-qualification" et l'homologation d'un vaccin passaient par des procédures "rigoureuses".De nombreux pays se sont lancés dans une véritable* course contre la montre pour trouver un vaccin efficace contre le coronavirus qui a tué plus de 740.000 personnes à travers la planète depuis son apparition l'an dernier en Chine.Les Etats-Unis sont le pays le touché avec 5,1 millions de cas de Covid-19 qui a fait plus de 164.000 morts. M. Azar s'est dit confiant sur la capacité des Américains à mettre au point un vaccin."Nous pensons qu'il est très crédible que nous ayons des dizaines de millions de doses de vaccin de référence, sûres et efficaces d'ici la fin de cette année, et plusieurs centaines de millions de doses au début de l'année prochaine", a-t-il affirmé.Le président américain a lancé l'opération "Warp Speed" qui vise explicitement à obtenir de quoi vacciner tous les Américains dès janvier 2021. |
| Abstract | Gold | Le ministre américain de la Santé a fait part mercredi de son scepticisme après l'annonce du développement par la Russie du "premier" vaccin contre le coronavirus assurant une "immunité durable". |
| | mBART | Le ministre américain de la Santé s'est exprimé mercredi sur la possibilité d'un vaccin efficace contre le coronavirus. |
| | mBARThez | Le ministre de la Santé américain Alex Azar a déclaré mercredi que les Etats-Unis et la Russie ont fait des essais cliniques d'un vaccin contre le coronavirus "sans danger", alors que le président russe Vladimir Poutine a déclaré mardi avoir inoculé une de ses filles. |
| | BARThez | Les Etats-Unis sont le pays le plus touché par la pandémie de coronavirus après la Russie qui a pourtant annoncé avoir mis au point le "premier vaccin", a estimé le ministre américain de la Santé. |
| | C2C | Le ministre américain de la Santé a souligné que la Russie avait développé des capacités capables de détecter et de tester si le pays n'était pas vaccin contre le nouveau coronavirus, mais a jugé prudent "dans l'attente de la publication d'une réponse scientifique", |
| Title | Gold | Coronavirus: le ministre américain de la Santé sceptique au sujet du vaccin russe |
| | mBART | Vaccin " sans danger": les Américains investis en Russie, selon Alex Azar |
| | mBARThez | Vaccin russe: les Américains appelés à la prudence |
| | BARThez | Un vaccin russe contre le Covid-19 en vue aux Etats-Unis, selon le ministre américain de la Santé |
| | C2C | Coronavirus: les Etats-Unis pas en "cours de combattant" face à un vaccin expérimental |

Table A.2 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 33555.

| | | |
|---|---|---|
| | Document | Une première depuis la Seconde guerre mondiale, la consommation d'alcool ne baisse plus en France. L'Académie nationale de médecine a appelé lundi 29 avril les pouvoirs publics à "prendre des mesures plus fortes" pour lutter contre les problèmes de santé publique causés par la consommation d'alcool. "Pour la première fois depuis la Seconde guerre mondiale, la consommation d'alcool ne baisse plus en France. C'est une défaite majeure pour la santé publique, car l'alcool en est un déterminant fondamental", estime l'Académie dans un communiqué diffusé lundi 29 avril. L'organisme déplore en particulier "l'affaiblissement continu de la loi Evin sous la pression du lobby alcoolier, jusqu'à autoriser la publicité sur l'internet, support médiatique particulièrement affectionné des jeunes". L'alcool serait la première cause évitable* de mortalité des 15-30 ans, selon l'Académie de Médecine. Elle invite donc le gouvernement à revenir aux "principes initiaux" de la loi. Pour un pictogramme plus visible pour les femmes enceintes À l'instar d'autres institutions et associations, l'Académie recommande d'interdire la publicité pour l'alcool et de faire figurer sur les boissons alcoolisées la mention "l'alcool est dangereux pour la santé" (et non le seul excès). L'Académie de médecine veut également voir taxées les boissons au gramme d'alcool et demande la mise en place d'un prix minimum de vente par gramme d'alcool, comme c'est le cas en Ecosse depuis un an. Elle réclame également un pictogramme plus grand et plus lisible sur les bouteilles pour "dissuader de toute consommation la femme enceinte ou qui désire l'être".L'académie de médecine pointe clairement la responsabilité du lobby alcoolier. "Malgré l'enjeu de prévenir la première cause de retard mental évitable* du nouveau-né et de l'enfant, les discussions pour l'agrandir et le contraster s'enlisent depuis des années face à l'opposition farouche du lobby alcoolier". L'alcool serait la première cause de retard mental de l'enfant et de démence précoce souligne l'organisme. Un quart des Français boit trop Dans des chiffres publiés mi-février, Santé publique France avait indiqué que la consommation des Français n'avait quasiment pas reculé depuis 10 ans, passant de 27 g à 26 g d'alcool pur par jour entre 2009 et 2015. "C'est en février 2019 que Santé Publique France annonce que la consommation française d'alcool est la même en 2017 qu'en 2013", note l'académie dans son communiqué. Près d'un quart des Français, soit environ 10,5 millions d'adultes, boivent trop d'alcool, avait également estimé fin mars Santé publique France. L'agence sanitaire a diffusé de nouveaux repères de consommation, résumés par le message "pour votre santé, c'est maximum deux verres par jour, et pas tous les jours".L'alcool constitue la deuxième cause de mortalité évitable* après le tabac, avec 41.000 décès qui lui sont attribuables chaque année en France, 30.000 hommes et 11.000 femmes. L'alcool "est impliqué dans 40% des violences faites aux femmes et aux enfants et un tiers des décès par accidents de la route", ajoute l'Académie dans son communiqué. |
| Abstract | Gold | Elle demande des "mesures plus fortes" pour lutter contre les problèmes de santé causés en France par une consommation d'alcool qui ne diminue plus. |
| | mBART | En février 2019, Santé publique France avait indiqué que la consommation des Français n'avait quasiment pas reculé depuis 10 ans. |
| | mBARThez | Près d'un quart des Français boivent trop d'alcool. |
| | BARThez | L'Académie de médecine réclame notamment "l'affaiblissement continu de la loi Evin sous la pression du lobby alcoolier", jusqu'à autoriser la publicité sur l'internet. |
| | C2C | À l'inverse de ce qui se fait en France, la mesure doit inciter à la consommation d'alcool dès l'âge de 18 ans. |
| Title | Gold | Stagnation de la consommation d'alcool en France : "une défaite majeure pour la santé publique" |
| | mBART | Santé : l'Académie de médecine demande des mesures plus fortes |
| | mBARThez | alcool : l'Académie de médecine appelle le gouvernement à des mesures plus fortes |
| | BARThez | Alcool : une " défaite majeure" pour la santé publique, selon l'Académie de médecine |
| | C2C | La consommation d'alcool en forte hausse : l'Académie de médecine appelle à plus de fermeté |

Table A.3 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 25148.

