UNIVERSITÉ DE GENÈVE
Département d'Informatique

FACULTÉ DES SCIENCES
Professeur Bastien Chopard

HES-SO – HAUTE ÉCOLE DE GESTION DE GENÈVE
Département des Sciences de l'Information

Professeur Patrick Ruch

TransBERT: Leveraging Automatic Translation for Domain-Specific Knowledge Transfer

THÈSE

Présentée à la Faculté des Sciences de l'Université de Genève
Pour obtenir le grade de docteur ès sciences, mention Bio-informatique

Par

Julien David Marc Knafou

Bottens (VD)

Thèse N° 5898

GENÈVE

2025

Acknowledgements

Tout d'abord, je souhaite exprimer ma sincère gratitude à Patrick Ruch, tant pour la confiance qu'il m'a témoignée dans mes recherches que pour le soutien qu'il m'a apporté tout au long de cette thèse.

Je souhaite aussi présenter mes remerciements les plus sincères à Arnaud Gaudinat et Douglas Teodoro, qui m'ont offert la chance de débuter dans le milieu académique. Ma reconnaissance s'étend aussi à l'ensemble de mes collègues pour leur collaboration précieuse.

Enfin, je suis particulièrement reconnaissant envers ma famille, mon père, ma soeur et ma mère, pour leur soutien inconditionnel durant toutes ces années.

Abstract

Natural Language Processing (NLP) and machine learning technologies have transformed many facets of life science research and healthcare in recent years. However, the development of advanced Language Models (LMs) and NLP tools for life sciences has largely been limited to English because of the scarcity of scientific publications in other languages. This language barrier presents significant difficulties for international researchers and healthcare professionals, limiting their ability to use the most recent NLP features in their native languages. The aim of this thesis is to explore innovative methods for developing competitive life science LMs for non-English languages, focusing on French, by exploiting recent progress in Machine Translation (MT). The study was structured around two primary hypotheses:

1. The current state of MT enables the development of a LM trained entirely on an automatically translated corpus, maintaining competitiveness with State-of-the-Art (SOTA) models in the field.

2. Domain-Specific (DS) tokenization enhances the performance of Pre-trained Language Models (PLMs) on specialized downstream tasks.

To support the first hypothesis, TransBERT, a French life science model, was trained exclusively on an extensive collection of automatic translated MEDLINE abstracts. Specifically, the advanced M2M-100 translation model was deployed to translate more than 22M MEDLINE abstracts from English to French, creating TransCorpus, the most extensive French life science corpus to date, encompassing roughly 36GB of raw text. Subsequently, following the training of a BERT architecture on a Masked Language Model (MLM) task utilizing this synthetic corpus, TransBERT was evaluated against two SOTA PLMs through comprehensive experiments. The first model, CamemBERT, is a French LM trained on a general corpus, whereas the latter, DrBERT, is a life science focused LM developed using a native French corpus. The performance of these models was assessed on various life science NLP tasks employing an adaptation of DrBenchmark, the first French biomedical benchmark for Natural Language Understanding (NLU). Even though TransBERT was pre-trained solely on translated data, our results showed that it achieved competitive or better performance compared to these leading models. Statistical analyses validated the strong performance of TransBERT in two key tasks of the field, classification and Named Entity Recognition (NER).

To evaluate the second hypothesis, the effect of DS tokenization on model performance was analyzed by comparing TransBERT, which employs a DS tokenizer trained on TransCorpus, with cTransBERT, an equivalent model architecture pre-trained on the same corpus but using CamemBERT's general domain tokenizer. Using the same benchmark, our analysis shows that the model using the DS

tokenizer repeatedly enhanced performance while getting statistical significance for NER. These results highlight the necessity of tailoring the tokenizer to the specific domain when developing specialized LMs.

Beyond just validating our core hypotheses, this research makes several key contributions to the field of multilingual life science NLP. Firstly, we illustrate a scalable method for swiftly developing competitive DS LMs for low-resource languages by leveraging high-quality MT. This approach can potentially be applied to other domains and language pairs. Secondly, we provide TransCorpus as a valuable new resource for life science NLP research. Finally, our comprehensive evaluation framework and statistical analysis methodology offer a rigorous way to compare LMs performance that goes beyond simple metric comparisons.

This thesis introduces innovative strategies for bridging the linguistic gaps in life science NLP by leveraging MT and DS pre-training. The success of TransBERT demonstrates that it is feasible to develop highly effective DS LMs for non-English languages, even in the absence of extensive native corpora. These insights have significant implications for democratizing access to advanced NLP capabilities across various languages and domains. Future research can build on this foundation to further improve cross-lingual transfer learning and domain adaptation techniques, ultimately aiming for truly multilingual biomedical Artificial Intelligence (AI) systems that can benefit researchers and clinicians worldwide.

Résumé

Les technologies de Traitement du Langage Naturel (TLN) et d'apprentissage automatique ont transformé de nombreux aspects de la recherche en sciences de la vie et dans le domaine de la santé ces dernières années. Cependant, le développement de Modèles de Langage (ML) avancés et d'outils de TLN pour les sciences de la vie a été largement limité à l'anglais en raison de la rareté des publications scientifiques dans d'autres langues. Cette barrière linguistique présente des difficultés importantes pour les chercheurs internationaux et les professionnels dans le domaine de la santé, limitant leur capacité à utiliser les fonctionnalités de TLN les plus récentes dans leur langue maternelle. L'objectif de cette thèse est d'explorer des méthodes innovantes pour développer des MLs compétitifs en sciences de la vie pour les langues non-anglaises, en se concentrant sur le français. Pour ce faire, cette thèse exploite les progrès récents en Traduction Automatique (TA). L'étude a été structurée autour de deux hypothèses principales :

1. L'état actuel de la TA permet le développement d'un ML entraîné entièrement sur un corpus automatiquement traduit, tout en restant compétitif avec les modèles de pointe dans le domaine.

2. L'entrainement d'un tokenizer spécifique au domaine (SD) améliore les performances des Modèles de Langage Pré-entraînés (MLPs) sur des tâches spécialisées.

Pour soutenir la première hypothèse, TransBERT, un modèle français en sciences de la vie, a été entraîné exclusivement sur une vaste collection de résumés MEDLINE automatiquement traduits. Plus précisément, le modèle de traduction avancé M2M-100 a été déployé pour traduire plus de 22 millions de résumés MEDLINE de l'anglais vers le français, créant TransCorpus, le corpus français en sciences de la vie le plus grand à ce jour, englobant environ 36 Go de texte brut. Par la suite, après l'entraînement d'une architecture BERT sur une tâche de Modèle de Langage Masqué (MLM) utilisant ce corpus synthétique, TransBERT a été comparé à deux MLPs de pointe sur plusieurs tâches. Le premier modèle, CamemBERT, est un PML français, tandis que le second, DrBERT, est un PML développé à partir de documents biomédicaux natifs français. Les performances de ces modèles ont été évaluées sur diverses tâches de TLN en sciences de la vie en utilisant une adaptation de DrBenchmark, le premier benchmark biomédical français pour la Compréhension du Langage Naturel (CLN). Bien que TransBERT ait été pré-entraîné uniquement sur des données traduites, nos résultats ont montré qu'il atteignait des performances compétitives ou meilleures par rapport aux modèles de pointe. Des tests statistiques ont confirmé l'efficacité de TransBERT dans deux tâches essentielles du domaine, à savoir la classification et la reconnaissance des entités nommées (REN).

Pour évaluer la seconde hypothèse, l'effet de la tokenisation spécifique au domaine (SD) sur les performances du modèle a été analysé en comparant TransBERT, qui utilise un tokenizer SD entraîné sur TransCorpus, avec cTransBERT, un modèle d'architecture équivalente pré-entraîné sur le même corpus, mais utilisant le tokenizer de domaine général de CamemBERT. En utilisant le même benchmark, notre analyse montre que TransBERT obtient des résultats compétitifs voir meilleurs sur toutes les tâches, et le confirme avec un test statistique en REN. Ces résultats soulignent la nécessité d'adapter le tokenizer au domaine lors du développement de MLs spécialisés.

Au-delà de la simple validation de nos hypothèses principales, cette recherche apporte plusieurs contributions clé au domaine du TLN multilingue en sciences de la vie. Premièrement, nous illustrons une méthode évolutive pour développer rapidement des MLs SD compétitifs pour les langues à faibles ressources en exploitant la traduction automatique de haute qualité. Cette approche peut potentiellement être appliquée à d'autres domaines et paires de langues. Deuxièmement, nous fournissons TransCorpus comme une nouvelle ressource précieuse française pour la recherche en TLN en sciences de la vie. Enfin, notre cadre d'évaluation et notre méthodologie d'analyse statistique offrent une façon rigoureuse de comparer les performances des PMLs qui va au-delà des simples comparaisons de métriques.

Cette thèse introduit des stratégies innovantes pour combler les lacunes linguistiques dans le TLN des sciences de la vie en exploitant la TA et le pré-entraînement SD. Le succès de TransBERT démontre qu'il est possible de développer des ML SD hautement efficaces pour les langues non-anglaises, même en l'absence de corpus natifs. Ces résultats ont des implications significatives pour démocratiser l'accès aux capacités avancées de TLN à travers diverses langues et domaines. Les recherches à venir pourront s'appuyer sur cette fondation pour perfectionner encore les techniques d'apprentissage par transfert inter-langues et d'adaptation aux différents domaines, dans le but de développer des systèmes d'Intelligence Artificielle (IA) biomédicaux véritablement multilingues, profitant ainsi aux chercheurs et cliniciens à l'échelle mondiale.

Table of Content

List of Figures

List of Tables

List of Equations

Acronyms

AI Artificial Intelligence.

ANOVA Analysis Of Variance.

AUC Area Under the Cruve.

B billion.

BERT Bidirectional Encoder Representations from Transformers.

BLEU Bi-Lingual Evaluation Understudy.

BLUE Biomedical Language Understanding Evaluation.

BLURB Biomedical Language Understanding & Reasoning Benchmark.

BPE Byte Pair Encoding.

BT Backtranslation.

CBOW Continuous Bag-of-Words.

ChEMU Cheminformatics Elsevier Melbourne University.

CI Confidence Interval.

CLEF Conference and Labs of the Evaluation Forum.

CNN Convolutional Neural Network.

COAP COVID-19 Open Access Project.

CORD-19 COVID-19 Open Research Dataset.

DEFT DÉfi Fouille de Textes.

DL Deep Learning.

DS Domain-Specific.

EBMT Example-Based Machine Translation.

EHRs Electronic Health Records.

ELMo Embeddings from Language Models.

EMR Exact Match Ratio.

GloVe Global Vectors for Word Representation.

GLUE General Language Understanding Evaluation.

GPT Generative Pre-trained Transformer.

GPU Graphics Processing Unit.

GRU Gated Recurrent Units.

HPC High-Performance Computing.

HPO Hyperparameter Optimization.

IOB Inside-Outside-Beginning.

IR Information Retrieval.

k thousand.

LLaMA Large Language Model Meta AI.

LLM Large Language Model.

LM Language Model.

LSTM Long Short Term Memory.

M million.

M2M Many-to-Many.

MBR MEDLINE/PubMed Baseline Repository.

MCQA Multiple-Choice Question Answering.

MeSH Medical Subject Headings.

MLM Masked Language Model.

MT Machine Translation.

NCBI National Center for Biotechnology Information.

NER Named Entity Recognition.

NIH National Institutes of Health.

NLG Natural Language Generation.

NLM National Library of Medicine.

NLP Natural Language Processing.

NLTK Natural Language Toolkit.

NLU Natural Language Understanding.

NMT Neural Machine Translation.

NN Neural Network.

NRA Normalized Ranking Average.

NSP Next Sentence Prediction.

OOV Out-of-Vocabulary.

OSCAR Open Super-large Crawled Aggregated coRpus.

PLLs Pseudo-Log-Likelihood scores.

PLM Pre-trained Language Model.

PMC PubMed Central.

POS Part-Of-Speech.

PPPL Pseudo-Perplexity.

QA Question Answering.

RBMT Rule-Based Machine Translation.

RE Relation Extraction.

RMSE Root Mean Squared Error.

RNN Recurrent Neural Network.

RoBERTa Robustly optimized BERT approach.

Seq2Seq Sequence-to-Sequence.

SMBO Sequential Model-Based Optimization.

SMT Statistical Machine Translation.

SOTA State-of-the-Art.

SQuAD Stanford Question Answering Dataset.

SRL Semantic Role Labeling.

SRW Semantically Related Words.

STS Semantic Textual Similarity.

SWAG Situations with Adversarial Generations.

T5 Text-to-Text Transfer Transformer.

TF-IDF Term Frequency-Inverse Document Frequency.

TREC Text REtrieval Conference.

UniProtKB UniProt Knowledgebase.

W-NUT Workshop on Noisy User-generated Text.

W2V Word2Vec.

WMT Workshop on Machine Translation.

WWM Whole-Word Masking.

Chapter 1

Introduction

1.1 Motivation

Natural Language Processing (NLP) has become increasingly essential in the life sciences domain, revolutionizing the manner in which researchers and healthcare professionals manage substantial amounts of unstructured data. Indeed, considering that most biomedical data appear in unstructured formats, NLP tools are indispensable to extract important insights from scientific literature, clinical notes, and patient records. In practice, NLP methods are used to construct and update biomedical knowledge graphs, enabling swift access to and integration of the latest research outcomes from numerous clinical studies (Nicholson and Greene, 2020). In the field of drug discovery, AI-driven Language Model (LM), including NLP methods, are employed to recognize potential drug targets and expedite various phases of pharmaceutical development (Khan et al., 2022). Furthermore, NLP approaches are utilized to process free-text information in electronic health records, aiding in the development of clinical decision support systems that can help healthcare providers make more informed decisions (Demner-Fushman et al., 2009). One notable advancement is the development of architectures like BERT-based models, which have shown near-human performance in categorizing adverse drug reaction reports. This underscores their ability to rapidly process and classify safety information in pharmacovigilance (Bergman et al., 2023). As the volume of biomedical data continues to expand rapidly, NLP has become an essential tool for uncovering the hidden value in unstructured information and driving innovation within the life sciences sector.

However, the scarcity of life science data in languages other than English significantly hinders the development of LMs in this domain. This linguistic gap stems from the historical dominance of English in the scientific literature and international research collaborations. Consequently, most high-quality scientific papers, clinical trial records, and medical databases are written mostly in English. This lack of linguistic diversity creates substantial barriers for non-English speaking researchers and healthcare professionals, limiting their access to recent scientific findings and potentially reducing progress in their regions. Furthermore, this linguistic imbalance obstructs the creation of robust NLP tools crafted for life sciences purposes. The lack of comprehensive datasets in multiple languages presents a major hurdle in training models that can accurately interpret or generate scientific content or annotation in various linguistic contexts. To tackle this problem,

one approach could be to encourage scientific publishing in multiple languages, construct annotated datasets in various languages, or even develop cross-lingual transfer learning methods that utilize the extensive English-language data while adapting to other languages.

The progress in the development of Domain-Specific (DS) LMs for life sciences has been remarkable in recent years, especially with the introduction of models such as BioBERT (Lee et al., 2019) and PubMedBERT (Gu et al., 2020) for English. BioBERT, which was introduced in 2019, was trained on a vast amount of biomedical literature and showed significant improvement over general-domain BERT models in various biomedical text mining tasks. PubMedBERT, released in 2020, took the advancements further by training exclusively on PubMed abstracts and full-text articles from scratch, outperforming BioBERT's performance on many biomedical NLP benchmarks. In the French language domain, similar initiatives have recently been launched to meet the demand for specialized biomedical language models. CamemBERT-bio (Touchent et al., 2023), introduced in 2023, continually pre-trains the original CamemBERT (Martin et al., 2020) model on a new public French biomedical dataset, achieving notable improvements in various biomedical tasks. Likewise, DrBERT (Labrak et al., 2023b), also published in 2023, was trained from scratch solely on a life science corpus. These models employed different pre-training strategies and were assessed on a range of biomedical tasks, showing comparable or superior performance to existing French models. The arrival of CamemBERT-bio and DrBERT signifies a major advancement in providing specialized LMs for French biomedical text mining, reflecting the progress made in English LMs and meeting the growing demand for robust NLP tools in French-speaking medical and research communities. Nonetheless, it is important to note that although these French models show progress, they leverage only a small fraction of the data compared to their English counterparts, highlighting the sustained disparity in DS foreign resources.