| | | |
|---|---|---|
| | Document | De petites dimensions (20 cm de largeur et 30 de hauteur), ces ouvertures à hauteur d'homme percées à côté du porche des somptueux palais appartenant aux grandes familles florentines servaient à écouler le vin directement du producteur au consommateur.Au fil des siècles, ce détail architectural et sa fonction sont tombés dans les oubliettes de l'Histoire jusqu'à ce que Massimo Casprini, un érudit florentin, parte à leur redécouverte et y consacre un livre, "I finestrini del vino" ("Les fenêtres à vin"), publié en 2005.Ces fenêtres "ont été créées à partir de 1532 après la chute de la République, quand les Médicis sont revenus au pouvoir et ont voulu favoriser l'agriculture, incitant les grands propriétaires florentins à investir dans les oliveraies et les vignes (...) tout en leur donnant des avantages fiscaux pour revendre directement leur production en ville", explique à l'AFP M. Casprini lors d'une promenade à travers les rues de Florence dans la touffeur estivale.Unique restriction: "Ils pouvaient y vendre seulement le vin de leur propre production et sous un format particulier d'environ 1,4 litre"."L'autre fonction de ces petites fenêtres était sociale, en permettant aux gens du peuple d'acquérir du vin à prix plus raisonnable que chez les commerçants, sans intermédiaire", ajoute-t-il, précisant dans un sourire qu'"à l'époque la consommation de vin était énorme".- Episodes de peste -A l'heure du coronavirus et de la distanciation sociale, Massimo Casprini rappelle que "grâce à ce système on évitait les contacts", alors qu'"épidémies et épisodes de peste étaient très fréquents au XVIe siècle"."En effet, la fenêtre à vin était fermée par un panneau de bois, le client se présentait et frappait avec le heurtoir, à l'intérieur il y avait un caviste qui prenait la bouteille vide et la remplissait. Il n'y avait donc pas de contact direct!" s'extasie le fringuant septuagénaire, également amateur de motos anciennes et auteur de quelque 70 ouvrages centrés sur la capitale toscane.Jusqu'ici, 267 de ces fenêtres à vin ont été répertoriées en Toscane, dont 149 dans le centre de Florence. "Il y en avait beaucoup plus!" estime M. Casprini, "presque tous les propriétaires terriens avaient une fenêtre à vin, mais nombre d'entre elles ont disparu, notamment lors des bombardements de la Seconde Guerre mondiale".Certaines ont aussi été murées, mais grâce à l'œil de lynx de notre expert on réussit encore à reconnaître les contours de leur encadrement en pietra serena (grès gris) ou pierre des carrières de Fiesole, près de Florence.Dans le fil du livre du professeur Casprini a été fondée une association, baptisée "Le buchette del vino", qui recense et appose une plaque sur chaque fenêtre. Son site internet (https://buchettedelvino.org/) propose même une carte interactive permettant de partir à leur découverte, ainsi qu'une galerie de photos et une présentation historique de ces petits trésors architecturaux.On y cite par exemple un guide en français de Florence datant de 1892 qui mentionne la fenêtre d'un palais: "cette cave assez renommée pour ses vins millésimés ne livre aux consommateurs que ceux provenant des propriétés de la marquise Leonia degli Albizi Frescobaldi".Tombées en désuétude, les "finestrini del vino" font aujourd'hui l'objet d'un regain d'intérêt et d'une forme de recyclage: présentoir de magasin, passe-plat dans un café, ou encore petit autel dédié à la Vierge.Même si elles sont protégées par la loi, M. Casprini déplore que "trois fenêtres ont déjà disparu" depuis son premier recensement en 2005. |
| Abstract | Gold | Florence, joyau de la Renaissance, peut s'enorgueillir d'un patrimoine mondialement célèbre, mais dont certains détails restent encore aujourd'hui méconnus: c'est le cas des discrètes "fenêtres à vin" ornant la façade de certains palais, qui permettaient la vente de vin "sans contact", un concept redevenu d'actualité en ces temps de coronavirus. |
| | mBART | "A l'heure du coronavirus et de la distanciation sociale, il n'y avait pas de contact direct !" A Florence, des fenêtres à vin, remplacées par des pierres, auraient été oubliées dans lesoubliettes de l'Histoire. |
| | mBARThez | Massimo Casprini, spécialiste des fenêtres à vins, est revenu mercredi sur la révolution de Florence (Italie) où il a redécouvert l'existence de ces ouvertures en plein air et à ciel ouvert qui permettaient, autrefois, à des propriétaires de vins de revendre leur production à la ville. |
| | BARThez | De 1532 à nos jours, les fenêtres à vin des palais anciens de Florence sont les plus souvent murées, un détail qui a sans doute survécu à l'épidémie de nouveau coronavirus. |
| | C2C | Au lieu de la pandémie de coronavirus, un jardin italien a retrouvé des crus du monde entier: ils étaient des caves à vin français, à quelques dizaines de mètres du sol, pour ne pas être contaminés par le Covid-19. |
| Title | Gold | Virus: comment la Florence des Médicis inventa la vente de vin "sans contact" |
| | mBART | Les fenêtres à vin sont tombées dans les oubliettes de l'Histoire |
| | mBARThez | Florence: les fenêtres à vin cachées dans un livre |
| | BARThez | Florence: des fenêtres à vin traditionnelles à l'heure des peste |
| | C2C | Les fenêtres de la Florence en "huile de vin" : actualité automobile, infos, scoop |

Table A.4 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 34657.

| | | |
|---|---|---|
| | Document | L'ancien chef de l'État était entendu depuis mardi matin, avec une interruption dans la nuit, dans les locaux de l'office anticorruption (OCLCIFF) situés à Nanterre (Hauts-de-Seine). L'ancien président de l'UMP a regagné son domicile parisien du XVIe arrondissement après la fin de sa garde à vue. Également entendu, mais sous le statut de "suspect libre", Brice Hortefeux, un proche de l'ex-président qui occupa plusieurs postes ministériels pendant le quinquennat Sarkozy (2007-2012), a de son côté quitté les locaux de l'office anticorruption mardi soir, assurant sur Twitter avoir apporté des précisions pour "permettre de clore une succession d'erreurs et de mensonges".Depuis la publication, en mai 2012, par le site d'informations Mediapart d'un document libyen - attribué à l'ex-chef des renseignements Moussa Koussa - accréditant un financement d'environ 50 millions d'euros, les investigations des juges ont considérablement avancé. Plusieurs protagonistes du dossier, dont plusieurs ex-responsables libyens, ont accrédité la thèse de versements illicites. Ziad Takieddine persiste et signeLe sulfureux homme d'affaires Ziad Takieddine a lui-même assuré avoir remis entre fin 2006 et début 2007 trois valises contenant 5 millions d'euros en provenance du régime de Kadhafi à Nicolas Sarkozy, alors ministre de l'Intérieur, et à son directeur de cabinet Claude Guéant. Sur BFMTV, il a réédité ses accusations mais répété que cet argent "n'était pas lié à la campagne présidentielle" de 2007."Cet argent faisait partie des accords entre les deux pays sur le contrôle des frontières maritimes, avec échanges d'informations", a précisé l'homme d'affaires, mis en examen autour de ce dossier pour complicité de corruption et complicité de diffamation. "Il y avait un devoir de former en France des équipes libyennes avant la livraison du matériel. Dans ce cadre-là, il y avait des formations à destination de quelques centaines de Libyens. Ils ont établi en France que ça allait coûter dans les cinq millions d'euros", a-t-il ajouté.Nicolas Sarkozy "est un vrai menteur et vous allez voir, il va passer son temps avec les juges d'instruction à dire 'non, non, non c'est pas vrai'. Tout ça pour gagner du temps, c'est sa méthode habituelle", a également lancé Ziad Takieddine sur BFMTV mercredi, assurant "dire la vérité". L'ancien chef de l'État a toujours rejeté ces mises en cause. D'autres dignitaires libyens ont démenti tout financement de la Libye de Mouammar Kadhafi, que Nicolas Sarkozy avait reçu en grande pompe à l'Élysée en 2007.De nouveaux éléments compromettantsOuverte notamment pour "détournements de fonds publics" et "corruption active et passive", l'enquête a été élargie en janvier à des soupçons de "financement illégal de campagne électorale", suite à un rapport de l'office anticorruption qui pointe une circulation importante d'argent liquide dans l'entourage de Nicolas Sarkozy durant la campagne 2007. Selon Le Monde, plusieurs anciens dignitaires du régime Kadhafi auraient livré de récents témoignages confirmant les soupçons de financement illicite. Les investigations ont aussi mis en lumière un virement de 500.000 euros perçu par Claude Guéant en mars 2008, en provenance d'une société d'un avocat malaisien. L'ex-secrétaire général de l'Élysée a toujours affirmé qu'il s'agissait du fruit de la vente de deux table*aux, sans convaincre les juges qui l'ont mis en examen notamment pour "blanchiment de fraude fiscale en bande organisée".Les juges s'interrogent également sur la vente suspecte en 2009 d'une villa située à Mougins, sur la Côte d'Azur, à un fonds libyen. Ils soupçonnent l'homme d'affaires Alexandre Djouhri d'avoir été le véritable* propriétaire de ce bien et de l'avoir cédé pour 10 millions d'euros, soit plus du double du prix du marché, ce qui aurait pu permettre de dissimuler d'éventuels versements occultes. |
| Abstract | Gold | Après une vingtaine d'heures, la garde à vue de Nicolas Sarkozy s'est achevée mercredi soir. L'ancien président a été mis en examen pour "corruption passive", "financement illégal de campagne électorale" et "recel de fonds publics libyens" et placé sous contrôle judiciaire dans le cadre de l'enquête sur des soupçons de financement de sa campagne présidentielle de 2007 par la Libye de Mouammar Kadhafi. |
| | mBART | Ziad Takieddine, mis en examen autour de l'affaire des soupçons de financement libyen de la campagne présidentielle de 2007 de Nicolas Sarkozy, a de nouveau quitté les locaux de son établissement, à Nanterre, dans la nuit de mardi à mercredi. |
| | mBARThez | Cinq jours après la révélation d'un document par Mediapart, l'ancien président de l'UMP et principal suspect dans l'affaire des soupçons de financement libyen de sa campagne présidentielle de 2007 a quitté mardi soir les locaux où il était auditionné. Brice Hortefeux et Claude Guéant ont apporté des précisions. |
| | BARThez | L'ancien président de la République Nicolas Sarkozy a quitté mardi matin les locaux de l'office anticorruption où il était entendu. Les soupçons de financement libyen de sa campagne présidentielle de 2007. |
| | C2C | Nicolas Sarkozy est mis en examen dans le cadre de l'enquête sur les soupçons de financement libyen de sa campagne présidentielle de 2007. Selon plusieurs médias, l'ancien chargé de mission a dit mercredi n'être "pas au courant" de ce que l'ex- |
| Title | Gold | Soupçons de financement libyen : Nicolas Sarkozy mis en examen |
| | mBART | Affaire libyenne : Nicolas Sarkozy en garde à vue |
| | mBARThez | Affaire libyenne : Nicolas Sarkozy entendu par les juges |
| | BARThez | Nicolas Sarkozy en garde à vue, la piste d'un financement libyen s'éloigne |
| | C2C | Nicolas Sarkozy est "un vrai traître" selon l'entourage de Nicolas Sarkozy |

Table A.5 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 22208.

| | | |
|---|---|---|
| | Document | Jean-Paul Dufrègne a passé un sale quart d'heure sur les réseaux sociaux mercredi soir. Cet élu communiste de l'Allier a été filmé par les caméras de TF1, dans un reportage diffusé le 4 avril au journal de 20 heures. Mais téléspectateurs et internautes n'ont nullement prêté attention aux arguments du député sur les inquiétudes persistantes des territoires ruraux et la réforme institutionnelle sur laquelle planche le gouvernement. Non, ils étaient bien trop captivés par son compteur de vitesse, filmé le temps de quelques plans par les caméras de la première chaîne, comme le relève LCI.Car, sur une route départementale limitée à 90 km/heure, Jean-Paul Dufrègne avait le pied au plancher. Son compteur affichait 124 km/heure, plus de 30 km/heure au-dessus de la limite autorisée. Une infraction que n'ont pas manqué de relever de nombreux internautes. " Trois points et 135 euros d'amende ", note un utilisateur de Twitter. " Bonjour, les limitations de vitesse ne s'appliquent pas aux parlementaires ? ", ironise un autre.Opposant au 80km/heureCertains ont par ailleurs fait le lien entre les positions politiques de l'élu communiste et cet excès de vitesse. Car Jean-Paul Dufrègne est un farouche opposant au projet du gouvernement de limiter le réseau français de routes secondaires à 80 km/heure. Avec une trentaine d'autres élus du Massif Central, il avait d'ailleurs adressé une lettre ouverte à Emmanuel Macron sur le sujet, dénonçant une mesure " injuste et pénalisante ", et un frein au développement du Massif Central. |
| Abstract | Gold | Jean-Paul Dufrègne, élu de l'Allier, a fait sensation sur les réseaux sociaux après être apparu dans un reportage de TF1. On le voyait rouler à plus de 120 km/heure sur une route secondaire. |
| | mBART | L'élu communiste de l'Allier a été filmé le 4 avril au journal de 20 heures de TF1, comme le relève LCI. Son compteur affichait 124 km/heure au-dessus de la limite autorisée. |
| | mBARThez | Son compteur affichait 124 km/heure au-dessus de la limite autorisée. |
| | BARThez | Un élu communiste de l'Allier a été filmé le temps d'un reportage par les caméras de TF1, dans la soirée du 4 avril, alors qu'il roulait 90 km/heure au-dessus de la limite autorisée. |
| | C2C | Un député communiste du Puy-de-Dôme n'a pas manqué de se faire remarquer en pleine séance de questions au gouvernement. Il était l'invité de Jean-Marc Remontini, l'élu apparenté du PCF, qui a fait l'éloge de |
| Title | Gold | Un élu épinglé à 124 km/heure sur une route limitée à 90 |
| | mBART | Un élu communiste flashé à 124 km/heure sur TF1 |
| | mBARThez | Un élu communiste flashé à 124 km/heure sur TF1 |
| | BARThez | Un élu communiste passe un sale quart d'heure à cause de son compteur de vitesse |
| | C2C | Ce député du Doubs qui n'a plus le temps de répondre aux radars |

Table A.6 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 22077.

| | | |
|---|---|---|
| | Document | Mais où est donc passé Gérald Thomassin ? L'acteur français qui avait obtenu un César en 1991 est introuvable depuis le 28 août dernier, rapporte RTL. Le comédien âgé de 45 ans devait se rendre à un rendez-vous judiciaire dans une affaire de meurtre. Mais il ne s'y est jamais rendu. Et depuis, c'est toute sa famille qui s'inquiète. Interrogé par RTL, le frère de l'acteur, Jérôme Thomassin, a montré toute son inquiétude avant d'apporter des détails sur la journée du 28 août.Selon lui, Gérald Thomassin a bien "pris le train Rochefort-Lyon pour se rendre à la confrontation avec deux autres mis en examen". Parmi ces hommes, précise RTL, le principal suspect dans cette affaire de meurtre dans un bureau de poste. Les avocats du comédien qui appartiennent au cabinet d'Éric Dupond-Moretti ont signalé "une disparition inquiétante" au commissariat de Rochefort (Charente-Maritime) où l'acteur vivait. En tout état de cause, son frère était "très heureux de pouvoir se rendre à ce rendez-vous judiciaire." "L'affaire Burgod" L'affaire remonte à 2013, lorsque Gérald Thomassin est interpellé et mis en examen pour "vol avec arme et homicide sur une personne chargée d'une mission de service public". Une employée de La Poste, Catherine Burgod, enceinte, avait été tuée de 28 coups de couteau. Tenu responsable, l'acteur avait été incarcéré en 2013 avant d'être remis en liberté, mais placé sous contrôle judiciaire en octobre 2015. Sauf qu'il décide de briser son bracelet électronique et retourne en prison. Gérald Thomassin sort finalement en 2016, après trois ans de détention provisoire, la limite. L'affaire prend une autre tournure en 2017 et 2018 avec l'arrestation d'un suspect et la mise en examen d'un autre, mais la justice ne parvient toujours pas à trancher. La reconstitution du jeudi 29 août aurait dû permettre une confrontation entre les trois protagonistes, mais Gérald Thomassin ne s'est jamais présenté, au grand dam de l'avocate des parties civiles. Aujourd'hui, la famille de l'acteur demande de vraies recherches. Ses appels sonnent dans le vide. Ses proches craignent qu'il ait pu faire une mauvaise rencontre en cours de route. Le comédien avait reçu le César du Meilleur jeune espoir en 1991 pour son rôle dans "Le Petit Criminel" de Jacques Doillon, avant de connaitre une carrière ponctuée de plus ou moins grands rôles. |
| Abstract | Gold | Gérald Thomassin a disparu fin août selon ses proches qui s'inquiètent, révèle RTL. Il devait se rendre à un rendez-vous judiciaire pour éclairer une affaire d'homicide. |
| | mBART | Gérald Thomassin a disparu fin août. Il devait se rendre à un rendez-vous judiciaire pour éclairer une affaire d'homicide, révèle RTL. |
| | mBARThez | Gérald Thomassin a disparu fin août. Il devait se rendre, selon RTL, à un rendez-vous judiciaire pour éclairer une affaire de meurtre. |
| | BARThez | Le comédien de 45 ans devait se rendre à un rendez-vous judiciaire dans une affaire de meurtre, mais il ne s'y est jamais rendu, rapporte RTL. |
| | C2C | Gérald Thomassin a disparu fin août. Il se rendait à un rendez-vous judiciaire pour éclairer une affaire d'homicide. |
| Title | Gold | Mystérieuse disparition d'un acteur césarisé soupçonné de meurtre |
| | mBART | Disparition de l'acteur Gérald Thomassin : sa famille s'inquiète |
| | mBARThez | L'acteur Gérald Thomassin porté disparu depuis le 28 août |
| | BARThez | L'acteur Gérald Thomassin porté disparu depuis le 28 août |
| | C2C | Disparition de l'acteur Gérald Thomassin : la famille n'est plus introuvable |