The domain of Machine Translation (MT) has recently experienced significant advances with the introduction of advanced models such as M2M-100 in 2023 (Fan et al., 2020). This progress is marked by a movement towards more comprehensive and efficient multilingual translation frameworks. Before the advent of M2M-100, many multilingual models heavily relied on English as a pivot language, often resulting in diminished nuance and accuracy for translations between non-English languages. Facebook AI's development of M2M-100 represented a major leap forward, as it became the first model capable of translating directly between any two of 100 languages without utilizing English data. This methodology has demonstrated up to about 8 points improvement over English-centric models on the BLEU metric, a standard measure for MT quality. M2M-100 was trained with an extensive dataset encompassing 2,200 language directions, which is 10 times larger than previous top English-centric multilingual models, allowing it to maintain meaning more effectively during translations between different language pairs. This progress has facilitated more equitable and precise translations, particularly helping speakers of underrepresented languages and enhancing communication across a wider range of global communities.

1.2 Model Scope

This thesis focuses only on Natural Language Understanding (NLU) models, which are designed to transform sequences of words into vector representations. In contrast, Natural Language Generation (NLG) models are crafted to produce words derived from a given context. Although both models employ similar technologies to some extent, the methods for training the encoding component differ considerably from those used in the decoding one. For example, at training time, the generative module focuses solely on the left-side context, since the model is meant to generate a word conditionally on the past words. Conversely, the NLU module, that is, the encoder, is not limited to attend only to the preceding text, as each word is informed by its neighboring words to obtain a contextual representation for each word. This awareness allows NLU models to perform sequence-level classification and clustering, as well as word-level tasks such as Named Entity Recognition (NER) and Part-Of-Speech (POS). Consequently, NLU may be regarded as analogous to the process of reading, whereas NLG can be likened to the act of writing.

Understanding the difference between these two approaches for solving NLP tasks is essential as both models are widely used in the literature. For example, in classification tasks, a generative model needs a particular prompting such as "Categorize the following text into one of these classes: positive, negative, or neutral. Text:", allowing it to predict a response in text form, one word at a time. Because a generative model produces a text sequence, inadequate training might lead to random or irrelevant output during a classification task. The current State-of-the-Art (SOTA) generative models are typically Transformer-based, either as encoder-decoder or decoder-only architectures, and they require more parameters than NLU models to work properly, since they tend to be trained on a wide range of tasks such as summarizing, text enhancement, and more.

Conversely, when using a NLU model for a classification task, the usual bi-directional vector representation of the sequence is fed to a classifier to produce class probabilities. This implies that NLU models demand custom-designed datasets for training and deliver a one-step classification whose output is directly pertinent to the specific problem. Although this enables them to make sequence classification predictions in a single pass, it also implies that NLU models are incapable of performing generative tasks such as generative Question Answering (QA)/summarization or text enhancement.

In summary, generative models act as versatile incremental solutions for broad problems, while NLU models respond more specifically and efficiently to defined problems, necessitating task-specific training data.

1.3 Hypothesis

The fast progress in MT and LMs has created new opportunities to mitigate linguistic disparities in specialized fields such as life sciences. With MT systems such as M2M-100 showing unmatched proficiency in translating directly between various language pairs, an intriguing hypothesis arises.

The current state of Machine Translation (MT) enables the development of a Language Model (LM) trained entirely on an automatically translated corpus, maintaining competitiveness with State-of-the-Art (SOTA) models in the field.

This hypothesis questions the conventional dependence on native language corpora for the development of DS LMs and proposes that high-quality MT might help close the gap in foreign scientific data. Should this be proven effective, this method could reshape language boundaries by creating DS corpora for languages that lack DS data, especially in areas like biomedicine, where the majority of resources are in English. Such an advancement would not only make cutting-edge NLP tools accessible across different languages, but could also facilitate scientific research and the distribution of knowledge in regions where English is not the primary language.

Although DS LMs have demonstrated significant advancements in various specialized domains, research that specifically measures the effect of DS tokenization on model performance in downstream tasks is surprisingly sparse. Most studies have emphasized the advantages of pre-training on DS corpora or task-specific fine-tuning, often neglecting or assuming the importance of tokenization. Tokenization, which involves segmenting text into units meaningful for LMs, can drastically affect a model's ability to capture DS nuances and terminology. In specialized areas such as biomedicine, characterized by complex technical jargon and abbreviations, conventional tokenizers may not adequately reflect the vocabulary, potentially impairing model performance. This gap in our knowledge prompts a critical research question and supports our second hypothesis.

Domain-Specific (DS) tokenization enhances the performance of Pre-trained Language Models (PLMs) on specialized downstream tasks.

This hypothesis indicates that customizing the tokenization process to align with DS vocabulary and linguistic characteristics of a particular domain can yield additional enhancements in model performance, surpassing the gains obtained from DS LM pre-training alone.

1.4 Manuscript Overview

This thesis is structured to explore the use of automatic translation for DS knowledge transfer. The subsequent paragraphs describe the arrangement and substance of each chapter, giving an overview of the research discussed in this document. Each chapter extends the groundwork established in the preceding one, leading to a thorough evaluation of our suggested methods and their impact on the domain.

Chapter 2 offers an extensive summary of NLP tools and their increasing influence in daily life. It investigates NLU tasks within various life sciences fields, including biomedical, clinical, and chemical areas, and reviews evaluation techniques such as the Biomedical Language Understanding & Reasoning Benchmark (BLURB) benchmark. The chapter follows the development of LM from Word2Vec (W2V) to Bidirectional Encoder Representations from Transformers (BERT), exploring how computers interpret text and presenting essential concepts such as subword

segmentation and Sequence-to-Sequence (Seq2Seq) models. It finishes with a review of MT advancements and research utilizing translation for NLU system training.

Chapter 3 provides the methodological groundwork for the forthcoming chapters, offering a modular way of seeing each section of the research methods. It elaborates on the development of TransCorpus, an innovative fully translated life sciences corpus, and describes the training procedures for TransTokenizer, TransBERT, and cTransBERT. The chapter also presents datasets and tasks for fine-tuning, establishing a basis for future experiments. While the main emphasis is on methodology, interim results are provided when possible to ensure each module delivers the correct output for the subsequent one.

Chapter 4 extends the framework introduced in Chapter 3 to examine the hypothesis that present MT quality facilitates the development of competitive LMs trained on automatically translated corpora. It establishes a comprehensive reporting system incorporating statistical testing to evaluate the model's performance on datasets derived from DrBenchmark, a newly introduced life science benchmark in French. The chapter ends by evaluating TransBERT's competitiveness against CamemBERT and DrBERT on genuine French downstream tasks using task-level statistical analysis.

Chapter 5 expands the experimental setup to evaluate TransBERT versus cTransBERT, with the goal of confirming the hypothesis that DS tokenization improves Pre-trained Language Models (PLMs) performance on specific downstream tasks. This chapter modifies the statistical testing approach for comparing two models and emphasizes combined results per task to underscore the effects of tokenization.

Chapter 6 offers a comprehensive summary of the research performed in this thesis. It starts by revisiting the key hypotheses and methods used throughout the investigation. Subsequently, it analyzes the limitations of the study, particularly addressing the challenges in applying the findings to different domains and languages, especially those with low resources. A notable segment is devoted to proposing future research directions, such as exploring new languages, creating multilingual models, and comparing the results with Large Language Models (LLMs). The chapter wraps up by underscoring the thesis's contributions to the NLP field in life sciences and highlighting the potential influence of the developed resources on future research in DS NLP, with a focus on languages with limited resources.

Chapter 2

Literature Review

In this literature review, Natural Language Processing tools will be presented to highlight the increasing impact of this active research field on our daily lives. Subsequently, a concise overview of Natural Language Understanding (NLU) tasks in various life sciences sectors, i.e., biomedical, clinical, and chemical, will be provided along with evaluation methods through benchmarks such as BLURB. Prior to discussing the progress of Language Models from Word2Vec to BERT, we will show how computers handle text from its raw form, its treatment through subword segmentation algorithms, up to modeling approaches including Sequence-to-Sequence, a Natural Language Generation model that combines two Recurrent Neural Networks. Following the introduction of the Transformer model, we will delve into the evolution of Machine Translation leading to the current State-of-the-Art model. Finally, we will conclude with a discussion of relevant research leveraging translation for NLU system training.

2.1 Natural Language Processing tools

People often use Natural Language Processing (NLP) tools unknowingly, as these tools are designed to operate seamlessly, allowing users to focus on their tasks. For instance, when writing a document such as this thesis, functions like spell checking, LaTeX code suggestions, tables of contents/acronyms, and bibliography generation are managed in the background, enabling the author to focus on the content.

These types of features have been around for a long time, improving over the years. For example, in previous versions of Microsoft Word, spell-checking was a basic task of matching words with a dictionary. Today, thanks to the progress of Artificial Intelligence (AI), it is possible not only to correct grammar errors in a sentence but also to enhance its overall quality. Even more impressive, generative models like ChatGPT[1], a Large Language Model (LLM) combined with instruction tuning tasks, are able to assist users with features such as scientific papers writing[2], webpages summarizing, or questions answering as if they were interacting with someone with extensive encyclopedic knowledge.

Certain research areas within NLP have seen higher levels of activity compared to others. In the 1980s, the focus of automatic translation was primarily on

[1] `https://chat.openai.com/`

[2] `https://jenni.ai/`

translating sentences word-by-word (Brown et al., 1988), without considering the context of neighboring words. More recently, after the emergence of Transformer models (Vaswani et al., 2017), computers have become proficient at translating texts effectively, even when they include technical terms or spelling errors. It is possible to display a website in one's preferred language, as web browsers automatically translate any foreign language content. We have reached a point where generated answers from AI models can be used as a source of inspiration, and with basic knowledge coupled with some fact-checking, writing a trustworthy essay on a wide range of topics seems possible.

2.1.1 Natural Language Understanding

Natural Language Understanding (NLU) is a subset of NLP, which aims to extract meaning from a textual source. There exist a multitude of tasks, e.g., text classification, Named Entity Recognition, Question Answering, each of which focuses on different "understanding" facets of the language. NLU systems can be evaluated on one or more tasks using one or more metrics. In the following subsections, the most common tasks and metrics, as well as common training practices such as cross-validation, will be defined.

2.1.1.1 Classification

Classification is employed to assign a predefined class to a text sequence. Sentiment analysis is a well-known classification task where a model determines whether a sequence conveys a positive or negative sentiment. As illustrated in Figure 2.1, there are three classification types: binary, multi-class, which aims to differentiate mutually exclusive categories, and multi-label, which can be viewed as multiple independent binary classification tasks.

A softmax function is typically applied in the final layer of a Neural Network (NN), following the projection of the last hidden layer to a vector with a dimension C (the number of classes). This function transforms the raw unnormalized scores, known as logits, into exclusive probabilities for each class. The softmax function computes the exponential of every element in an input vector and subsequently normalizes these results by dividing them by the sum of all the exponentials. This normalization step guarantees that the resulting output probabilities add up to 1, making the function suitable for both binary and multi-class classification problems. The formula for the softmax function is given by:

$$softmax(x_i) = \frac{e^{x_i}}{\sum_j^C e^{x_j}} \tag{2.1}$$

Usually, when softmax is defined at the end of a model, it is conjointly deployed with the categorical cross-entropy loss defined in Equation 2.2, which is then the objective function that needs to be optimized in order to fit the model.

$$\mathcal{L}_{cross-entropy} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{2.2}$$

Where y_i is the actual label and $\hat{y}$ is the prediction of the system.

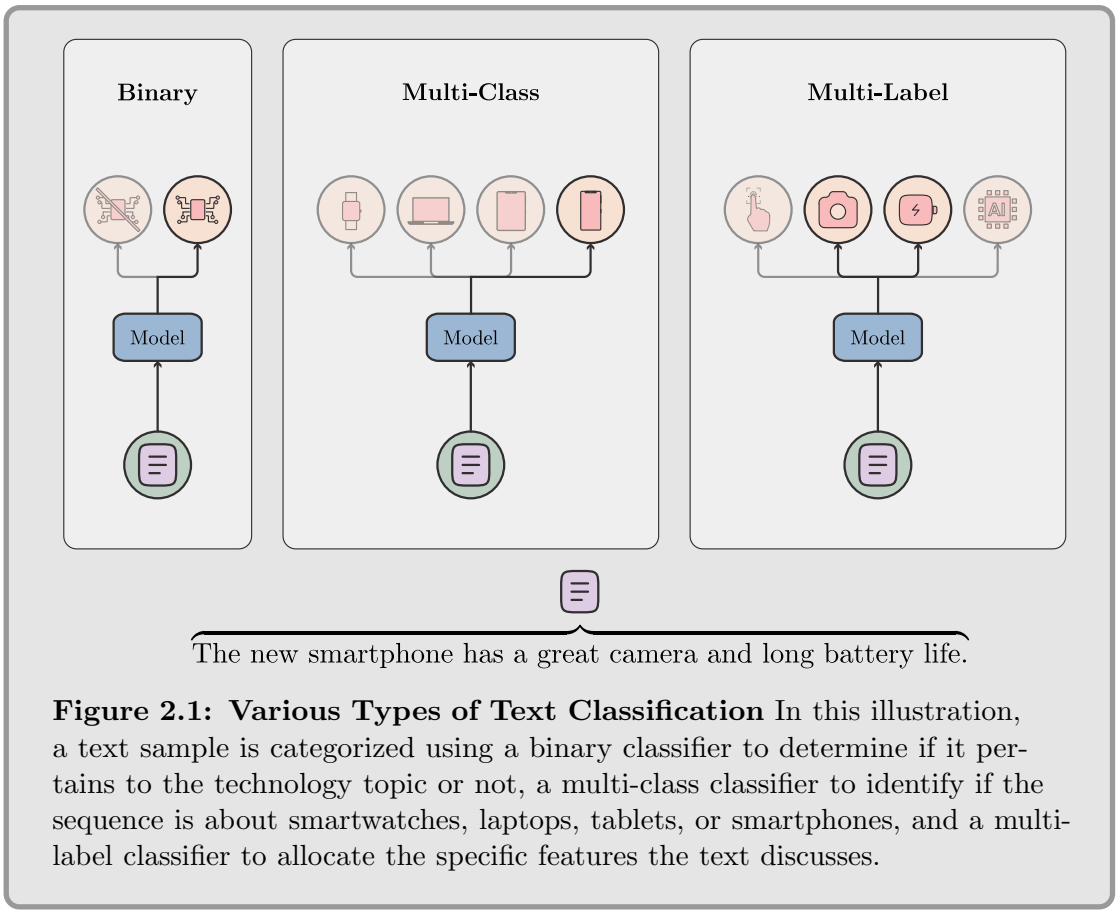

Figure 2.1: Various Types of Text Classification In this illustration, a text sample is categorized using a binary classifier to determine if it pertains to the technology topic or not, a multi-class classifier to identify if the sequence is about smartwatches, laptops, tablets, or smartphones, and a multi-label classifier to allocate the specific features the text discusses.