Table A.7 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 22168.

| | | |
|---|---|---|
| | Document | Dans un rapport adressé aux ministres de l'Intérieur, de la Justice, et à la secrétaire d'Etat à l'Egalité femmes-hommes Marlène Schiappa, les cinq députés chargés d'étudier la verbalisation du harcèlement de rue recommandent la mise en place d'"une contravention de 4e classe d'outrage sexiste et sexuel". L'infraction devra être constatée "en flagrance" par les agents de la toute récente "police de proximité du quotidien", précise leur texte, qui, selon les informations du Huffington Post, devrait être remis mercredi 28 février.Jusqu'à 1.500 euros d'amendesLe montant de l'amende forfaitaire serait de 90 euros pour un paiement immédiat, 200 euros pour un paiement sous 15 jours et 350 euros en peine majorée. En cas de circonstances aggravantes (si l'auteur est dépositaire de l'autorité publique, en cas de réunion, ou de bande organisée), une contravention de 5e classe (jusqu'à 1.500 euros) pourrait être délivrée par un tribunal de police.Pour Sophie Auconie (UDI, Agir et Indépendants), Laetitia Avia (LREM), Erwan Balanant (Modem), Elise Fajgeles (LREM) et Marietta Karamanli (Nouvelle gauche), le harcèlement subi dans l'espace public est un "fléau". Ils estiment nécessaire de "définir une nouvelle infraction visant à sanctionner, entre autres, les gestes déplacés, les sifflements, les regards insistants ou remarques obscènes, le fait de suivre volontairement à distance une personne créant ainsi une situation d'angoisse", soulignent-ils.68% des Français favorables aux amendesLe rapport souhaite également que les auteurs participent à un stage de sensibilisation à l'égalité femmes-hommes, et que la police municipale et les agents des services de sécurité des transports soient habilités à constater cette infraction. D'après un sondage Opinionway réalisé pour Public Sénat, Les Echos et Radio Classique et publié le 5 février, une large majorité de Français est favorable à la mise en place d'une amende pénalisant le harcèlement de rue. À la question "êtes-vous favorable ou pas favorable à ce que le harcèlement de rue (sifflements, remarques...) soit passible d'une amende ?", 68 % des personnes interrogées se disent favorables (40 % "plutôt favorables" et 28 % "tout à fait favorables"). 30 % y sont opposés (23 % "plutôt opposés" et 7 % "tout à fait opposés") et 2 % ne se prononcent pas. |
| Abstract | Gold | Des parlementaires préconisent de créer une infraction d'"outrage sexiste" sanctionnant d'une amende immédiate de 90 euros "tout propos, comportement ou pression à caractère sexiste ou sexuel" dans l'espace public. |
| | mBART | Selon un rapport, dévoilé par le Huffington Post, le gouvernement envisage une amende forfaitaire de 90 euros pour lutter contre le harcèlement de rue. En cas de circonstances aggravantes, elle pourrait être délivrée par un tribunal de police. |
| | mBARThez | Dans un rapport adressé aux ministres de l'Intérieur, de la Justice et à la secrétaire d'Etat à l'Egalité femmes-hommes, les députés chargés d'étudier la verbalisation du harcèlement de rue recommandent la mise en place d'une contravention de 4e classe. |
| | BARThez | D'après un sondage Opinionway réalisé pour Public Sénat, Les Echos et Radio Classique, une large majorité de Français sont favorables à la mise en place d'une amende pénalisant le harcèlement de rue. |
| | C2C | Selon un sondage Elabe pour Le Huffington Post, 54% des Français sont opposés au projet de loi sur le harcèlement de rue. Une première en soi, alors que la question de l'emprise sexuelle se pose déjà : les contraventions seront en effet posées |
| Title | Gold | Harcèlement de rue : bientôt une amende immédiate de 90 euros ? |
| | mBART | Harcèlement de rue : vers une contravention de 4e classe ? |
| | mBARThez | Harcèlement de rue : vers une contravention de 4e classe ? |
| | BARThez | Harcèlement de rue : vers une contravention de 4e classe ? |
| | C2C | Harcèlement de rue : un rapport préconise une amende de 5 à 5 euros |

Table A.8 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 22423.

| | | |
|---|---|---|
| | Document | Le 18 octobre dernier, Jacline Mouraud se faisait connaître en publiant sur Facebook une vidéo dans laquelle elle poussait un ”coup de gueule” contre le gouvernement. Aujourd'hui, la Bretonne a pris ses distances par rapport au mouvement, notamment face à d'autres figures plus radicales comme Éric Drouet.Jacline Mouraud réfléchit désormais à créer son propre parti, ”la seule chose envisageable”, comme elle l'explique au JDD.Nicolas Sarkozy, ”le seul qui a des couilles”Cette figure des ”gilets jaunes”, accusée de faire le jeu de LREM estime que ”le problème” d'Emmanuel Macron ”c'est qu'il est jeune”. ”Il devrait y avoir un âge minimum pour être président : 50 ans”, souligne Jacline Mouraud.Dans le JDD, elle raconte d'ailleurs avoir voté blanc lors de la dernière présidentielle. En 2007 et 2012, c'est Nicolas Sarkozy, ”le seul qui a des couilles”, que la figure des ”gilets jaunes” avait soutenu. En attendant de se lancer, pas question pour elle en tous les cas d'être candidate aux européennes sur une liste de La République en marche. |
| Abstract | Gold | L'une des figures du mouvement ne sera toutefois pas candidate aux prochaines élections européennes. |
| | mBART | Jacline Mouraud, figure des ”gilets jaunes”, estime que le président d'Emmanuel Macron est trop jeune pour être président. |
| | mBARThez | Dans un entretien au JDD, la figure des ”gilets jaunes” Jacline Mouraud révèle qu'elle réfléchit à créer son propre parti. |
| | BARThez | Dans les colonnes du JDD, la figure des ”gilets jaunes” explique qu'elle envisage de se présenter aux européennes sur une liste La République en marche. |
| | C2C | Retirée de la vie politique depuis plusieurs mois, Bretone Mouraud envisage de se lancer en politique. Et elle réfléchit à quelque chose de plus, rapporte le JDD. |
| Title | Gold | ”Gilets jaunes” : Jacline Mouraud réfléchit à créer son parti |
| | mBART | ”Gilets jaunes” : Jacline Mouraud lance son propre parti |
| | mBARThez | ”Gilets jaunes” : Jacline Mouraud prend ses distances |
| | BARThez | La figure des ”gilets jaunes” Jacline Mouraud va créer son propre parti |
| | C2C | ”Gilets jaunes” : Jacline Mouraud réfléchit à sa propre candidature |

Table A.9 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 19233.

| | | |
|---|---|---|
| | Document | Invité du "Grand rendez-vous Europe 1/CNews/Les Échos dimanche 8 avril, Jean-Luc Mélenchon a appelé "à faire baisser la température dans ce pays". En cause : les menaces de mort dont il ferait l'objet, ainsi que d'autres élus LFI. Le député des Bouches-du-Rhône a confirmé avoir récemment demandé que le ministre de l'Intérieur Gérard Collomb soit entendu dans l'enquête sur un projet d'attentat d'ultra-droite où il a été cité comme cible potentielle. "Je me suis porté partie civile dans cette affaire. J'ai appris en octobre dernier qu'un groupe de gens avait l'intention de me tuer, ainsi que (le secrétaire d'État) M. Castaner". Or pendant la campagne législative de juin 2017, "j'ai demandé à être protégé" car "j'avais reçu à Marseille des menaces de mort. On me l'a refusé, et puis après je découvre que le 28 mai, ils ont arrêté ce personnage (...) Quatre mois plus tard ils en arrêtent neuf autres qui étaient toujours en action pendant ces quatre mois". "LA RECRUDESCENCE D'UN EXTRÉMISME D'EXTRÊME DROITE EXTRÊMEMENT VIOLENT"Un ancien militant du groupuscule royaliste Action française en Provence, Alexandre Nisin, a été mis en examen début juillet pour association de malfaiteurs terroriste criminelle. Huit autres suspects ont été mis en examen, soupçonnés d'appartenir à son réseau. "Ni moi, ni Castaner n'avons été prévenus de rien", a déploré l'ancien candidat à la présidentielle. "Sur 17 que nous sommes au groupe La France insoumise (à l'Assemblée, ndlr), il y en a cinq qui font l'objet de menaces de mort"(/BOLD], a-t-il par ailleurs révélé. Jean-Luc Mélenchon a dénoncé "la recrudescence d'un extrémisme d'extrême droite extrêmement violent, dans toutes sortes de villes, qui va jusqu'à des tentatives d'assassinat". "L'extrême droite doit être prise au sérieux comme danger de violence et de meurtre. C'est eux qui attaquent à Montpellier un amphi d'étudiants, c'est eux qui attaquent à Tolbiac, c'est eux qui me menacent de mort. C'est eux qui font des contrôles d'identité dans la rue dans au moins deux villes. Ça suffit. Maintenant le ministre de l'Intérieur doit prendre au sérieux la menace que représentent les groupuscules radicalisés de l'extrême droite", a-t-il poursuivi."Il y a des groupes d'extrême droite qui prolifèrent dans le pays. Qui souvent ont commencé leurs premiers pas avec le Front national et qui maintenant vont au bout de cette logique", a-t-il ajouté. |
| Abstract | Gold | Le leader de La France insoumise (LFI) dénonce la "recrudescence" d'une "extrême droite extrêmement violent(e)" en France, qui doit être "prise au sérieux" par le gouvernement. |
| | mBART | S'il dénonce la recrudescence d'un extrémisme d'extrême droite "extrêmement violent" dans certaines villes, le chef de file de La France insoumise (LFI), Jean-Luc Mélenchon, s'est attaqué au ministre de l'Intérieur, Gérard Collomb. |
| | mBARThez | Au micro d'Europe 1 dimanche 8 avril, Jean-Luc Mélenchon a réagi aux menaces de mort dont il fait l'objet et dénoncé "la recrudescence d'un extrémisme d'extrême droite extrêmement violent". |
| | BARThez | - Le chef de file de La France insoumise et ancien candidat à la présidentielle est vent debout contre le projet d'attentat déjoué à Marseille. Il estime que le ministre de l'Intérieur, Gérard Collomb, est menacé de mort par un groupe d'extrême droite. |
| | C2C | Selon le leader de La France insoumise (LFI), le député des Bouches-du-Rhône, Jean-Luc Mélenchon, "rappelle à tous ceux suspectés d'avoir menacé d'assassiner le ministre de l'Intérieur, ce que conteste le parti et |
| Title | Gold | VIDÉO. Cinq députés de La France insoumise font l'objet de menaces de mort, selon Jean-Luc Mélenchon |
| | mBART | Menaces de mort : Jean-Luc Mélenchon s'en prend à Castaner |
| | mBARThez | Jean-Luc Mélenchon dénonce les "menaces de mort" de Gérard Collomb |
| | BARThez | VIDÉO. Jean-Luc Mélenchon dénonce les menaces de mort dont Gérard Collomb est victime |
| | C2C | Projet VIDÉO. Menace de mort à Marseille : Jean-Luc Mélenchon menace de démissionner |

Table A.10 – C2C stands for CamemBERT2CamemBERT. OrangeSum document 22060.

Table B.1 – Summary-level Pearson correlation (pyramid score/responsiveness).

| | Metric | Model | TAC-2008 | TAC-2009 | TAC-2010 | TAC-2011 | Macro Avg. Score | Runtime | Params |
|---|---|---|---|---|---|---|---|---|---|
| a | BERTScore | BERT-Tiny | 52.1/44.4 | 62.2/51.9 | 54.6/49.9 | 52.7/43.6 | 55.4/47.5 | 1m 27s | 4.4M |
| b | BERTScore | BERT-Small | 56.0/47.8 | 70.0/54.6 | 61.1/54.5 | 59.1/49.2 | 61.6/51.5 | 2m 20s | 29.1M |
| c | BERTScore | BERT-Medium | 57.3/48.5 | 70.6/55.3 | 63.1/56.2 | 59.7/49.5 | 62.7/52.4 | 2m 28s | 41.7M |
| d | BERTScore | BERT-Base | 61.3/52.2 | 73.2/58.7 | 63.3/56.8 | 61.0/51.2 | 64.7/54.7 | 3m 28s | 110M |
| e | BERTScore | RoBERTa-Large | 56.4/50.9 | 71.1/58.3 | **69.1**/**61.4** | 60.3/50.8 | 64.2/55.4 | 5m 17s | 355M |
| f | BERTScore | DeBERTa-XLarge | 60.9/**54.5** | **73.9**/**60.4** | 62.6/56.0 | 61.5/**53.0** | 64.5/**56.0** | 6m 20s | 900M |
| g | MoverScore | BERT-Base | **64.7**/54.2 | **73.9**/58.2 | 64.7/57.0 | **62.6**/52.5 | **66.5**/55.4 | 301m 29s | 110M |
| i | FrugalScore$_d$ | BERT-Tiny | 60.9/50.0 | 72.5/56.4 | 64.8/57.5 | 61.4/50.0 | 64.9/53.5 | 1m 28s | 4.4M |
| ii | FrugalScore$_d$ | BERT-Small | 61.9/51.8 | 73.0/57.3 | 62.6/55.8 | 61.3/50.0 | 64.7/53.7 | 1m 35s | 29.1M |
| iii | FrugalScore$_d$ | BERT-Medium | 62.0/52.2 | 73.3/**58.1** | 62.6/56.0 | 61.3/50.6 | 64.8/54.2 | 1m 55s | 41.7M |
| iv | FrugalScore$_e$ | BERT-Tiny | 54.8/46.4 | 66.8/54.2 | 61.8/53.1 | 56.4/46.7 | 60.0/50.1 | 1m 28s | 4.4M |
| v | FrugalScore$_e$ | BERT-Small | 59.1/49.6 | 72.7/55.7 | **68.1**/59.8 | 63.0/50.1 | 64.1/53.8 | 2m 29s | 29.1M |
| vi | FrugalScore$_e$ | BERT-Medium | 57.9/48.4 | 71.8/54.4 | 65.7/57.0 | 60.3/48.5 | 63.9/52.1 | 3m 41s | 41.7M |
| vii | FrugalScore$_f$ | BERT-Tiny | 57.8/48.5 | 68.6/55.7 | 63.0/54.8 | 57.5/47.8 | 61.7/51.0 | 1m 28s | 4.4M |
| viii | FrugalScore$_f$ | BERT-Small | 60.1/51.0 | 73.5/57.5 | 67.3/59.5 | 63.1/51.7 | 66.0/54.9 | 2m 29s | 29.1M |
| ix | FrugalScore$_f$ | BERT-Medium | 59.0/50.3 | 73.3/57.4 | 67.2/**60.2** | 62.4/51.5 | 65.5/54.9 | 3m 41s | 41.7M |
| x | FrugalScore$_g$ | BERT-Tiny | 63.6/51.7 | **74.4**/57.3 | 68.0/60.1 | **63.2**/51.2 | **67.3**/**55.1** | 1m 28s | 4.4M |
| xi | FrugalScore$_g$ | BERT-Small | 63.2/52.5 | 73.1/57.1 | 65.1/57.6 | 62.3/51.5 | 65.9/54.7 | 2m 29s | 29.1M |
| xii | FrugalScore$_g$ | BERT-Medium | **63.8**/**53.2** | 73.6/57.7 | 65.3/57.5 | 62.1/**51.8** | 66.2/**55.1** | 3m 41s | 41.7M |