2.1.1.2 Named Entity Recognition & Part-Of-Speech Tagging

Named Entity Recognition (NER) and Part-Of-Speech (POS) Tagging are fundamentally word-level classification tasks within a text sequence, allowing the identification of various categories. POS tagging usually involves associating each word with a semantic category, whereas NER assigns a sequence of words to a class, such as a company or city name.

To segment entities into word sequences, it is typical to extend POS tagging using the Inside-Outside-Beginning (IOB) format. This format involves assigning each word a tag that indicates whether it is the beginning, inside, or outside of a given entity. Assigning a label to a word is challenging because of polysemy, as a word can possess different meanings. For this reason, recent word representation methods, such as self-attention (Section 2.3.2.3), consider the surrounding context.

Figure 2.2 demonstrates the use of a multi-class NER model to detect names, locations, and dates. Notably, using the IOB format, to categorize 'Kat Graham', the model first had to classify 'Kat' as `B-Name` and 'Graham' as `I-Name`. Similar to other classification tasks, NER can cover binary, multi-class, or multi-label scenarios, with the ability to include nested classes, i.e., a class within a class. It's important to mention that IOB does not support nested classes, so custom solutions are often developed for such scenarios.

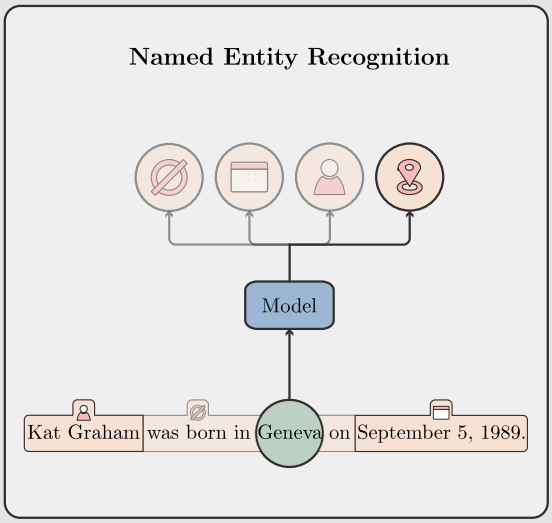

Figure 2.2: Example of a NER Application This figure demonstrates how a NER model assigns classes to individual words, resulting in a refined text containing entities such as 'dates', 'names', and 'locations'.

2.1.1.3 Question Answering

In its extractive form, Question Answering (QA) involves locating answers to questions within a sequence. Various methods can be used for solving Question Answering (QA), with one of the most straightforward approaches being to view the task as a multi-class NER problem. In this setup, the model is required to categorize each word in the context sequence into one of the classes: `[None, start, end]`. The segment between the predicted positions `start` and `end` is identified as the answer to the question. As illustrated in Figure 2.3, the model takes a question concatenated with a passage that potentially contains the answer, classifies the `start` and `end` positions, if any, and then generates the correct answer accordingly.

2.1.1.4 Sementic Textual Similarity

Semantic Textual Similarity (STS) seeks to assess the degree of similarity between two texts in terms of their meanings. Contrary to basic lexical similarity, which compares the surface forms of words, STS targets the underlying semantics, presenting a more difficult problem due to the intricacies of natural language. Addressing STS generally involves performing a regression on a dataset containing sentence pairs annotated for similarity. Figure 2.4 provides an example in which two sentences are compared to one another, with the most similar receiving a high similarity score and the other a low score.

2.1.1.5 Metrics: Precision, Recall, F_1-Score & More

In order to evaluate the performance of models in a given task, it is essential to compute metrics that compare the model's predictions with the actual values (gold standard). Precision, Recall, and F_1-Score are the standard metrics used in the field and can be applied to the tasks discussed in the previous sections.

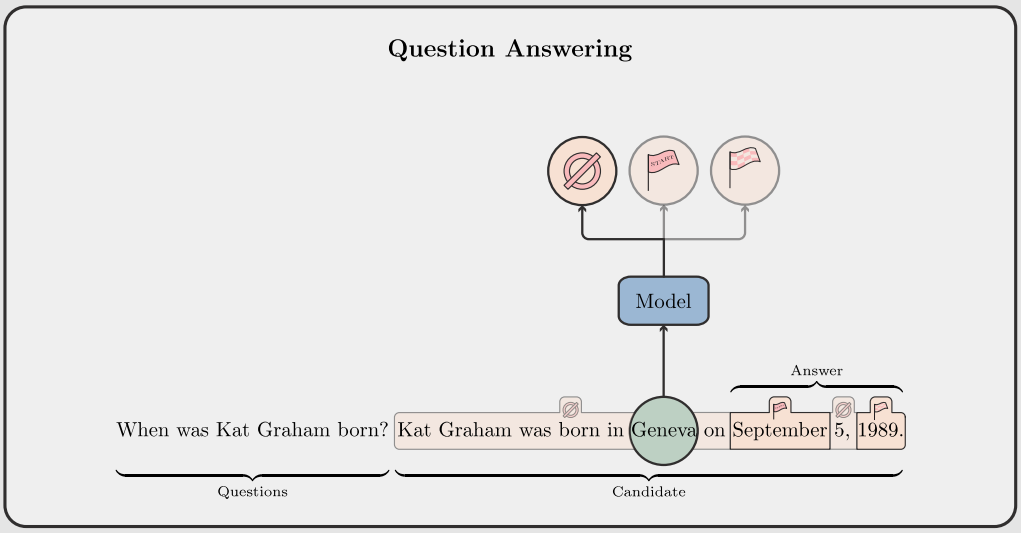

Figure 2.3: Example of a QA Application A question is appended to a candidate that may include the answer. The model then categorizes each word to identify the beginning and end of the answer using the special classes `start` and `end`, respectively. The answer is determined by selecting the words between the `start` and `end` tokens.

A prediction is called positive (P) when a class/label is detected and negative (N) otherwise. Once compared to the gold standard, an observation is called true (T) when correctly classified and false (F) otherwise. Thus, for a given class, TP and TN are, respectively, positive and negative instances that are classified correctly, while FP and FN instances are misclassified observations that had been classified by the model as positive and negative, respectively.

After comparing the predictions with the gold standard throughout the dataset, the Precision can be computed in the following manner:

$$Precision = \frac{TP}{TP + FP} \tag{2.3}$$

Precision refers to the ratio of accurately predicted positive instances to all instances that were predicted as positive, whereas Recall represents the ratio of accurately predicted positive instances to all actual positive instances.

$$Recall = \frac{TP}{TP + FN} \tag{2.4}$$

There exist multiple ways of aggregating Precision and Recall together, the F_1-Score, which is the harmonic mean of both Precision and Recall, is the commonest.

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{2.5}$$

It is important to note that all the metrics are computed using a dataset to which the model has not been exposed, namely the test set. This enables the

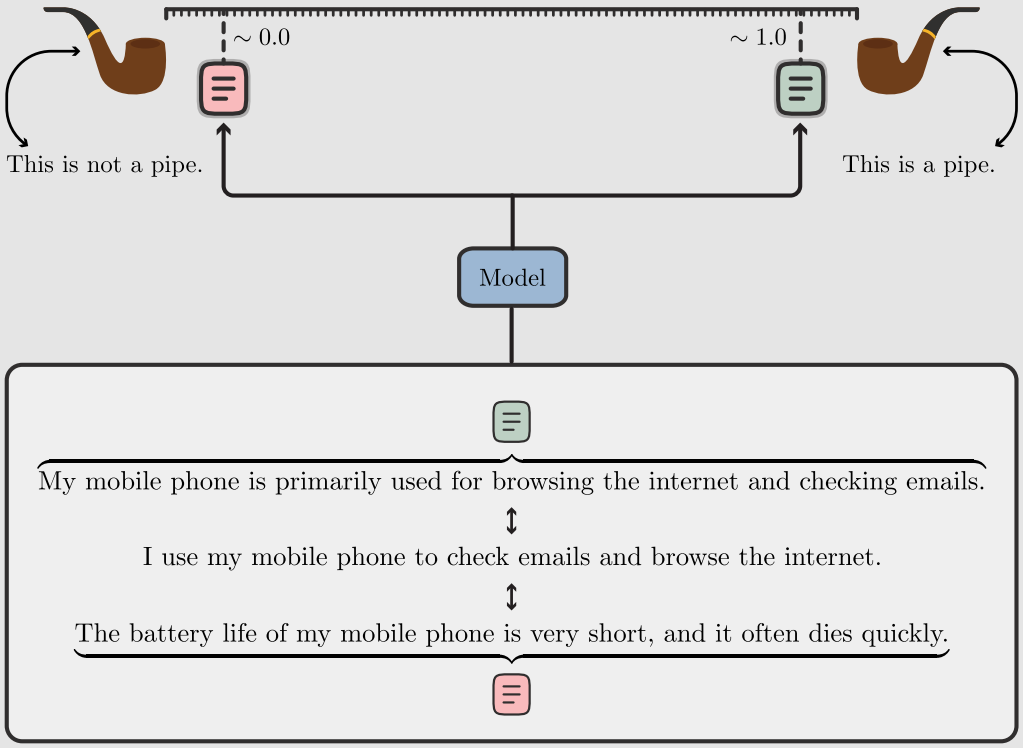

Figure 2.4: Example of a Semantic Textual Similarity Application - This figure illustrates two sets of sentences, each using the middle sentence as a reference. The top sentences set is displayed in green and the bottom pair of sentences in red. After processing through a regression model, each set of sentences receives a score, with the green sentence obtaining a high similarity score and the red sentence receiving a low one.

assessment of the model's true ability to generalize. The F_1-Score is commonly used as a standard metric. Assigning all instances to a class will result in a Recall value of one, but Precision is more difficult to improve; due to the inverse relationship between Precision and Recall, an increase in one leads to a decrease in the other. This is the reason why the F_1-Score is valuable, as it forces the enhancement of both metrics simultaneously. In domains like Information Retrieval (IR), where Recall is a fundamental metric, using a cut-off @k to retrain only the k highest scores returned is seen as a strategy to address Recall limitations.

When assessing models on multi-class or multi-label problems, it's common to use a metric for each category. However, various aggregation methods can be used to provide a comprehensive understanding across all categories. The macro aggregation averages the metric across all categories, ensuring each category is given equal importance regardless of dataset balance. Conversely, weighted average aggregation considers the category distribution, meaning categories appearing more frequently will have their metrics proportionally represented as in the dataset. Micro aggregations are another way to aggregate metrics, it can be done by summing all TP, FP, and FN values to compute the overall metric. Equations 2.6, 2.7 and 2.8

refer, respectively, to the weighted, micro and macro F_1-Scores.

$$F_{1_{weighted}} = \sum_{i=1}^{n} w_i \cdot \left(2 \cdot \frac{\text{Precision}_i \cdot \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \right) \tag{2.6}$$

Where w_i represents the weight of class i, Precision_i and Recall_i are the Precision and Recall for class i.

$$F_{1_{micro}} = 2 \cdot \frac{\text{Precision}_{micro} \cdot \text{Recall}_{micro}}{\text{Precision}_{micro} + \text{Recall}_{micro}} \tag{2.7}$$

Where Precision_{micro} and Recall_{micro} are the overall Precision and Recall calculated by aggregating the TP, FP, and FN across all classes.

$$F_{1_{macro}} = \frac{1}{n} \sum_{i=1}^{n} \left(2 \cdot \frac{\text{Precision}_i \cdot \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \right) \tag{2.8}$$

The micro, macro, and weighted F_1-Scores each comes with their own benefits and drawbacks when assessing multi-class classification models. $F_{1_{micro}}$ aggregates the contributions of all classes to derive the average metric, making it particularly advantageous for imbalanced datasets. By giving equal weight to each sample, it tends to favor the performance of majority classes, potentially masking the poor performance of minority classes. In contrast, $F_{1_{macro}}$ calculates the metric separately for each class and then finds the unweighted mean, treating all classes equally irrespective of their support. This makes it more sensitive to how well a model performs on minority classes, but it might not be an accurate reflection of overall accuracy in imbalanced datasets. $F_{1_{weighted}}$ strikes a middle ground between the two by computing the average F_1-Score weighted by the support of each class. This method accounts for class imbalance while providing insight into performance across all classes.

The selection of these metrics depends on the specific needs of the classification task and the importance of minority class performance in the given context. It should be noted that $F_{1_{micro}}$ is equivalent to the accuracy in multi-class classification with single-label samples. In situations where the dataset is perfectly balanced, which means that each class contains the same number of instances, all F_1 aggregations can be identical. In such balanced scenarios, the weights in $F_{1_{weighted}}$ are equal, which makes $F_{1_{weighted}}$ equivalent to $F_{1_{macro}}$. Consequently, the overall Precision and Recall would also be equal to both aggregated Precision and Recall, which means $F_{1_{macro}}$ would be equal to $F_{1_{micro}}$.

An alternative approach to assessing the performance of a multi-label problem is by considering the metric from an observation-wise perspective rather than a category-wise one. In this context, Exact Match Ratio (EMR) identifies the proportion of instances where the predicted labels match the actual labels exactly. EMR is especially advantageous in contexts where every label must be correctly predicted for the instance to be correct, such as in medical diagnostics or document classification. This metric is highly sensitive, as it demands complete accuracy; even a single incorrect label causes the entire instance to be classified as incorrect. The formula for computing EMR is given by:

$$\text{EMR} = \frac{1}{n}\sum_{i=1}^{n} I(y_i = \hat{y}_i) \tag{2.9}$$

where n is the total number of instances, y_i is the true set of labels for the i-th instance, $\hat{y}_i$ is the predicted set of labels for the i-th instance, and I is the indicator function that returns 1 if y_i equals $\hat{y}_i$, and 0 otherwise. Although EMR offers a precise and straightforward measure of model performance, its rigidity often necessitates the use of additional metrics to capture partial correctness and deliver a more detailed assessment.

Root Mean Squared Error (RMSE) is a commonly used and straightforward metric for assessing regression model performance for tasks such as STS. It quantifies the average size of the errors between predicted and actual values, giving a clear sense of the model's effectiveness. RMSE is especially intuitive because it is presented in the same units as the target variable, simplifying interpretation. A lower RMSE value suggests a better model fit, indicating closer alignment between predicted and actual values. Furthermore, RMSE places more emphasis on larger errors by squaring the residuals, which helps to spotlight models that produce significant prediction errors. This susceptibility to large errors makes RMSE an excellent metric for evaluating model accuracy, particularly in scenarios where substantial deviations are highly undesirable.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{2.10}$$

where y_i are the actual values, $\hat{y}_i$ are the predicted values, and n is the number of observations. Although RMSE is a straightforward metric for evaluating regression, it diminishes with the error magnitude and does not scale with other task metrics for aggregation. Conversely, the R^2 value, which ranges from 0 to 1, indicates the proportion of variance in the dependent variable explained by the independent variables. Higher R^2 values signify better model fits. Furthermore, R^2 is easy to understand and can be combined with other metrics, such as the F_1 score, making it a useful tool for model comparison and aggregation in performance evaluation.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{2.11}$$

2.1.1.6 Cross-Validation

Cross-validation is a method that evaluates the generalization performance of a model on a dataset that is separated from the one used for training. It involves the splitting of the dataset into various subsets (see Figure 2.5), training the model on one part of the data, and subsequently evaluating it on the remaining unseen data. In k-fold cross-validation, the training procedure is iterated k times, each time on different splits. This guarantees the uniformity of the model performance over various data partitions, thus mitigating overfitting and offering a more robust assessment of the model's metrics. Overfitting occurs when a model learns the

noise in the $\text{set}_{training}$ instead of the true underlying patterns, leading to poor generalization of new and unseen data. This technique facilitates model evaluation and comparison on a larger dataset indirectly, but it is time-consuming because the model needs to be trained for each fold.

Figure 2.5 shows the training process for a given fold after splitting the dataset into k-folds. First, the $\text{set}_{training}$ is divided into two different subsets, the $\text{subset}_{training}$ that is used to fit the model with a gradient-based optimizer and the set_{dev} that is iteratively predicted by the model, e.g. each epoch or n steps. Applying Hyperparameter Optimization (HPO), the model is fine-tuned with the optimal hyperparameters identified for a specific fold/metric/set_{dev} and subsequently utilized to predict the set_{test}.

This process is repeated for each fold, allowing the accumulation of k predicted splits. Subsequently, the model performance metrics are computed on all these predictions, which encompass the whole dataset. This approach allows for a more robust model assessment, particularly in situations where the size of the dataset is limited.

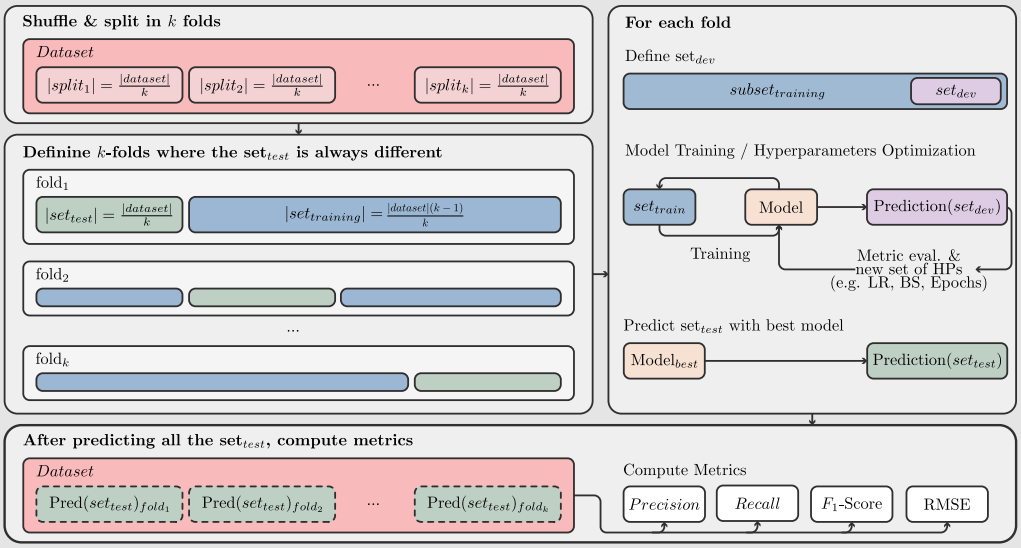

Figure 2.5: Cross-Validation Workflow - First, the dataset is divided into k equally sized segments, and each fold involves assigning the set_{test} to a different split each time. Subsequently, in each fold, a model is iteratively trained and assessed using a predefined $\text{subset}_{training}$ and a set_{dev}. Based on a predetermined metric, different sets of hyperparameters are tested, and the model that achieves the highest score is employed to predict the current fold's set_{test}. Finally, once all folds have been processed, metrics are calculated for the entire dataset which has been covered by each fold's set_{test}.

2.1.1.7 Hyperparameters Optimization

Before performing tasks such as classification, NER, or QA, models need to undergo a training phase to align with the data, usually using gradient-based optimizers. The field of optimization covers numerous methods that will not be discussed in

this thesis. The Adam optimizer (Kingma and Ba, 2017) will be the go-to optimizer as it is commonly used in various NLP applications, including the fine-tuning of Language Model (LM), a topic that is covered in Section 2.4.2.3.

During the training phase, the model weights are updated, after defining a loss function, an optimizer, and a set of hyperparameters which may include batch size (the amount of data used per gradient update), the number of epochs (how many times the training data is presented to the model), the learning rate and more. Optimizing these hyperparameters greatly impacts model performance, hence it is recommended to employ HPO.

Grid search is arguably the most straightforward HPO method because it requires assessing each possible combination of hyperparameters within a predefined range and granularity. However, this method can quickly become impractical and potentially counterproductive, especially when early trials show a decrease in performance upon tuning a particular hyperparameter. For instance, if we choose to experiment with batch sizes of $[16, 32, 64, 128]$, noting that performance is high at 16 but drops with 32 and 64, it would seem inefficient to test a batch size of 128 every time we explore new hyperparameter settings.

When looking at more sophisticated HPO methods, Sequential Model-Based Optimization (SMBO) iteratively builds models to estimate the performance of hyperparameters using past measurements and then selects new hyperparameters to evaluate based on these models. Different models can then be trained with varying sets of hyperparameters, and the decision to continue training or not can be based on a metric from the set_{dev}, which comprises data not used in model training. This is exactly the kind of HPO method that can be used in conjunction with cross-validation as illustrated in Figure 2.5.

2.1.1.8 Statistical Significance

After training all models, a direct comparison of their performances does not definitively indicate the best model. Instead, statistical tests must be conducted to determine if there are meaningful differences between the models. The level at which these statistical tests are conducted is crucial; they can either compare prediction differences between models to discern if the differences are random or analyze various metrics to see if they are statistically similar or differ with statistical significance. In the latter situation, the choice of the test depends on the distribution of the metrics and the number of models being compared. Table 2.1 shows the appropriate tests for typical scenarios.

$\mathbf{N}_{Models}$	**Parametric Tests**	**Non-parametric Tests**
=2	t-test	Wilcoxon
>2	ANOVA → Tukey	Friedman → Nemenyi

Table 2.1: Statistical Tests for Comparing Models Metrics - This table displays the type of test based on the number of models being compared and whether the data adheres to a specific distribution when dealing with more than two models, a two-folds conditional test is usually performed to get each 1-vs-1 comparisons.

The primary consideration when choosing a test is the number of models being compared. Performing multiple pairwise tests instead of a single test for more

than two groups increases the likelihood of a Type I error, which is the incorrect rejection of the hypothesis that the models are identical (H_0). Tests intended to compare more than two groups account for variability both within and between groups. If the overall result suggests that at least one group significantly differs from others, a follow-up test can provide detailed pairwise comparisons, adjusted for Type I error.

On a different note, parametric tests usually offer higher statistical power than non-parametric tests when their assumptions are met, as they more efficiently detect real effects or differences by using more detailed data information, such as actual values instead of ranks. However, ensuring that data meets specific assumptions, such as normality and homoskedasticity for Analysis Of Variance (ANOVA), is critical before conducting a parametric test. In practice, these assumptions are often unmet in model metrics, leading to the use of non-parametric tests in machine learning model comparisons (Demšar, 2006). Thus, when evaluating multiple models simultaneously, it is common to apply a Friedman test (Friedman, 1937) followed by a Nemenyi test. Obtaining statistical significance is difficult with a small dataset; therefore, it is crucial to test models on various datasets for improved comparison.

When evaluating the performance of various models, it is essential for the metric distributions to be independent. Although this approach is widely used in the field, running the same experiment repeatedly on the same dataset is not viable for comparing model metrics. The abstraction level is what provides meaning to the testing; in fact, for a given dataset, one can merely demonstrate that models behave differently and, at most, identify a model as superior for that specific dataset. To claim that a model excels at a particular task, such as classification, it should be evaluated on multiple datasets to assess differences in metrics.

To assess variations in model predictions, the McNemar test is frequently utilized for binary tasks. Despite being essentially a pairwise test, it is often employed to compare multiple groups, combined with a Bonferroni (Dunn, 1961) correction to handle the numerous pairwise comparisons required (Dietterich, 1998). Although this adjustment is often deemed overly conservative, it is widely used for its simplicity.

2.1.2 General Language Understanding Evaluation: a Benchmark for Natural Language Understanding

Typical NLP services often consist of sequences of tasks that may include document retrieval/classification and re-ranking, NER, Machine Translation (MT), QA, and Relation Extraction (RE). Researchers are continuously working to enhance the State-of-the-Art (SOTA) techniques for each NLP task by experimenting with various datasets in pursuit of improved outcomes.

An example of a QA dataset is the famous Stanford Question Answering Dataset (SQuAD) (Rajpurkar et al., 2016) that contains more than 100 thousands (k) questions related to a set of Wikipedia articles where the answer to each question can be found. Its published leaderboard can be found online[3] where the current best model gets better results than their evaluated human performance.

[3] `https://rajpurkar.github.io/SQuAD-explorer/`

Following the emergence of LM publications, it became evident that assessing the performance of a NLU model required the grouping of various downstream tasks to create a metric for global comparisons. (Wang et al., 2018) introduced the General Language Understanding Evaluation (GLUE) benchmark to facilitate such comparisons. GLUE comprises nine tasks sourced from diverse datasets with varying sizes, scopes, and complexities. The public leaderboard, including its upgraded version SuperGLUE that features more demanding tasks, is available online[4].

The datasets used for these benchmarks are commonly referred to as general-domain or non-Domain-Specific datasets since they are not specific to any particular field and do not require Domain-Specific (DS) knowledge to achieve optimal performance. The following section will introduce key concepts essential to understanding the application of NLP in the life sciences domain.

2.2 Natural Language Processing in Life Sciences

In the field of bioinformatics, maintaining and organizing scientific literature is crucial to support researchers in navigating through a vast and constantly changing collection of papers. Therefore, specialized NLP tools play a vital role in handling the influx of new publications. For instance, the UniProt Knowledgebase (UniProtKB) (Consortium, 2022) serves as a protein database where certain entries are annotated by automated systems. UPCLASS (Teodoro et al., 2020) is one of them. It is a classifier that assigns categories such as function, interaction, and expression to a publication related to a specific protein. These types of models enable scientists to manage the growing volume of research data and concentrate on extracting the pertinent information they seek about a particular protein.

Another illustration of a tool based on a classification task can be found in (Mottin et al., 2023), which is developed to facilitate the automated categorization of the four Response Evaluation Criteria in Solid Tumors (RECIST) scales using radiology reports. It also investigates the impact of language variations and institutional characteristics of Swiss teaching hospitals on the accuracy of classification in French and German languages.

Throughout the COVID-19 pandemic, there was an unprecedented surge in the volume of digital data from various fields, resulting in over 140k papers being produced in just a few weeks for the COVID-19 Open Research Dataset (CORD-19) (Wang et al., 2020). This rapid increase made it challenging for medical professionals and epidemiologists to expand the epidemiological curation process (Chen et al., 2020). The COVID-19 Open Access Project (COAP) is a living evidence of COVID-19 (Project, 2020), now leveraging LM to perform reference indexing and assist with epidemiological curation and review. These automated categorizations enable curators to stay abreast of the escalating volume of COVID-19-related publications (Knafou et al., 2023).

There are numerous competitions available in the field of life sciences and our research group has participated in several of them involving NER. An example is the task proposed at the Workshop on Noisy User-generated Text (W-NUT) 2020 event, which focuses on Wet Lab protocols, referring to chemistry or biology experiments

[4] `https://gluebenchmark.com/`

described in natural language. The objective of addressing this challenge is to streamline the automation of experimental procedures using robots. The highest F_1 score in the competition is achieved using a combination of models (Knafou et al., 2020). This approach is typically more robust compared to a standalone model (Naderi et al., 2021). Two additional instances of competitive NER tasks were (1) the third competition of DÉfi Fouille de Textes (DEFT) 2020, which aimed to recognize particular details in 12 categories within a set of clinical cases written in French (Copara et al., 2020a) and (2) the first task of Cheminformatics Elsevier Melbourne University (ChEMU) 2020, which concentrated on the identification and categorization of chemical compounds according to their functions in a chemical reaction (Copara et al., 2020b).

Every year, the renowned Text REtrieval Conference (TREC) organizes various tracks where participants can showcase their methods and assess their performance against others. In the realm of bioinformatics, a range of tracks have been available from 2003 through the most recent campaign in 2023. These include TREC Genomics (2003-2007), which targeted genomics researchers in search of pertinent biomedical literature; TREC Medical Records (2011-2012), which focused on retrieving patient cohorts from Electronic Health Records (EHRs); Clinical Decision Support (2014-2016), which catered to clinicians seeking evidence-based literature to aid in diagnosis, treatment, and testing decisions; Precision Medicine (2017-2020), which addressed oncologists in search of evidence-based treatment literature and clinical trials; Clinical Trials (2021-2023), an ongoing track that aims to match patients with suitable clinical trials; new tasks are still being introduced in the campaign of this year.

During the pandemic, TREC-COVID (Voorhees et al., 2020), a IR track was introduced to create a test dataset for pandemics by carrying out several rounds using CORD-19 as the document collection and a series of biomedical questions as the topics. Unlike traditional TREC tracks, this specific task involved condensed rounds where systems were allowed to incorporate relevant feedback from previous rounds. In order to achieve optimal performance measures, different teams have developed a two-step process involving a conventional IR system for document retrieval, followed by a LM for re-ranking (Roberts et al., 2021; Teodoro et al., 2021).

During the eleventh edition (Nentidis et al., 2023) of the BioASQ challenge, held within the framework of Conference and Labs of the Evaluation Forum (CLEF) 2023, two biomedical QA tasks were featured. The first task consisted of two stages. In phase A, participants were required to identify and submit relevant content from specified sources, specifically PubMed/MEDLINE abstracts and extracted snippets. During phase B, the participants' systems were required to give accurate responses in the form of entity names or short sentences, along with optimal answers presented as natural language summaries of the information requested. The second task, named Synergy, seeks to enhance collaboration between automated QA systems and biomedical experts. These systems offer pertinent information and responses to experts who have raised unresolved queries. The experts evaluate these answers and share their feedback with the systems. Subsequently, the systems use this feedback to offer more relevant information, incorporating recent data that may have become accessible in the meantime, and to provide enhanced responses to the experts. In this campaign, the optimal strategy for the first task (Almeida et al.,

2023) entails a two-phase retrieval method to address phase A, using the Anserini BM25 (Yang et al., 2017) as the primary stage, followed by the implementation of a re-ranking model based on a LM. In phase B, their strategy includes incorporating the article from phase A into a LM model tuned on instruction e.g., Large Language Model Meta AI (LLaMA) (Touvron et al., 2023). This results in generating answers conditioned on the articles collected in phase A.

2.2.1 Biomedical Language Understanding & Reasoning Benchmark

In the last section, we briefly discussed the importance of having a NLU benchmark by introducing GLUE. In the field of biomedicine, several models, such as BioBERT (Lee et al., 2019), have shown how a Pre-trained Language Model (PLM) trained on a DS corpus can outperform a generic model on a DS downstream task. In (Gu et al., 2021), the researchers evaluated their model using Biomedical Language Understanding & Reasoning Benchmark (BLURB), a benchmark they introduced along with a PLM trained from scratch on a biomedical corpus they call PubMedBERT.

BLURB is designed to evaluate the effectiveness of NLU models within the biomedical domain. Inspired by Biomedical Language Understanding Evaluation (BLUE) (Peng et al., 2019), it enhances the existing benchmark by addressing certain limitations, incorporating a QA component, and increasing the proportion of biomedical/clinical datasets. The BLURB framework consists of five NER tasks, a module for extracting evidence-based medical information, three RE tasks, an STS, a document classification task, and two QA tasks. The leaderboard for BLURB is publicly accessible online[5].

2.3 Introduction to Modern Natural Language Processing Approaches

This section introduces some model architectures by covering the process from the input to the prediction, as understanding these basics is crucial for comparing models in the subsequent sections. First, text encoding will be covered from the simplest whitespace separation method to the subword segmentation algorithms that are currently deployed in the latest models. Next, the transformer architecture, on which most current SOTA models are built, will be introduced by following the development of Recurrent Neural Networks (RNNs). This section seeks to investigate the methods for comprehending models such as BERT, which will be discussed in Section 2.4.2, along with machine translation models like Sequence-to-Sequence (Seq2Seq) (Sutskever et al., 2014) or the more recent M2M-100 introduced in Section 2.5.4.