Table B.2 – Segment-level Pearson correlation.

| | Metric | Model | de-en | fi-en | gu-en | kk-en | lt-en | ru-en | zh-en | Macro Avg. Score | Runtime | Params |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| a | BERTScore | BERT-Tiny | 29.7 | 32.5 | 33.9 | 52.0 | 40.5 | 30.7 | 44.2 | 37.6 | 1m 22s | 4.4M |
| b | BERTScore | BERT-Small | 30.0 | 33.6 | 34.6 | 52.4 | 42.3 | 31.8 | 49.1 | 39.1 | 1m 42s | 29.1M |
| c | BERTScore | BERT-Medium | 30.8 | 34.4 | 35.2 | 52.8 | 42.8 | 32.4 | 50.3 | 39.8 | 2m 04s | 41.7M |
| d | BERTScore | BERT-Base | 32.8 | 37.4 | 37.1 | 54.0 | 44.7 | 33.7 | 53.7 | 41.9 | 2m 09s | 110M |
| e | BERTScore | RoBERTa-Large | 35.3 | 38.7 | 38.7 | 52.0 | 45.3 | 34.3 | **58.3** | 43.2 | 3m 03s | 355M |
| f | BERTScore | DeBERTa-XLarge | **37.6** | **39.2** | **40.3** | 53.4 | **47.3** | **35.7** | 57.8 | **44.5** | 3m 49s | 900M |
| g | MoverScore | BERT-Base | 36.5 | 39.1 | 39.3 | **55.0** | 46.5 | 35.6 | 56.0 | 44.0 | 64m 32s | 110M |
| i | FrugalScore$_d$ | BERT-Tiny | 30.2 | 32.8 | 34.6 | 52.4 | 39.9 | 31.2 | 47.7 | 38.4 | 1m 18s | 4.4M |
| ii | FrugalScore$_d$ | BERT-Small | 32.6 | 35.9 | 37.1 | 54.1 | 43.5 | 33.6 | 52.3 | 41.3 | 1m 35s | 29.1M |
| iii | FrugalScore$_d$ | BERT-Medium | 32.9 | 37.0 | 37.4 | 54.4 | 44.3 | 34.1 | 53.2 | 41.9 | 1m 55s | 41.7M |
| iv | FrugalScore$_e$ | BERT-Tiny | 30.6 | 32.8 | 33.0 | 49.8 | 38.7 | 29.8 | 48.1 | 37.5 | 1m 18s | 4.4M |
| v | FrugalScore$_e$ | BERT-Small | 33.7 | 35.4 | 35.4 | 51.6 | 42.6 | 32.6 | 52.5 | 40.5 | 1m 35s | 29.1M |
| vi | FrugalScore$_e$ | BERT-Medium | 35.2 | 37.1 | 35.6 | 52.0 | 44.0 | 33.8 | 54.4 | 41.7 | 1m 55s | 41.7M |
| vii | FrugalScore$_f$ | BERT-Tiny | 30.8 | 33.1 | 34.4 | 50.8 | 39.4 | 30.4 | 47.1 | 38.0 | 1m 18s | 4.4M |
| viii | FrugalScore$_f$ | BERT-Small | 34.5 | 36.4 | 37.0 | 52.7 | 43.9 | 33.4 | 52.6 | 41.5 | 1m 35s | 29.1M |
| ix | FrugalScore$_f$ | BERT-Medium | 35.8 | **38.3** | 37.7 | 53.4 | 45.7 | 34.8 | **55.1** | 43.0 | 1m 55s | 41.7M |
| x | FrugalScore$_g$ | BERT-Tiny | 33.0 | 34.0 | 36.2 | 53.6 | 40.5 | 32.7 | 48.6 | 39.8 | 1m 18s | 4.4M |
| xi | FrugalScore$_g$ | BERT-Small | 35.6 | 37.4 | 38.9 | 55.0 | 44.8 | 34.8 | 52.8 | 42.8 | 1m 35s | 29.1M |
| xii | FrugalScore$_g$ | BERT-Medium | **36.2** | **38.3** | **39.1** | **55.6** | **45.8** | **35.3** | 54.7 | **43.6** | 1m 55s | 41.7M |

Table B.3 – System-level Pearson correlation (pyramid/responsiveness).

| | Metric | Model | TAC-2008 | TAC-2009 | TAC-2010 | TAC-2011 | Macro Avg. Score | Runtime | Params |
|---|---|---|---|---|---|---|---|---|---|
| a | BERTScore | BERT-Tiny | 82.5/77.6 | 87.4/81.8 | 77.5/75.0 | 82.1/79.2 | 82.4/78.4 | 1m 27s | 4.4M |
| b | BERTScore | BERT-Small | 84.4/81.4 | 95.8/84.0 | 81.3/78.0 | 87.6/85.3 | 87.3/82.2 | 2m 20s | 29.1M |
| c | BERTScore | BERT-Medium | 86.3/82.7 | 96.0/84.6 | 84.0/80.6 | 87.8/85.5 | 88.5/83.3 | 2m 28s | 41.7M |
| d | BERTScore | BERT-Base | 90.6/87.5 | 96.5/87.5 | 83.7/80.9 | 88.3/86.4 | 89.8/85.6 | 3m 28s | 110M |
| e | BERTScore | RoBERTa-Large | 80.0/80.9 | 94.7/87.7 | **92.7**/**89.8** | 88.9/89.2 | 89.1/86.9 | 5m 17s | 355M |
| f | BERTScore | DeBERTa-XLarge | 88.0/**89.8** | **97.5**/**89.8** | 85.7/84.0 | **90.7**/**91.8** | 90.5/**88.9** | 6m 20s | 900M |
| g | MoverScore | BERT-Base | **95.4**/89.5 | 96.9/85.9 | 85.7/84.0 | 88.6/86.0 | **91.7**/86.3 | 301m 29s | 110M |
| i | FrugalScore$_d$ | BERT-Tiny | 91.6/85.3 | 95.8/84.7 | 86.2/82.9 | 88.3/84.4 | 90.5/84.3 | 1m 28s | 4.4M |
| ii | FrugalScore$_d$ | BERT-Small | 90.9/86.8 | 96.2/85.4 | 82.8/79.6 | 87.8/84.3 | 89.4/84.0 | 1m 35s | 29.1M |
| iii | FrugalScore$_d$ | BERT-Medium | 90.6/87.0 | 96.6/86.3 | 82.5/79.6 | 87.6/84.9 | 89.3/84.5 | 1m 55s | 41.7M |
| iv | FrugalScore$_e$ | BERT-Tiny | 86.3/81.1 | 95.1/87.1 | 84.5/80.2 | 84.5/80.9 | 87.6/82.3 | 1m 28s | 4.4M |
| v | FrugalScore$_e$ | BERT-Small | 85.1/81.7 | 95.7/83.6 | **91.2**/87.5 | 91.7/87.5 | 90.9/85.1 | 2m 29s | 29.1M |
| vi | FrugalScore$_e$ | BERT-Medium | 81.6/80.7 | 95.7/84.1 | 90.9/87.5 | 87.6/85.3 | 89.0/84.4 | 3m 41s | 41.7M |
| vii | FrugalScore$_f$ | BERT-Tiny | 89.7/84.5 | 95.3/**87.6** | 85.1/81.4 | 84.8/81.2 | 88.7/83.7 | 1m 28s | 4.4M |
| viii | FrugalScore$_f$ | BERT-Small | 86.8/85.1 | 96.7/85.4 | 89.5/86.2 | 91.6/88.7 | 91.2/86.3 | 2m 29s | 29.1M |
| ix | FrugalScore$_f$ | BERT-Medium | 85.4/86.3 | **97.2**/87.2 | 91.1/**88.9** | **92.3**/**91.0** | 91.5/**88.3** | 3m 41s | 41.7M |
| x | FrugalScore$_g$ | BERT-Tiny | **93.7**/86.1 | 96.2/83.9 | 90.1/87 | 89.4/84.8 | **92.3**/85.5 | 1m 28s | 4.4M |
| xi | FrugalScore$_g$ | BERT-Small | 93.2/**87.6** | 96.4/84.2 | 85/81.7 | 87.9/84.9 | 90.6/84.6 | 2m 29s | 29.1M |
| xii | FrugalScore$_g$ | BERT-Medium | **93.7**/87.5 | 96.5/84.5 | 84.8/81.6 | 87.3/84.7 | 90.6/84.6 | 3m 41s | 41.7M |