2.3.1 Raw Text Tokenization

Data analysis typically employs models that require a vector or matrix as an input. Consequently, when working with unprocessed text, it is necessary to perform a

[5] `http://aka.ms/BLURB`

data transformation before proceeding to the data analysis stage. These preliminary procedures generally seek to represent the input with minimal interference and substitute it with numerical values. The process of transforming raw text into its final form may include several operations such as converting to lowercase, removing stop-words, stemming, and spell-checking. In this context, our focus will be on tokenization, the process of segmenting sentences into tokens, which typically serves as the final preparatory step before supplying data to a model.

2.3.1.1 Whitespace Separation

The conventional approach to segmenting text involves using spaces as a delimiter for words. When a corpus is segmented on the basis of these spaces, it is converted into a collection of individual words. Given the potentially high number of unique words in a corpus, it is common to establish a threshold for the vocabulary size and preserve the most frequently occurring words in the original corpus. Subsequently, a word dictionary is generated to associate each word with a distinct identifier ranging from 1 to the specified vocabulary size. A fundamental method for representing a textual sequence in vector format is to initialize a vector of the vocabulary size with 0's and then add 1's to the indices corresponding to the words present in the sequence.

Figure 2.6 illustrates this kind of representation with a preset dictionary size of 10 words. This example shows a few particularities of such approaches. First, rare words do not appear in the dictionary and get ignored in the vector representation, for instance, "superduper" and "encoded" are not in the initial dictionary and end up being discarded. Misspelled words suffer from the same effect and are not taken into account, even if their correct version is in the dictionary. There is an obvious trade-off between a massive vocabulary size that creates large sparse vectors and a small vocabulary size that tends to ignore words that give sense to a sequence. Besides these flaws, this is by far the fastest way to represent text. Combined with Term Frequency-Inverse Document Frequency (TF-IDF) or Word2Vec (W2V), it can sometimes be the best speed/performance trade-off.

2.3.1.2 Subword segmentation algorithm

Using the whitespace separation method implies having a large vocabulary size to compensate for Out-of-Vocabulary (OOV) issues, that is, words that will be ignored from the model dictionary. Subword segmentation algorithms allow us to represent all sequences possible by encoding a sequence with only parts of words.

Byte Pair Encoding (BPE) (Gage, 1994) is a data compression algorithm that processes a complete sequence by identifying the most common pair of characters and substituting it for an unused character. The algorithm is trained by repeating this process until the vocabulary reaches a specific predetermined size. The initial vocabulary includes all the individual characters present in the text. Initially used as a tokenizer in (Sennrich et al., 2016), BPE helped with the handling of OOV terms, which arise from infrequent or incorrectly spelled words.

In contrast to the approach of BPE which repeatedly selects the most common pair of characters to build its vocabulary, the Unigram Language Model (Kudo, 2018) relies on the probability of occurrence to determine the vocabulary from a given corpus. By treating each subword as independent, this technique enables the

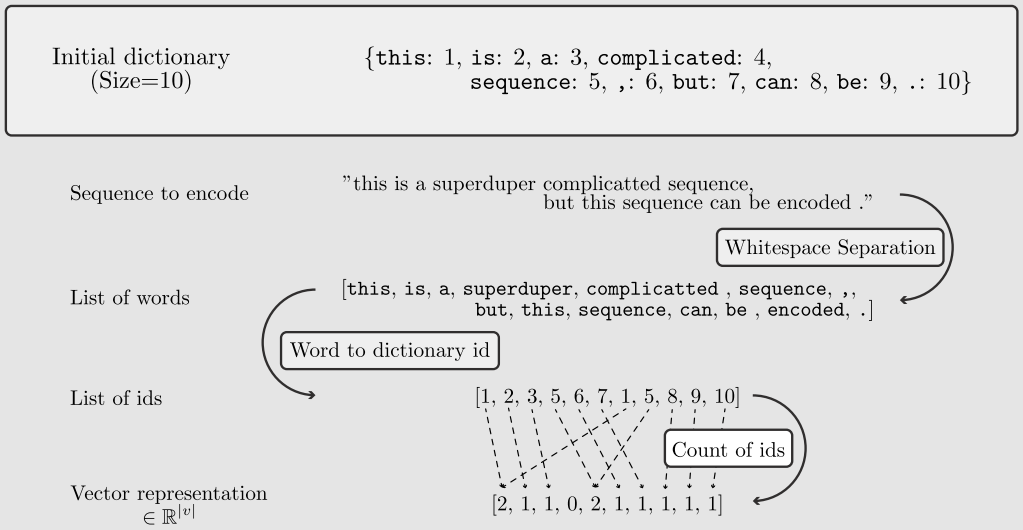

Figure 2.6: Count-Based Vector Representation - Illustration of the process from the sequence to the count-based vector representation based on preset dictionary

optimal segmentation of a sentence based on likelihood. Furthermore, by storing the probability for each subword, this strategy supports subword sampling, a technique introduced in the aforementioned paper.

SentencePiece[6] (Kudo and Richardson, 2018) incorporates both methods within a single package, enhancing speed while operating at sentence level. This is particularly important in languages where words are not explicitly separated by spaces. Previously, subword segmentation was performed based on words to construct their vocabulary. However, SentencePiece does not depend on whitespace separation or treat words as separate units; instead, it views whitespaces as ordinary characters.

Figure 2.7 demonstrates the iterative process through which the BPE model constructs its vocabulary by identifying the pair of characters that occurs the most frequently. Initially, at iteration 0, the vocabulary comprises all the individual characters in the training sequence. Subsequently, at iteration 1, the pair of characters ['e', 'n'] is identified as the most common, leading to the inclusion of the subword 'en' in the vocabulary. This process continues iteratively, with the model replacing the pair of most frequent tokens at each step. By iteration 4, the model identifies ['en', 'c'] as the predominant pair, resulting in the emergence of tokens consisting of three characters. After 30 iterations, the advantages of subword segmentation become apparent. For instance, the word 'superduper' is segmented into ['_s', 'uper', 'd', 'uper'], enabling the retention of meaningful word components even in the case of misspellings. By iteration 44, all words in the sequence are included in the dictionary, facilitating word-by-word tokenization. It is important to highlight that even at iteration 0, the model could already encode the entire sequence using individual characters.

[6] https://github.com/google/sentencepiece

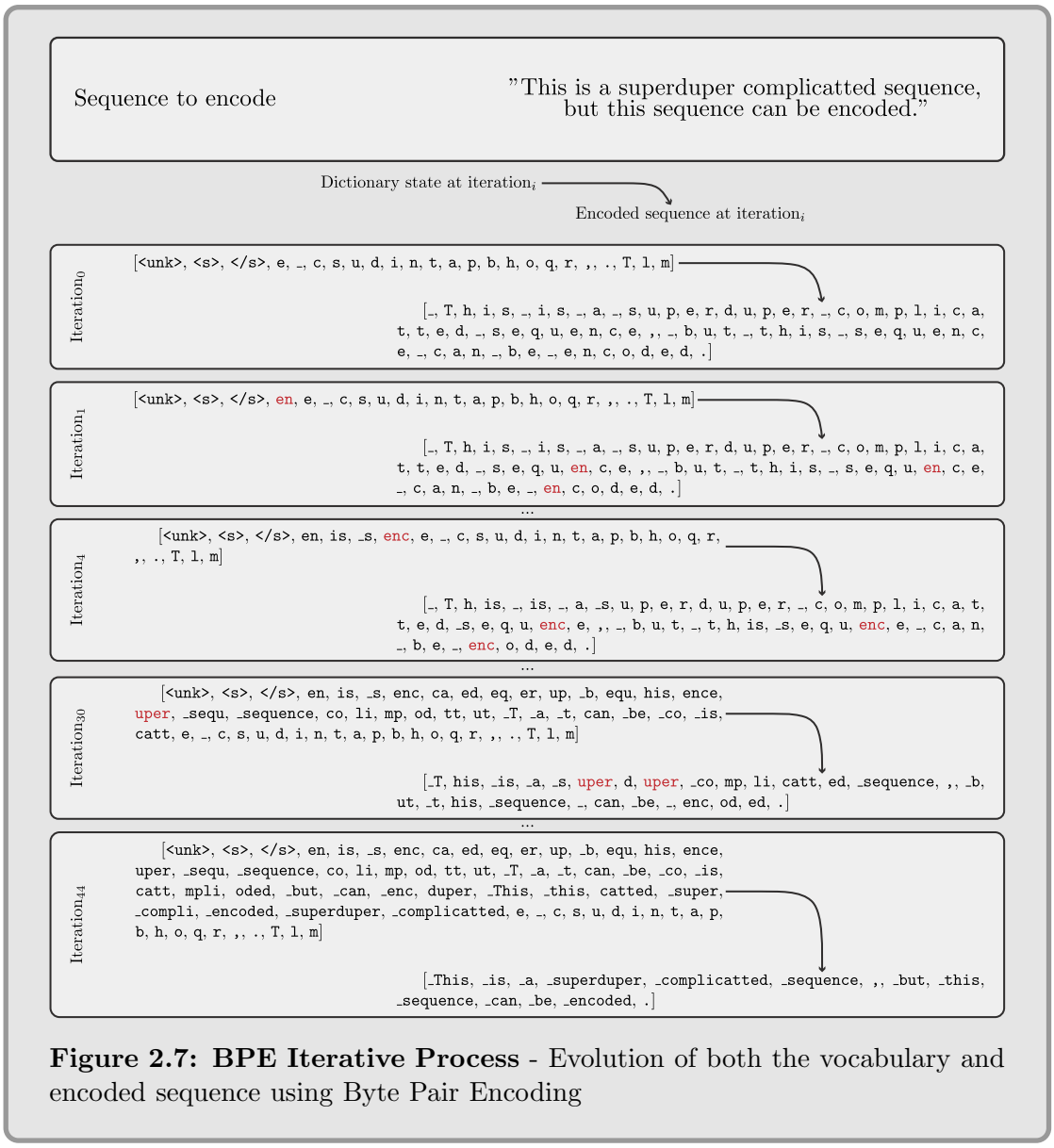

Figure 2.7: BPE Iterative Process - Evolution of both the vocabulary and encoded sequence using Byte Pair Encoding

2.3.2 The Fundamentals of Transformer

This section will introduce the Transformer model (Vaswani et al., 2017), which will facilitate the discussion of SOTA models in LM and MT in Sections 2.4 and 2.5, respectively, including BERT and other Transformer-based models.

2.3.2.1 Recurrent Neural Network

Prior to the introduction of attention mechanisms, which is a fundamental concept in Transformers, SOTA performances were achieved primarily using RNNs. In all Natural Language Generation (NLG) tasks, such as translating or summarizing, recent models are composed of an encoder that reads and encodes the input sentence, serving as a NLU module, and a decoder that predicts translated sentences one word at a time.

RNNs exhibit recurrence because the computation of each word vector depends on the hidden state of the previous word, as depicted in Figure 2.8. Although RNNs

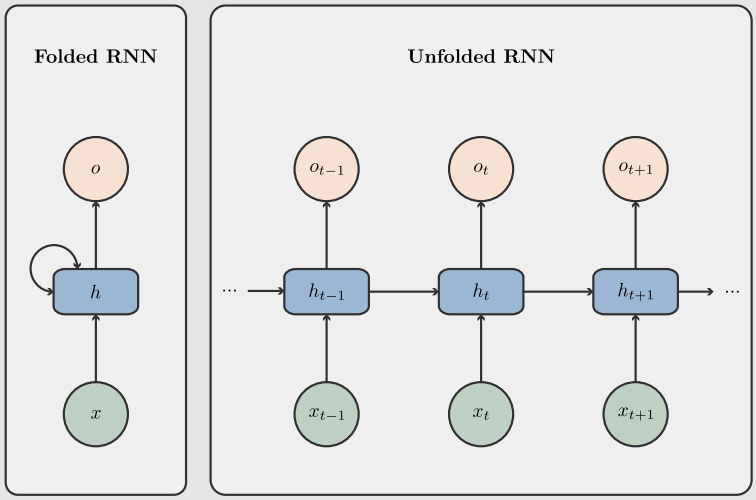

Figure 2.8: Folded and Unfolded RNN Diagram - Illustration of the model's recurrence, h_{t-1} is always required to compute h_t.

were leading the translation leaderboards, they tend to struggle with long sentences because the decoder lacks awareness of the part of the sentence being translated (Cho et al., 2014). To address this issue, some strategies involve presenting the input sentence in reverse order, leading to better performance (Sutskever et al., 2014). The assumption is that giving the sentence backward simplifies the optimization problem by introducing many short-term dependencies. Fundamentally, when providing a sentence to the encoder, the first word to be translated is typically placed at the start of the sentence, necessitating a reliance on long-term memory.

The process is better illustrated in Figure 2.9 that shows an encoder-decoder with an attention mechanism. Without the attention mechanism, when translating the final words of a long sentence, the decoder, which should translate the first words at the start, will be provided with (1) the encoder latent representation, which has flaws due to the long-term dependencies, and (2), words predicted in earlier iterations, which tend to be wrong as they were predicted using a flawed latent representation. In the scenario of giving as input a backward sentence, the decoder will be provided with (1) the encoder latent representation with short-term dependencies, as last words would be the first to be predicted, and (2) when predicting the last words, words predicted in earlier iterations, specifically, o_i up to o_{t-1} as illustrated in Figure 2.9.

RNNs commonly employ cells to compute h_t and o_t, representing the hidden and current states of the cell, respectively. These cells function as components with matrix multiplication and activation functions designed to mitigate the problem of vanishing gradient encountered by conventional RNNs. The most popular cell types are Long Short Term Memory (LSTM) and Gated Recurrent Units (GRU), both of which receive the current word embedding (i.e., word vector representation) and the previous cell's hidden state as inputs. Figure 2.8 illustrates how a RNN cell uses x_t (the current word embedding) and h_{t-1} (the previous hidden state) to compute h_t (the hidden state at time t) and o_t (the current cell state), which

can be forwarded to a classifier or the subsequent layer. In the original Seq2Seq architecture introduced in (Sutskever et al., 2014), the final hidden state h_{t+n} is transferred to a decoder, a RNN responsible for sequentially generating words. For each word prediction, the decoder incorporates the previously predicted words until it predicts a token <STOP>. Similarly, the decoder starts with a token <GO> as the initial input word.

2.3.2.2 Attention Mechanism

Enhanced performance on long sentences is achieved by incorporating attention mechanisms into Seq2Seq models (Bahdanau et al., 2014). The fundamental concept involves providing the decoder with a weighted sum of all the hidden states from the encoder, emphasizing the words relevant to the model's current prediction. Specifically, a probability distribution of the source words is computed for each predicted word, and a combination of this distribution and the source word states is fed to the decoder. Figure 2.9 illustrates how attention allows signals to take shortcuts, thus avoiding long-term dependencies that may arise when translating the first words into long sentences.

During the decoding process, such as in translation or summarization tasks, when the model is predicting a word in the middle of a sequence, it will assign higher importance to the words linked to that same word in the source sequence compared to others (see Figure 2.9). This enables the model to focus on the most relevant words. In the context of translation, prior to the introduction of the attention mechanism, the decoder would consider the final state of the word as the overall representation of the sequence, which was inadequate and led to loss of information, particularly with lengthy sequences. As illustrated in Figure 2.9, the model remains similar to a standard Seq2Seq model; however, the decoder now receives a weighted combination of representations of all words rather than relying solely on the last hidden state h^e_{t+n} in this scenario, thus addressing the challenges posed by lengthy sentences.

2.3.2.3 Self-Attention

The introduction of self-attention (Lin et al., 2017; Paulus et al., 2017; Parikh et al., 2016; Cheng et al., 2016) expands the application of attention mechanisms to models that do not have decoders. Now, large documents can be classified using RNNs combined with self-attention without encountering the problem of vanishing signals. In (Cheng et al., 2016), the concept is referred to as 'intra-attention' as the main idea is to compute a word's representation using other words' hidden states. In their version based on a LSTM model, each word is allowed to attend to its preceding words.

There is no standard attention or self-attention, as various publications simultaneously introduce their own variations. The one version that is now considered as a standard is depicted in figure 2.10 which can be viewed as a soft averaging lookup as it consists of comparing a query vector (q) to a set of key vectors (k) and return a weighted average of each key's values vectors (v).

In other words, each token has a set of $q/k/v$ vectors and in order to perform attention on a given word, one needs to compute a similarity score between the

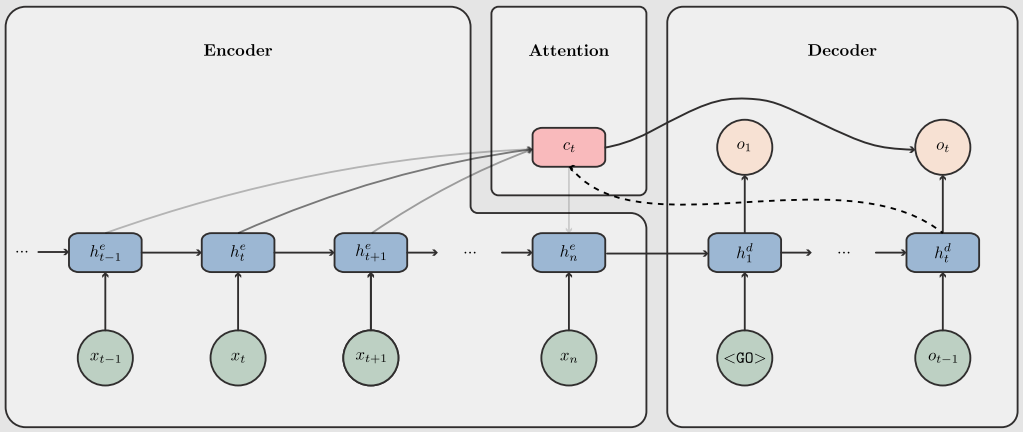

Figure 2.9: Attention Mechanism Diagram - In this representation of an encoder-decoder system, we can see that without the attention, the model will predict the t^{th} word relying solely on a fixed source representation. Attention allows to assemble a context vector c_t at each prediction step allowing the model to focus on the source sentence part that matters the most when predicting the t^{th} word.

word's query and all the sequence's key vectors and scale it with a softmax function (see Equation 2.1):

$$\alpha_{ij} = \frac{exp(q_i^\intercal k_j)}{\sum_{j'} exp(q_i^\intercal k_{j'})} \tag{2.12}$$

Once all the α are computed for a given token, the attention output will simply be equal to the weighted sum of the key's value vectors:

$$o_i = \sum_j \alpha_{ij} v_j \tag{2.13}$$

Figure 2.10 shows an intuitive way of picturing self-attention, first, q_i is compared to other k vectors, resulting in different shade of α according to the similarity of q and k, then o_i is calculated as the sum of the v vectors weighted by α's intensity. Although this way of computing word representation is parallelizable, it doesn't take word order into account.

2.3.2.4 Transformers

Shortly after the introduction of models that incorporate self-attention, the idea of getting rid of the recurrent component of the model was investigated in the Transformer architecture (Vaswani et al., 2017). Transformer is a MT model consisting of an encoder that receives the source sequence and a decoder that predicts the target sequence word-by-word. Similar to the Seq2Seq framework, the decoder in the Transformer takes its own previous predictions as input each time a word is being predicted. Through the use of self-attention, the model can concurrently compute contextual word representations for all input sequences. In order to solve the order problem discussed in Section 2.3.2.3, a position embedding

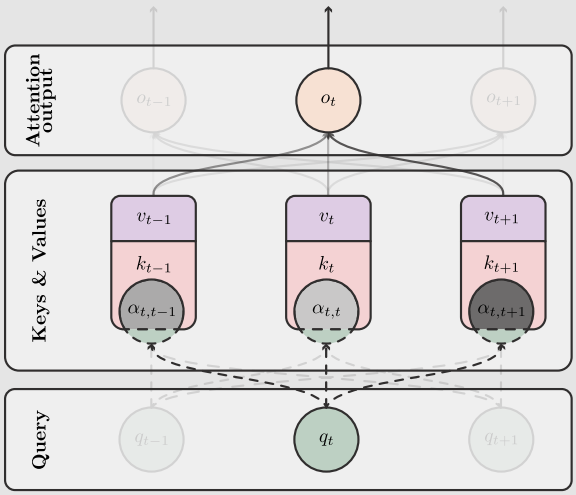

Figure 2.10: Self-Attention Diagram - Self-Attention computes all the α as a similarity score between a token query and all the sequence's key vectors. Finally, a sum of the values vectors (v) is computed using the α's intensities.

vector is added to the word representation before going through the self-attention layer. Trained on 8 Graphics Processing Units (GPUs) for approximately 8 hours (64 GPU hours) for the base configuration, the model surpasses all prior models or ensembles, while requiring only a fraction of computational resources.

Such a high-performance level is achieved by stacking up multiple attention blocks as illustrated in Figure 2.11. Each self-attention component is known as an attention head. Following the computation of the multi-head attention projection (size=h), the model combines all attention heads and provides a linear combination as input to the subsequent layer. Each attention head consists of three trainable matrices, denoted K, V, and Q, representing the key, value, and query matrices, respectively. The linear combination of the query and key undergoes normalization and is then passed through a softmax layer (see Equation 2.1), which is subsequently multiplied by the value matrix. Since the output maintains the same structure as the input, all layers can be stacked sequentially. In their publication, the authors stack a total of $N = 6$ layers before transmitting the final output representation to the decoder.

By using self-attention, the model's recurrence is solely based on the number of layers. Specifically, to compute layer t, layer $t - 1$ must be computed first. Consequently, the entire sequence can now be parallelized, provided that it fits within the available memory. Figure 2.11 illustrates the functioning of the encoder, which comprises N modules with h attention heads. An almost similar decoder is then deployed. As in Figure 2.9, the decoder takes as input the encoder output as well as the previous word predictions. Although this seems to mirror the Seq2Seq framework we discussed in Section 2.3.2.1, there is no recurrence anymore through the sequence computation. The model's enhanced speed is attributed to its parallel processing capability, enabling it to be trained on larger datasets within the same

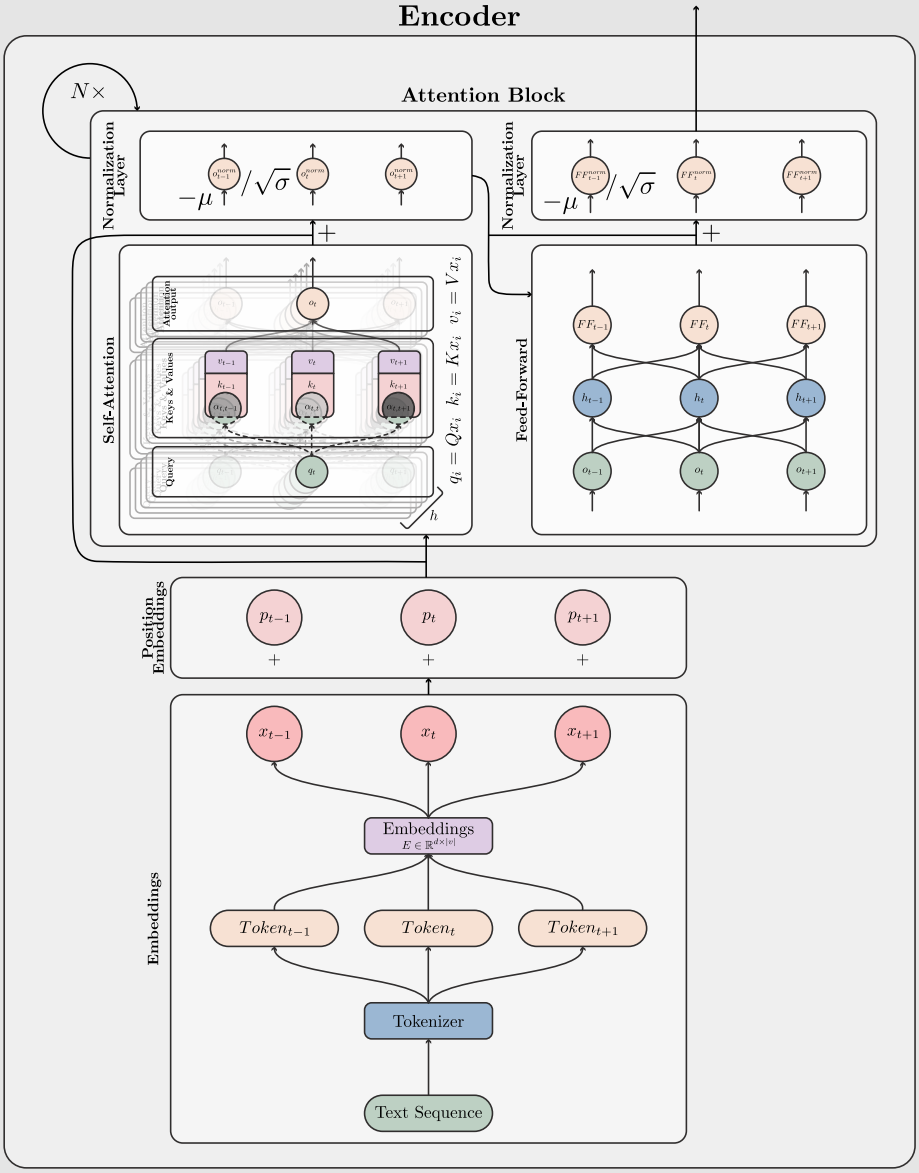

Figure 2.11: Transformer's Encoder Architecture - Text is first tokenized and each token is represented by a vector x_i and gets a position embeddings added to it. Using the matrix parameters Q, K, and V, each token gets a query, key, and vector representation, respectively. Then, each token is represented using $h-$ attention heads. Before getting normalized, the residual connection is added to the self-attention output. Finally, a feed-forward layer is also normalized after getting the residual connection. The attention block is repeated $N\times$ before being sent to the decoder.

fixed timeframe.

2.4 Language Models

This section will outline the progression of LMs leading to Bidirectional Encoder Representations from Transformers (BERT). Subsequently, the improved performance of this novel model in a wide range of NLP tasks will be discussed, followed by an exploration of different BERT variants.

2.4.1 Prelude to Modern Language Models

The concept of LM was initially introduced in the field of speech recognition (Bahl et al., 1983), followed by its application in MT (Brown et al., 1990) and automatic spell checking (Mays et al., 1991). During this period, the primary purpose of LMs was to determine the most likely sentence based on the output of another model. For example, in MT, once a sentence was translated word-by-word, a LM would compute the probabilities of one word given another. By assuming independence between words, the probability of the sentence would be the product of these conditional probabilities. Consequently, the system would generate the most probable sentence structure based on its translation.

Later, the use of NN enhances the performance of LMs (Bengio et al., 2003). During this period, reduction in perplexity was the sole metric employed for comparing different methods. In this setting, additional efforts were directed towards enhancing LMs. In particular, (Morin and Bengio, 2005; Mnih and Hinton, 2008) achieved superior results with a 258x acceleration in training time and a 193x improvement in testing time compared to the work by (Bengio et al., 2003).

In (Collobert and Weston, 2008), LM changes its function and is ultimately utilized for its ability to generate word representation to address various NLP tasks such as POS, chunking, NER, Semantic Role Labeling (SRL), and Semantically Related Words (SRW). Researchers are focusing on contrasting various methods of acquiring word representations and leveraging them to tackle specific tasks (Turian et al., 2010). As a result, the perplexity metric is no longer used to evaluate the performance of LMs.

This marks a shift in model comparison, as LM are now trained for word representation enhancement through the improvement of metrics across downstream tasks. In 2013, two NN-based methods to train word representations, namely Skip-Gram and Continuous Bag-of-Words (CBOW), were published along with their W2V trained model (Mikolov et al., 2013b,a). In both cases, the inputs and outputs are one-hot encoded indices, and their induced vector representations can be retrieved for further training by taking the first layers of the input. While CBOW focuses on predicting a central word based on a context window of surrounding words, Skip-Gram operates in reverse, predicting the surrounding words given a central word. Another word representation model, Global Vectors for Word Representation (GloVe), was introduced a year later, outperforming Word2Vec in various tasks (Pennington et al., 2014).

Although these new approaches preserve both semantic and grammatical patterns, they are unable to model polysemy. This implies that the term 'bank' in the sentences 'The bank interest rate is low.' and 'The bank I sat on is red.' are assigned identical vector representations, despite their distinct meanings. By using a Bi-LSTM RNN, Embeddings from Language Models (ELMo) (Peters et al., 2018)

can generate word embeddings that encompass contextual information around them. ELMo enables the dynamic computation of word embeddings based on its context. For instance, the word embeddings of 'bank' in 'The bank I sat on is red.' should be closer to a suitable synonym like 'chair' than to 'bank' in 'The bank interest rate is low.', which pertains to a financial institution. ELMo improves all previous SOTA results in six different tasks. Despite its good results, it came out just after Transformer models (Vaswani et al., 2017) and researchers quickly took advantage of the highly parallelizable feature of the said model to publish a LM taking advantage of this new model architecture.

2.4.2 Bidirectional Encoder Representations from Transformers

Introduced by Google[7], BERT is a groundbreaking LM that takes advantage of the Transformers architecture (Vaswani et al., 2017) and the WordPiece tokenizer (Schuster and Nakajima, 2012) while also addressing polysemy. In the training process, BERT leverages the encoder of the recently published Transformer model to perform simultaneously Masked Language Model and Next Sentence Prediction tasks. Due to its highly parallelizable design, the model can undergo pre-training on an unprecedentedly large corpus before being fine-tuned for various downstream tasks. The following sections further elaborate on the unique characteristics of the model.

2.4.2.1 WordPiece Tokenizer

Before subword segmentation, a common approach to handling OOV was to substitute each missing word with a pre-trained OOV word representation, usually denoted as `<UNK>`. Some papers started to use subword segmentation algorithms in MT and speech recognition to deal with OOV words (Luong et al., 2015). The first notable paper to implement word segmentation was (Chitnis and DeNero, 2015) using Huffman codes, then BPE (Gage, 1994) was applied to other MT models (Sennrich et al., 2016; Wu et al., 2016; Britz et al., 2017; Vaswani et al., 2017) and finally WordPiece (Schuster and Nakajima, 2012) in the BERT paper.

The tokenization model used in BERT plays a crucial role in its success. Although ELMo has the capability to capture polysemy, as discussed earlier, it generates representations at the word level, thereby making it challenging to encode OOV or misspelled words. By breaking words down into smaller units, WordPiece (Schuster and Nakajima, 2012) effectively handles misspellings and uncommon words. This approach is a variation of BPE, a tokenization technique introduced in Section 2.3.1.2 that computes pair scores differently. Unlike BPE, which uses the frequency of the pair as a score, WordPiece divides the pair frequency by the product of the frequency of each component of the pair. As the vocabulary size of the tokenizer increases, so does the number of parameters. With a vocabulary size of 30k tokens, the models require 110 millions (M) and 340M parameters for $\text{BERT}_{\text{base}}$ and $\text{BERT}_{\text{large}}$, respectively. For instance, if the vocabulary expands to 50k, the models would need 125M and 55M parameters for the base and large versions, respectively.

[7] `https://github.com/google-research/bert`

2.4.2.2 Pre-Training

The pre-training of BERT involves two distinct tasks: Masked Language Model (MLM) and Next Sentence Prediction (NSP). These tasks are executed together, with the former concentrating on the token level (using the WordPiece unit) and the latter on the sentence level. The pre-training stage is extensive and expensive, as it establishes the model's parameters on a vast unsupervised dataset, which is then leveraged to create generalized representations for various tasks. By virtue of its modular design, the model can subsequently undergo "fine-tuning" on supervised tasks, which is faster and cheaper than the pre-training, while taking advantage of the pre-learned representations.