Table B.4 – System-level Pearson correlation.

| | Metric | Model | de-en | fi-en | gu-en | kk-en | lt-en | ru-en | zh-en | Macro Avg. Score | Runtime | Params |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| a | BERTScore | BERT-Tiny | 74.1 | 97.9 | 93.1 | 99.77 | 87.9 | 94.5 | 91.7 | 91.3 | 1m 22s | 4.4M |
| b | BERTScore | BERT-Small | 82.6 | 97.5 | 88.2 | **99.87** | 95.3 | 96.4 | 93.0 | 93.3 | 1m 42s | 29.1M |
| c | BERTScore | BERT-Medium | 83.7 | 97.7 | 88.2 | 99.86 | 94.4 | 96.2 | 93.5 | 93.4 | 2m 04s | 41.7M |
| d | BERTScore | BERT-Base | 89.1 | 97.8 | 89.7 | 99.72 | 96.9 | 96.9 | 95.8 | 95.1 | 2m 09s | 110M |
| e | BERTScore | RoBERTa-Large | **94.0** | 98.4 | 98.1 | 98.00 | 96.1 | 91.0 | 98.2 | 96.3 | 3m 03s | 355M |
| f | BERTScore | DeBERTa-XLarge | 93.9 | 98.3 | **98.2** | 99.18 | **98.7** | 97.1 | **98.4** | **97.7** | 3m 49s | 900M |
| g | MoverScore | BERT-Base | 88.1 | **99.1** | 91.2 | 98.58 | 96.0 | **97.2** | 96.4 | 95.2 | 64m 32s | 110M |
| i | FrugalScore$_d$ | BERT-Tiny | 81.1 | 98.6 | 94.4 | 99.80 | 92.2 | 95.4 | 93.8 | 93.6 | 1m 18s | 4.4M |
| ii | FrugalScore$_d$ | BERT-Small | 86.5 | 98.5 | 93.6 | 99.82 | 95.9 | 97.1 | 94.7 | 95.2 | 1m 35s | 29.1M |
| iii | FrugalScore$_d$ | BERT-Medium | 88.3 | 98.3 | 92.1 | 99.79 | 96.4 | 97.2 | 95.4 | 95.4 | 1m 55s | 41.7M |
| iv | FrugalScore$_e$ | BERT-Tiny | 80.2 | 97.7 | 94.9 | 99.73 | 86.4 | 94.6 | 93.7 | 92.5 | 1m 18s | 4.4M |
| v | FrugalScore$_e$ | BERT-Small | 83.9 | 98.0 | 95.2 | 99.79 | 92.4 | 97.0 | 95.1 | 94.5 | 1m 35s | 29.1M |
| vi | FrugalScore$_e$ | BERT-Medium | 88.1 | 97.9 | 93.0 | 99.78 | 94.9 | **97.8** | 96.1 | 95.4 | 1m 55s | 41.7M |
| vii | FrugalScore$_f$ | BERT-Tiny | 81.3 | 97.9 | 96.1 | 99.81 | 89.8 | 94.7 | 93.7 | 93.3 | 1m 18s | 4.4M |
| viii | FrugalScore$_f$ | BERT-Small | 85.8 | 97.7 | **96.2** | **99.85** | 95.3 | 97.3 | 95.7 | 95.4 | 1m 35s | 29.1M |
| ix | FrugalScore$_f$ | BERT-Medium | **89.9** | 97.9 | 90.8 | **99.85** | **97.6** | **97.8** | **96.9** | **95.8** | 1m 55s | 41.7M |
| x | FrugalScore$_g$ | BERT-Tiny | 81.8 | **98.9** | 95.6 | 99.73 | 92.1 | 95.6 | 94.4 | 94.0 | 1m 18s | 4.4M |
| xi | FrugalScore$_g$ | BERT-Small | 85.4 | 98.8 | 95.8 | 99.52 | 94.9 | 96.8 | 95.3 | 95.2 | 1m 35s | 29.1M |
| xii | FrugalScore$_g$ | BERT-Medium | 87.0 | 98.8 | 93.5 | 99.29 | 95.6 | 97.0 | 95.9 | 95.3 | 1m 55s | 41.7M |

Table B.5 – Summary-level Pearson correlation between the FrugalScore$_{d,e,f,g}$ and the metrics $d, e, f, g$ used to generate the annotations

| | Metric | Model | TAC-2008 | TAC-2009 | TAC-2010 | TAC-2011 | Average |
|---|---|---|---|---|---|---|---|
| i | FrugalScore$_d$ | BERT-Tiny | 91.7 | 94.7 | 97.2 | 95.1 | 94.7 |
| ii | FrugalScore$_d$ | BERT-Small | 96.9 | 97.9 | 99.0 | 98.0 | 98.0 |
| iii | FrugalScore$_d$ | BERT-Medium | 98.3 | 98.8 | 99.4 | 99.0 | 98.9 |
| iv | FrugalScore$_e$ | BERT-Tiny | 77.9 | 82.4 | 87.5 | 75.9 | 80.9 |
| v | FrugalScore$_e$ | BERT-Small | 86.9 | 90.7 | 91.6 | 89.2 | 89.6 |
| vi | FrugalScore$_e$ | BERT-Medium | 87.1 | 90.7 | 86.3 | 90.9 | 88.8 |
| vii | FrugalScore$_f$ | BERT-Tiny | 80.0 | 85.5 | 89.4 | 81.3 | 84.0 |
| viii | FrugalScore$_f$ | BERT-Small | 88.9 | 92.8 | 92.6 | 91.4 | 91.4 |
| ix | FrugalScore$_f$ | BERT-Medium | 89.9 | 92.9 | 92.1 | 93.6 | 92.1 |
| x | FrugalScore$_g$ | BERT-Tiny | 91.1 | 94.8 | 95.7 | 94.8 | 94.1 |
| xi | FrugalScore$_g$ | BERT-Small | 94.8 | 97.4 | 98.4 | 98.0 | 97.1 |
| xii | FrugalScore$_g$ | BERT-Medium | 96.4 | 98.0 | 98.9 | 98.6 | 98.0 |

Table B.6 – Segment-level Pearson correlation between the FrugalScore$_{d,e,f,g}$ and the metrics $d, e, f, g$ used to generate the annotations.

| | Metric | Model | de-en | fi-en | gu-en | kk-en | lt-en | ru-en | zh-en | Macro Avg. Score | Runtime | Params |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| a | BERTScore | BERT-Tiny | 29.7 | 32.5 | 33.9 | 52.0 | 40.5 | 30.7 | 44.2 | 37.6 | 1m 22s | 4.4M |
| b | BERTScore | BERT-Small | 30.0 | 33.6 | 34.6 | 52.4 | 42.3 | 31.8 | 49.1 | 39.1 | 1m 42s | 29.1M |
| c | BERTScore | BERT-Medium | 30.8 | 34.4 | 35.2 | 52.8 | 42.8 | 32.4 | 50.3 | 39.8 | 2m 04s | 41.7M |
| d | BERTScore | BERT-Base | 32.8 | 37.4 | 37.1 | 54.0 | 44.7 | 33.7 | 53.7 | 41.9 | 2m 09s | 110M |
| e | BERTScore | RoBERTa-Large | 35.3 | 38.7 | 38.7 | 52.0 | 45.3 | 34.3 | **58.3** | 43.2 | 3m 03s | 355M |
| f | BERTScore | DeBERTa-XLarge | **37.6** | **39.2** | **40.3** | 53.4 | **47.3** | **35.7** | 57.8 | **44.5** | 3m 49s | 900M |
| g | MoverScore | BERT-Base | 36.5 | 39.1 | 39.3 | **55.0** | 46.5 | 35.6 | 56.0 | 44.0 | 64m 32s | 110M |
| i | FrugalScore$_d$ | BERT-Tiny | 30.2 | 32.8 | 34.6 | 52.4 | 39.9 | 31.2 | 47.7 | 38.4 | 1m 18s | 4.4M |
| ii | FrugalScore$_d$ | BERT-Small | 32.6 | 35.9 | 37.1 | 54.1 | 43.5 | 33.6 | 52.3 | 41.3 | 1m 35s | 29.1M |
| iii | FrugalScore$_d$ | BERT-Medium | 32.9 | 37.0 | 37.4 | 54.4 | 44.3 | 34.1 | 53.2 | 41.9 | 1m 55s | 41.7M |
| iv | FrugalScore$_e$ | BERT-Tiny | 30.6 | 32.8 | 33.0 | 49.8 | 38.7 | 29.8 | 48.1 | 37.5 | 1m 18s | 4.4M |
| v | FrugalScore$_e$ | BERT-Small | 33.7 | 35.4 | 35.4 | 51.6 | 42.6 | 32.6 | 52.5 | 40.5 | 1m 35s | 29.1M |
| vi | FrugalScore$_e$ | BERT-Medium | 35.2 | 37.1 | 35.6 | 52.0 | 44.0 | 33.8 | 54.4 | 41.7 | 1m 55s | 41.7M |
| vii | FrugalScore$_f$ | BERT-Tiny | 30.8 | 33.1 | 34.4 | 50.8 | 39.4 | 30.4 | 47.1 | 38.0 | 1m 18s | 4.4M |
| viii | FrugalScore$_f$ | BERT-Small | 34.5 | 36.4 | 37.0 | 52.7 | 43.9 | 33.4 | 52.6 | 41.5 | 1m 35s | 29.1M |
| ix | FrugalScore$_f$ | BERT-Medium | 35.8 | **38.3** | 37.7 | 53.4 | 45.7 | 34.8 | **55.1** | 43.0 | 1m 55s | 41.7M |
| x | FrugalScore$_g$ | BERT-Tiny | 33.0 | 34.0 | 36.2 | 53.6 | 40.5 | 32.7 | 48.6 | 39.8 | 1m 18s | 4.4M |
| xi | FrugalScore$_g$ | BERT-Small | 35.6 | 37.4 | 38.9 | 55.0 | 44.8 | 34.8 | 52.8 | 42.8 | 1m 35s | 29.1M |
| xii | FrugalScore$_g$ | BERT-Medium | **36.2** | **38.3** | **39.1** | **55.6** | **45.8** | **35.3** | 54.7 | **43.6** | 1m 55s | 41.7M |

ECOLE DOCTORALE

Titre : Contributions à la Génération de Langage Naturel : Systèmes et Evaluation

Mots clés : apprentissage par transfert, résumé abstractif, génération de langage naturel, métriques d'évaluation, traitement du langage naturel, apprentissage automatique, intelligence artificielle

Résumé :
Ces dernières années, le domaine de la génération du langage naturel (GLN) a radicalement changé. Ce changement, qui peut être en partie attribué à l'avancée notable du matériel, a conduit les récents efforts du GLN à se concentrer sur des méthodes basées sur les données tirant parti de grands réseaux de neurones pré-entraînés. Cependant, ces progrès ont donné lieu à de nouveaux défis liés aux exigences de calcul, à l'accessibilité et aux stratégies d'évaluation, pour n'en nommer que quelques-uns. Dans cette thèse, nous nous intéressons principalement à contribuer aux efforts visant à atténuer ces défis.
Pour remédier au manque de modèles génératifs monolingues pour certaines langues, nous commençons par présenter BARThez et AraBART, les premiers modèles seq2seq pré-entraînés à grande échelle pour le Français et l'Arabe, respectivement. Basés sur BART, ces modèles sont particulièrement bien adaptés aux tâches génératives. Nous évaluons BARThez sur cinq tâches discriminantes du benchmark FLUE et deux tâches génératives d'un nouvel ensemble de données de résumé, OrangeSum, que nous avons créé pour cette recherche. Nous montrons que BARThez est très compétitif avec les modèles de langue française basés sur BERT tels que CamemBERT et FlauBERT. Nous poursuivons également le pré-entraînement d'un BART multilingue sur le corpus de BARThez, et montrons que notre modèle résultant, mBARThez, améliore considérablement les performances génératives de BARThez. D'autre part, nous montrons qu'AraBART obtient les meilleures performances sur plusieurs ensembles de données de résumé abstractif, surpassant des bases de référence solides.
Enfin, nous nous concentrons sur l'évaluation des systèmes GLN en proposant DATScore et FrugalScore. DATScore utilise des techniques d'augmentation des données pour améliorer l'évaluation de la traduction automatique et d'autres tâches GLN. Notre principale conclusion est que l'introduction de traductions enrichies de données des textes source et de référence est très utile pour évaluer la qualité de la traduction générée. Nous proposons également deux nouvelles stratégies de calcul de la moyenne des scores et de pondération des termes pour améliorer le processus original de calcul des scores de BARTScore. Les résultats expérimentaux sur WMT montrent que DATScore est mieux corrélé avec les méta-évaluations humaines que les autres métriques récentes de l'état de l'art, en particulier pour les langues à faibles ressources. D'autre part, FrugalScore est une approche pour apprendre une version fixe et peu coûteuse de toute métrique GLN coûteuse tout en conservant la plupart de ses performances d'origine. Des expériences avec BERTScore et MoverScore sur sur le résumé et la traduction montrent que FrugalScore est comparable avec les métriques d'origine (et parfois mieux), tout en ayant plusieurs ordres de grandeur de moins de paramètres et en s'exécutant plusieurs fois plus rapidement. En moyenne, sur l'ensemble des métriques, tâches et variantes apprises, FrugalScore conserve 96,8% des performances, s'exécute 24 fois plus rapidement et comporte 35 fois moins de paramètres que les métriques d'origine.

Title : Contributions to Natural Language Generation : Systems and Evaluation

Keywords : transfer learning, abstractive summarization, natural langauge generation, evaluation metrics, natural language processing, machine learning, artificial intelligence

Abstract : In recent years, the Natural Language Generation (NLG) field has changed drastically. This shift, which can be partially attributed to the notable advance in hardware, led to recent efforts in NLG to be focused on data-driven methods leveraging large pretrained Neural Networks (NNs). However, this progress gave rise to new challenges related to computational requirements, accessibility, and evaluation strategies, to name a few. In this dissertation, we are primarily concerned with contributing to the efforts to mitigate these challenges.
To address the lack of monolingual generative models for some languages, we start by introducing BARThez and AraBART, the first large-scale pretrained seq2seq models for French and Arabic, respectively. Being based on BART, these models are particularly well-suited for generative tasks. We evaluate BARThez on five discriminative tasks from the FLUE benchmark and two generative tasks from a novel summarization dataset, OrangeSum, that we created for this research. We show BARThez to be very competitive with state-of-the-art BERT-based French language models such as CamemBERT and FlauBERT. We also continue the pretraining of a multilingual BART on BARThez' corpus, and show our resulting model, mBARThez, to significantly boost BARThez' generative performance. On the other hand, We show that AraBART achieves the best performance on multiple abstractive summarization datasets, outperforming strong baselines.
Finally, we focus on the NLG system evaluation by proposing DATScore and FrugalScore. DATScore uses data augmentation techniques to improve the evaluation of machine translation and other NLG tasks. Our main finding is that introducing data augmented translations of the source and reference texts is greatly helpful in evaluating the quality of the generated translation. We also propose two novel score averaging and term weighting strategies to improve the original score computing process of BARTScore. Experimental results on WMT show that DATScore correlates better with human meta-evaluations than the other recent state-of-the-art metrics, especially for low-resource languages. On the other hand, FrugalScore is an approach to learn a fixed, low-cost version of any expensive NLG metric while retaining most of its original performance. Experiments with BERTScore and MoverScore on summarization and translation show that FrugalScore is on par with the original metrics (and sometimes better), while having several orders of magnitude fewer parameters and running several times faster. On average overall learned metrics, tasks, and variants, FrugalScore retains 96.8% of the performance, runs 24 times faster, and has 35 times fewer parameters than the original metrics.

Institut Polytechnique de Paris
91120 Palaiseau, France