In the MLM task, as shown in Figure 2.12, when a token `[MASK]` feeds the model, its final hidden vector passes through a softmax layer (see Equation 2.1) that predicts the token that was originally in the text. In other words, the model is trained to retrieve the most probable tokens according to its surrounding context. At training time, each token has a 15% chance of being altered and thus updated by its gradient. Since the `[MASK]` token will not be used later in production, once a token is selected for gradient update, the input token will be replaced, 80% by a `[MASK]` token, 10% by a random token and remain unchanged otherwise (also 10%). A cross-entropy loss as defined in Equation 2.2 is used to fit the MLM task.

The NSP task consists of predicting whether two sentences follow each other by performing a binary classification of the token `[CLS]`, which stands for "classify". During the training phase, every sequence starts with the token `[CLS]`, which retains the representation of the sequence and is succeeded by two sentences divided by the token `[SEP]`. If the two sentences follow each other, the sequence will be labeled `IsNext` and `NotNext` if they do not. As an example, here would be a positive classification: `[CLS, 'my', 'dog', 'is', 'cute', SEP, 'he', 'likes', 'play', '##ing']` → `IsNext`, and a negative would be: `[CLS, 'my', 'dog', 'is', 'cute', SEP, 'he', 'likes', 'play', '##ing', 'piano']` → `NotNext`. In half of the cases, the following sentence will be its continuation, while in the remaining instances, a sentence will be selected at random from the corpus. After being pre-trained, the `[CLS]` token has the capability to represent the complete input sequence, which can be useful for unsupervised tasks like document clustering.

The PLM comes in various sizes, which involves a trade-off between resources and performance since larger models generally achieve better overall results. Although the paper only compares base and large models, a variety of sizes, such as tiny, mini, small, and medium, are also published on their GitHub repository[8]. These models have been pre-trained on the BooksCorpus (Zhu et al., 2015) and the English Wikipedia, which contain 800M and 2,5 billions (B) words, respectively. The training process uses a batch size of 256 sequences for 100M steps, a learning rate of 1e-4, and the Adam optimizer (Kingma and Ba, 2017) with a learning rate warm-up for the first 10k steps.

2.4.2.3 Fine-tuning

After completion of the pre-training phase, the modular design of the BERT model allows the connection of the PLM building block to a newly initialized layer for

[8] `https://github.com/google-research/bert`

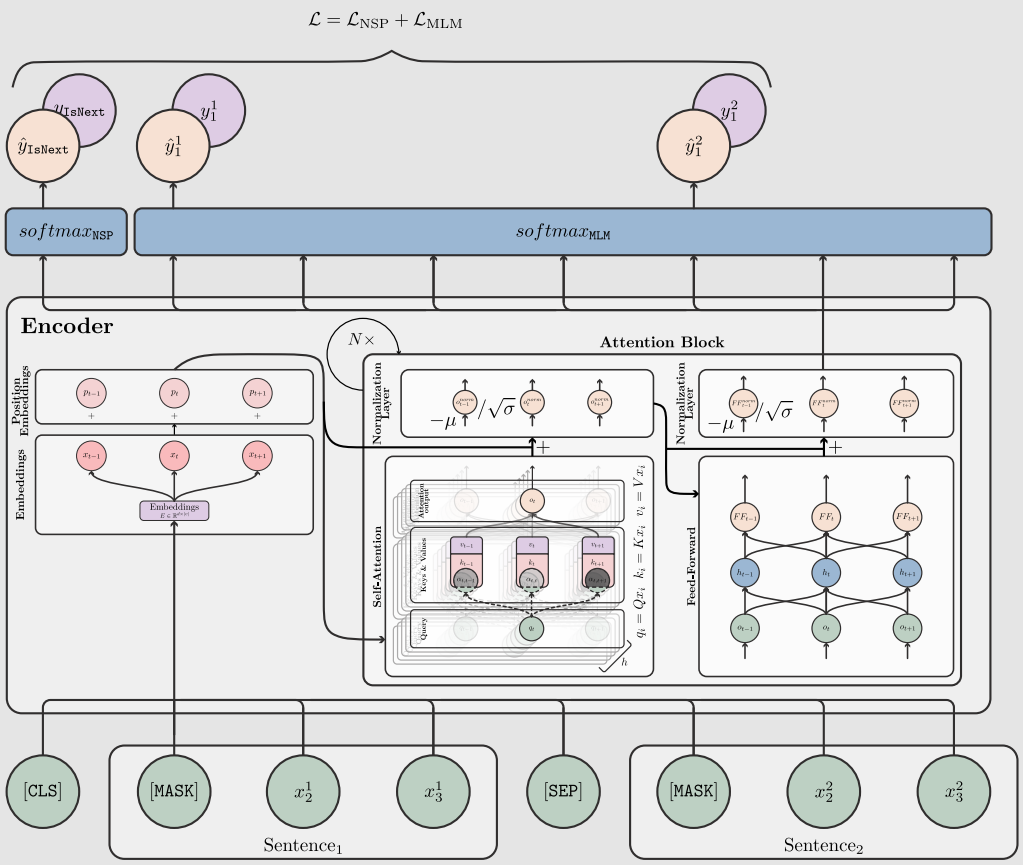

Figure 2.12: BERT Pre-Training Diagram - A pair of sentences, separated by the `[SEP]` token, are input into a Transformer encoder, which was introduced in Figure 2.11. The resulting outputs are linked to two different loss functions, which are optimized simultaneously. The first token, `[CLS]`, is linked to $\mathcal{L}_{\text{NSP}}$, responsible for determining whether the sentences are sequentially related. The second loss function, $\mathcal{L}_{\text{MLM}}$, focuses on predicting the `[MASK]` tokens by considering their surrounding context.

fine-tuning. In the case of a sequence classification task, a softmax layer is added on top of the final hidden layer corresponding to the `[CLS]` token. For tasks like NER and QA, the last hidden state of each token is used to classify their respective entities. The recommended hyperparameters for fine-tuning by the authors include a batch size ranging from 16 to 48 and a learning rate between 5e-5 and 2e-5.

Performance is assessed on GLUE (Wang et al., 2018), SQuAD (refer to Section 2.1.2) versions 1.1 and 2.0, as well as Situations with Adversarial Generations (SWAG) (Zellers et al., 2018), which assess grounded common sense using pairs of sentences. In every task, BERT significantly outperforms all previous SOTA models. On average, across all tasks, the larger version outperforms the previous best results by approximately 7 points.

2.4.3 BERT Variations

Following the success of BERT, numerous variations of models have emerged. Various aspects like hyperparameters, the language or domain of the corpus, or even the type of tokenizers offer endless possibilities for tweaking to enhance performance on downstream tasks. To this day, the renowned library `HuggingFace` hosts approximately 0.5M models in its repository[9]. In the upcoming sections, we will introduce the most notable models, emphasizing their distinctions and highlighting their contributions to the field. However, some model categories will not be covered in this section, notably distilled models, in particular, DistilBERT (Sanh et al., 2020) or generative models such as Generative Pre-trained Transformer (GPT).

2.4.3.1 Variations in Pre-training Methods & Hyperparameters

The first set of publications that tried to tweak the original BERT aimed to enhance the model for similar tasks by refining some of the model's methods. In (Liu et al., 2019), there are two main contributions to the field of language modeling with BERT. Firstly, the paper conducts an ablation study on several crucial steps of the method or hyperparameters, such as the format of the input sequence, the inclusion or exclusion of the NSP training objective, the batch size, the number of training steps on a larger corpus, and the use of a different tokenizer. Second, Robustly optimized BERT approach (RoBERTa), a PLM that incorporates the optimal hyperparameters identified in their research, is made publicly available.

Their ablation study reveals the following findings.

- The original BERT model applied static masking, which means that the same words were masked for each epoch during pre-training. The study demonstrates that employing dynamic masking, that is, varying masking for each epoch, yields a comparable or slightly improved performance compared to a static approach.

- The study compares four different methods to input data into the model, two with/without the NSP training objective. The results indicate similar or slightly better performance without the NSP task. Removing the NSP task allows loading the 512 tokens sequence to its maximum capacity, as there are no longer constraints on the alignment of the two sequences required by the NSP task.

- The results suggest that increasing the batch size leads to better performance for the same computational cost compared to the original BERT batch size of 256. Essentially, running the model for 125k steps with a batch size of 2k is more efficient than running it for 1M steps with the default BERT batch size.

- As mentioned already, the models BERT use a WordPiece tokenizer, although BPE seems to show slightly worse end-task performance on some tasks (results not shown in the paper), the fact that BPE uses a universal encoding scheme is believed to be a better choice. By using BPE, the authors set the vocabulary

[9] `https://huggingface.co/models`

size at 50k, which increases the number of parameters by 15M and 20M for base and large sizes, respectively.

RoBERTa, the released model demonstrates superior performance across all metrics in the GLUE benchmark. When evaluated on SQuAD v1.1, it achieves results comparable to the previous SOTA model at that time, XLNet (Yang et al., 2019), which was also released around the same period. In the context of SQuAD v2.0, RoBERTa surpasses the performance of all individual models.

To summarize the differences with BERT, RoBERTa uses a larger BPE tokenizer and dynamic masking without the NSP training objective with an 8k batch size for 500k steps. It is worth noting that without the NSP task, RoBERTa is unable to give a sequence vector representation out-of-the-box, as BERT can do through its `[CLS]` token.

Up to the present time, RoBERTa remains extensively used in the field, with approximately 16M downloads of the base version reported in January 2024[10]. For comparison, during the same period, the original monolingual non-DS base BERT model had 38M downloads, serving as a common reference point in academic work, while DeBERTa (He et al., 2021) ranked third with 5M downloads.

2.4.3.2 Variations in Corpora Domain

By using DS corpora in the pre-training phase, PLMs such as FinBERT (Yang et al., 2020) and LegalBERT (Chalkidis et al., 2020) show improvement, respectively, in both financial and legal NLP tasks. The first PLM on biomedical corpora is BioBERT (Lee et al., 2019). It takes BERT as a starting point and keeps pre-training the model on both PubMed and PubMed Central (PMC) corpora, accounting for about 18B words, which is more than the training corpora of BERT. The released model is then fine-tuned on a NER, a RE, and a QA biomedical tasks. The article shows that the more biomedical text feeds the model, the better it gets in the DS downstream tasks. BioBERT outperforms BERT in almost all biomedical tasks and has quickly become a standard of its kind.

With the abundance of biomedical literature, (Gu et al., 2021) demonstrate that it was not necessary to start with a general-domain PLM, in order to train a biomedical LM. Surpassing BioBERT in nearly all tasks, PubMedBERT is currently the most widely used biomedical PLM, with approximately 922k downloads in January 2024 compared to 230k for BioBERT. A notable distinction in their approach compared to BioBERT is the decision to train the model from the ground up. Consequently, PubMedBERT's weight initialization is independent of the BERT tokenizer. Instead, both a WordPiece tokenizer and a randomly initialized BERT architecture are trained on a PubMed corpus of 14M abstracts. In contrast to RoBERTa, PubMedBERT is pre-trained using both MLM and NSP training objectives, similar to the original BERT. The authors employ Whole-Word Masking (WWM) (Cui et al., 2021), which involves masking each token of a given word if one of its subwords is selected. PubMedBERT excels over all previous SOTA models in nearly all tasks of BLURB, which was introduced in the same paper (see Section 2.2.1).

The paper also shows how important the training of a DS tokenizer is to the success of the model. It first runs an ablation study showing the impact on

[10] `https://huggingface.co/FacebookAI/roberta-base`

performance when switching to an in-domain vocabulary. Then, it hypothesizes that the observed improvement is due to the fact that the tokenizer does not need to break down DS terms into word pieces as much as the BERT tokenizer. To demonstrate such behavior, the paper first shows that common biomedical terms are missing in general-domain vocabulary, forcing the tokenizer to break down important words into pieces. It finally shows a few examples of misclassification of models using general-domain tokenizers, depicting the difficulty of the model to trace a signal among word pieces. These examples are juxtaposed with PubMedBERT, which accurately classifies each example since its tokenizer does not need to break down any common biomedical terms.

2.4.3.3 Variations in Corpora Language

HuggingFace's platform hosts a variety of PLMs that have been trained on corpora in different languages, some monolingual and others multilingual. For instance, the Chinese version of BERT was developed directly by HuggingFace[11]. Although some models such as the Japanese[12] and Korean[13] variations have been pre-trained without formal publications, most models are published along with their research papers that describe their training methods. Examples include BETO and BERTIN for Spanish (Cañete et al., 2023; la Rosa et al., 2022), BERTimbau for Portuguese (Souza et al., 2020), and HindBERT for Hindi (Joshi, 2022). We will focus on three models, CamemBERT (Martin et al., 2020), FlauBERT (Le et al., 2020) and XLM-RoBERTa (Conneau et al., 2019), two French and a multilingual models, respectively.

The corpus used to train CamemBERT, the first French BERT PLM, consists of the French section of Open Super-large Crawled Aggregated coRpus (OSCAR) (Ortiz Suárez et al., 2019). The model employs the SentencePiece version of the BPE tokenizer, which is trained on the corpus with a vocabulary of 32k tokens. A WWM approach is implemented for an MLM pre-training objective using a batch size of 8k for 100k steps without the inclusion of the NSP task, which is similar to RoBERTa. Although the model outperforms other SOTA models in different downstream tasks, it should be noted that being the first French model, it is compared against other BERT-based models that are multilingual, which are not typically the most suitable baselines, as they often do not perform as well as monolingual models when comparing models of the same sizes (Lample and Conneau, 2019).

For its pre-training, FlauBERT's French corpora were gathered from various sources. Subsequently, they underwent processing using a BPE tokenizer with an expanded vocabulary of 50k tokens. In contrast to CamemBERT, FlauBERT employs a subword masking approach for the MLM objective instead of WWM. Unlike CamemBERT, FlauBERT also investigates a model of a larger size denoted as $\text{FlauBERT}_{\text{large}}$ with 373M parameters. Both models use the same batch size of 8k; however, the total number of training steps is not specified, only the computational time is provided, approximately 13k GPU hours for the base and 50k GPU hours for the large version. $\text{FlauBERT}_{\text{large}}$ generally exhibits superior performance compared

[11] `https://huggingface.co/google-bert/bert-base-chinese`
[12] `https://huggingface.co/tohoku-nlp/bert-base-japanese`
[13] `https://huggingface.co/kykim/bert-kor-base`

to other models, which is expected given its double parameter size. The performance of FlauBERT$_{\text{base}}$ is comparable to or slightly inferior to models of similar size. As of January 2024, the number of downloads for CamemBERT in that month stands at 2.6M, which is roughly 78 times higher than the downloads for FlauBERT.

Similar to previous models such as XLM (Lample and Conneau, 2019) or mBERT (Devlin et al., 2018), XLM-RoBERTa stands out as the current standard for a multilingual general-domain PLM. Both the model and the tokenizer underwent training on a corpus containing 100 languages. A substantial vocabulary size of 250k was created using SentencePiece's implementation of the Unigram Language Model. Expanding the size of the vocabulary results in an increase in the number of model parameters. For instance, adding 220k tokens to the vocabulary expands the BERT$_{\text{base}}$ model from 110M to 270M parameters, making it 2.4 times larger. To address the issue of splitting low-resource language words at the character level, the tokenizer was trained on sentences sampled from the multilingual corpus using a distribution that mitigates bias towards high-resource languages.

In their paper, various ablation studies highlight the challenge of multilingualism, which can be observed with a decrease in overall performance when incorporating new languages into a model. By analyzing performance decrease separately for high- and low-resource languages, their findings show that models tend to underfit. Indeed, although high-resource languages are negatively affected by the addition of each new language, low-resource languages initially show improvement due to language complementarity, benefiting from similarities with other languages. However, as the performance of both low- and high-resource languages starts to decline, one way to counteract this decline is by increasing the model size. Pre-trained on a 2.5TB corpus, XLM-RoBERTa surpasses other multilingual PLMs in various downstream tasks while being able to deal with 100 languages.

2.4.3.4 Variation in Both Corpora Domain & Language

Foreign DS PLM can be difficult to find, given the rarity of their training datasets. In fields such as biomedicine, where data scarcity is a common issue, models are often initialized based on another existing model due to limited data availability (Shrestha, 2021; Schneider et al., 2020; Carrino et al., 2021; Türkmen et al., 2023). For example, a recent development in the French language is CamemBERT-bio, which enhanced the performance of CamemBERT by pre-training it on French biomedical data (Touchent et al., 2023). However, as demonstrated in the PubMedBERT study (Gu et al., 2021), the CamemBERT tokenizer has a tendency to split common biomedical terms more frequently than a tokenizer trained on a standard dataset, potentially leading to inaccurate sequence representations in the model.

DrBERT (Labrak et al., 2023b) is the first French biomedical PLM to be developed entirely from the ground up, trained on a corpus comprising 4GB of private data and 7.4GB of public data. DrBERT tokenizer uses SentencePiece's BPE implementation with a vocabulary size of 32k tokens. Four fixed-size models with 110M parameters were pre-trained on varying corpus sizes to facilitate performance evaluation. These models underwent optimization over 80k steps with a batch size of 4k sequences. In addition to the models, a French biomedical benchmark encompassing two POS Tagging, three NER, a multi-label classification, and a Multiple-Choice Question Answering (MCQA) tasks were introduced. Since CamemBERT-bio, the only other French biomedical PLM, was released during

the same period, there were no performance comparisons with any French DS PLM. DrBERT results demonstrate improvements over English DS or French general-domain models such as BioBERT, PubMedBERT, and CamemBERT on their benchmark and four other proprietary tasks.

2.5 Machine Translation

In this section, we will first discuss the evaluation of MT systems by presenting BLEU, the metric commonly used in the field. We will then provide a brief overview of Statistical Machine Translation before delving into the evolution of Neural Machine Translation leading up to the introduction of the M2M-100 model. Unlike traditional multilingual systems, M2M-100 is trained not only on aligned corpora centered on English. Although this strategy requires substantial efforts in data gathering, we will see that it showcases significant improvement in performance in non-English directions.

2.5.1 Machine Translation Evaluation

In contrast to NLU, NLG is a field that aims to produce textual outputs. Text generation can be complex to evaluate automatically due to the numerous possible correct answers. As human evaluation is labor-intensive and costly, MT model assessments used to involve the comparison of model perplexities (Bengio et al., 2003). Fortunately, a proposed metric that requires a source sentence and one or more target translation options enables accurate comparisons of MT systems.

2.5.1.1 Bi-Lingual Evaluation Understudy

The Bi-Lingual Evaluation Understudy (BLEU) metric was first introduced in (Papineni et al., 2002) and has subsequently become the standard evaluation metric for MT. Calculating the metric is not straightforward, as it was designed to address various shortcomings that a translation metric might exhibit. The BLEU score is specified as follows, typically using the default values of $N = 4$ and $w_n = 1/N = 1/4$:

$$BLEU = BP \cdot \exp(\sum_{n=1}^{N} w_n \cdot \log p_n) \tag{2.14}$$

Where BP is a brevity penalty that exponentially tends to zero the smaller the generated translation (candidate) is with respect to the target sentence (reference):

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{1-r/c} & \text{if } c \leq r \end{cases} \tag{2.15}$$

Where c and r are the candidate and the reference number of words, respectively. In Equation 2.14, BP multiplies the geometric average of the modified n-gram Precision (p_n), which can be defined as the number of n-gram candidate/reference matches over the candidate's number of n-grams.

$$p_n = \frac{Count_{clip}(n-gram)}{Count(n-gram)} \tag{2.16}$$

In its modified variant, the n-gram Precision takes into account repetitions in the generated candidate by using $Count_{clip}$, which is basically $min(Count, max\_ref\_Count)$. That way, a n-gram is never counted more than the number of times it appears in a reference sentence.

2.5.2 Statistical Machine Translation

MT has been a challenge for seven decades, initially addressed by linguistic specialists who created Rule-Based Machine Translation (RBMT) systems between the 1950s and 1970s. Subsequently, in the 1980s, the focus in the field of MT shifted towards Example-Based Machine Translation (EBMT) systems, which operate by using pre-translated sentences as models and adjusting them as needed. Throughout a decade, this area experienced significant activity, with opportunities for enhancement across all aspects, including storage and matching techniques, as well as methods for adapting examples through rules.

Statistical Machine Translation (SMT) is the first MT data-driven method, consisting of splitting the translation problem into two, each of which can be treated by its own model. For a sentence pair (S, T) of, respectively, a source and target languages, the idea is to find S from which a translator would have produced T. Using Bayes' theorem, maximizing $P(S|T)$ is the same as (Brown et al., 1990):

$$\arg\max_S P(S)P(T|S) \tag{2.17}$$

The $P(S)$ is computed by a LM and $P(T|S)$ by a translation model. The problem can be seen as the translation model suggesting words or phrases from the source language that might have produced the words that we observe in the target sentence, which will then be ordered by the LM.

Considerable effort is put into enhancing both submodels (Brown et al., 1992; Schwenk et al., 2006). Thanks to research on word-alignment, such as IBM Model 1 (Brown et al., 1993), Phrase-Based systems can be trained on aligned corpora. This type of model has fueled translation services including Google Translate for more than a decade, until recent Deep Learning (DL) models revolutionize MT. Since the current state of the field does not directly incorporate any SMT SOTA models or concepts, we will not go into this area further.

2.5.3 Neural Machine Translation

Unlike SMT, Neural Machine Translation (NMT) systems produce output in a seamless manner without the need for word-aligned datasets. The Transformer model introduced earlier (see Section 2.3.2.4) falls under the category of NMT models. As depicted in the Seq2Seq workflow in Figure 2.9, all NMT systems consist of an encoder, responsible for deriving a latent representation from a source sentence, and a decoder, which uses the encoder's output for translation. Although modern SOTA NMT models are predominantly based on Transformers, they have been developed using various architectural approaches.

The first article introducing this type of model architecture uses a Convolutional Neural Network (CNN) for encoding and a RNN for decoding (Kalchbrenner and Blunsom, 2013). Similar to the variations of BERT discussed in Section 2.4.3.1, researchers have been experimenting with enhancing BLEU scores in different languages by adjusting various methods aspects such as preprocessing techniques (e.g., tokenization), model architecture, or specific components namely LSTM or GRU cells.

Taking advantage of their renowned Seq2Seq architecture, (Sutskever et al., 2014) demonstrated SOTA results in an English-to-French translation experiment employing RNNs for both the encoder and the decoder. Furthermore, the translation of long sentences was enhanced by feeding the source sequence in reverse order to a RNN with LSTM cells. Since only a single word can be predicted at a time, a beam search decoder that evaluates the k most likely predictions at each stage is used instead of relying solely on the most probable word. This optimization technique, which ultimately yields the most likely prediction in general, has become standard practice in modern MT frameworks.

Following the integration of attention mechanisms (Bahdanau et al., 2014) (see Section 2.3.2.2) into RNNs, researchers began to integrate subword segmentation algorithms (see Section 2.3.1.2) into NMT systems. This strategy, previously explored in the domain of speech recognition (Schuster and Nakajima, 2012), addresses issues related to infrequent words by dividing them into subword units, each receiving its unique vector representation. Through the use of the Huffman code method in their preprocessing step, (Chitnis and DeNero, 2015) managed to enhance a model, which encodes rare words using a token `[UNK]`, by up to 1.7 BLEU points in a French-to-English translation task.

As discussed in Section 2.3.2.4, Transformer (Vaswani et al., 2017) outperforms existing SOTA models in directions such as English-to-German and English-to-French. Its parallel processing capabilities enable it to handle an extensive amount of data, offering potential for further enhancements. The Transformer model stands as the leading architecture in various DL domains, including computer vision (Dosovitskiy et al., 2020) and speech recognition (Radford et al., 2022).

2.5.4 Many-to-Many Multilingual Translation Model

In this section, we introduce the well-documented M2M-100 model (Fan et al., 2020). The complete source code for both data generation and model training is available on GitHub[14]. Along with documentation, three different sizes of the model (418M, 1.2B and 12B parameters) are also available for four GPU memory size configurations (2x32GB/4x16GB/6x12GB/8x8GB).

2.5.4.1 M2M-100 Model

M2M-100 is the first non-English-centric Many-to-Many (M2M) multilingual translation model. Based on the Transformer architecture, it accounts for 12 encoder and decoder layers for a total of 1.2B parameters. For comparison, the Transformer (big) had 213M of parameters. Since all parameters are common across languages,

[14] `https://github.com/facebookresearch/fairseq/blob/main/examples/m2m_100/README.md`

special tokens are assigned to the encoder and decoder to indicate the source and target languages.

A shared vocabulary of 128k tokens was developed by training a SentencePiece tokenizer that covers all languages. To prevent underrepresentation in low-resource languages, monolingual data were injected into the corpus in addition to a sampling correction. During the generation process, a beam search with a size of 5 is used. The paper does not provide details on the batch size or the total number of steps. The model is openly available in three sizes 418M, 1.2B, and 12B parameters.

2.5.4.2 Dataset Building

Three main criteria were taken into account when choosing languages: (1) the selection should encompass language families spoken extensively in various regions, (2) each language should have an evaluation benchmark that is publicly available for evaluating the model, and (3) each language should have at least some monolingual data accessible to facilitate large-scale corpus mining. Following the compilation of a comprehensive list of 100 languages, a dataset containing 7.5B aligned sentences was produced through the application of data mining methods and Backtranslation (BT) covering a total of 2200 directions. This dataset was then used to train the M2M-100 model.

As most parallel sentence corpora (bitext data) pass through English, the authors used LASER (Artetxe and Schwenk, 2019), a multilingual encoder known to generate embeddings, in conjunction with CCMatric (Schwenk et al., 2020) and CCAligned (El-Kishky et al., 2020) projects to identify aligned sentences in a wide range of corpora. Given the impracticality of analyzing all 9900 language combinations, the selection of language pairs was done thoughtfully. Initially, the 100 languages were grouped into 14 clusters based on common characteristics such as geographical proximity or linguistic origin. This grouping facilitated the identification of language pairs within each cluster due to the inherent connections that bind them. Subsequently, to establish links between clusters, the top 1-3 languages with the most available resources from each cluster were selected to identify inter-cluster language pairs.

BT creates artificial bilingual texts using monolingual data, which involves generating additional data through the translation of monolingual target sentences into the source language. However, as it is time-consuming even for a single direction, the emphasis is placed on 100 directions with a BLEU score ranging from 2 to 10. As per (Caswell et al., 2019), a unique `BT` token has been included on the encoder side of these translations to signal to the model that they are synthetic. An experimental analysis demonstrates that incorporating the BT dataset with the mined data leads to improved BLEU scores in nearly all language directions.

2.5.4.3 Results

An ablation study emphasizes the importance of training the model on non-English-centric data, showing similar results for translation to and from English, but significant improvements of more than 5 BLUE points for non-English translation pairs. In zero-shot translation, which involves translating between language pairs the system was not trained on (Gu et al., 2019), the M2M approach outperforms the English-centric's by almost 11 BLUE points.

Another comparison has been conducted regarding the model density, demonstrating that wider models exhibit better scalability compared to deeper ones. Specifically, for the same words per second at training, wider models generally achieve higher overall BLUE scores. Additionally, the performance of all three model sizes is evaluated across low, medium, and high-resource languages. On average, the results show an increase of approximately 2 BLEU when switching from 418M to 1.2B parameters, and around 1.5 BLEU from 1.2B to 12B parameters. These findings suggest that the model may still be underfitting however, the runtime and memory requirements for further scaling up would be excessively prohibitive given the expected potential improvement in performance.

The main results illustrate the practicality of this model in a real-world case scenario. While the Switzerland case is not specifically referenced, there exist other areas globally where the use of multiple languages besides English is prevalent. In more than 30 different language pairs, recent SOTA BLEU scores have seen an average increase of 7.6 BLEU points. In particular, if we were to consider the case of Switzerland, improvements of 9.9 and 7.2 BLEU points have been achieved in the Italian-to-German and Italian-to-French language pairs, respectively.

To contrast the model with English-centric ones, the authors used benchmarks from the famous Workshop on Machine Translation (WMT). Across 13 directions, the current top-performing bilingual standalone model is outperformed by up to 7.6 BLEU points, while it falls short in four directions, with three showing minimal differences and one exhibiting a deficit of 5 BLEU points (English $\rightarrow$ Chinese). On average, there is an enhancement of approximately 2 BLEU points across all evaluated English-centric directions. Furthermore, for three additional multilingual benchmarks, the model outperforms previous SOTA results.

A final assessment is conducted by experts to evaluate the semantic accuracy of non-English-centric languages in both intra- and inter-cluster language directions. Most directions produce scores ranging from 8.5 to 9.5 on a scale of 10. However, despite remaining reasonable, the results tend to be lower for low-resource languages. A further human evaluation is performed in a blind test setting in 10 directions, where a comparison is made with an English-centric model. The results reveal a higher translation quality for the M2M system in all 10 directions.

2.6 Synthetic Translated Data in Natural Language Understanding

As demonstrated in this literature review, the introduction of BERT following the Transformer model has enabled pre-training on extensive datasets with remarkable efficiency across various common tasks. This level of generalization has made any previous generation of NLU system obsolete in performance, which means that trying to improve Word2Vec performance through translation would be impractical. Consequently, our examination of related work focuses on recent post-BERT publications that seek to boost the performance of LM by leveraging artificially generated translated data. Although the BT technique, as described in Section 2.5.4.2, has been proven to be effective, it will not be covered as it is specifically tailored for training MT systems.

To this day, a limited number of publications use translated data for training

NLU systems. This could be due to the fact that recent research on multilingual PLMs (Conneau et al., 2019), discussed in Section 2.4.3.3, has shown great results in various general-domain tasks. Therefore, researchers may opt to use synthetic data, such as translated text, when they encounter limitations in the availability of real data. As illustrated in Section 2.4.3.1, the scarcity of corpora is often associated with either low-resource languages or DS in foreign languages. Three approaches have been pinpointed as relevant in related work. The first subsection will investigate the application of translation in the final stage, specifically by translating the downstream task dataset. The subsequent subsection will explore recent researches that utilize translation or partial translation to pre-train a LM for low-resource languages. Lastly, a paper that leverages biomedical translation to keep pre-training a generative LM on a low-resource language will be presented.

2.6.1 Synthetic Translated Data at the Downstream Task Level

In (Isbister et al., 2021), sentiment analysis is approached in four low-resource Scandinavian languages using three different methods. The first approach fine-tunes a native monolingual PLM on the original downstream task datasets, the second translates each sequence of the downstream task datasets into English and then fine-tunes an English PLM on the translated data, and finally the third fine-tunes a multilingual PLM directly on the native downstream task datasets. Generally, the results favor the third method, which employs the multilingual model. However, it is worth noting that fine-tuning the English model with translated data generally produces superior results compared to fine-tuning the low-resource language PLM.

2.6.2 Synthetic Translated Data for Language Model Pre-Training in Low-Resource Languages

Luxembourgish is a low-resource language that has a close structural and etymological relationship with German. In their study, to address the scarcity of data to train a LM, a partial translation of common and unambiguous words from the high-resource auxiliary language was performed (Lothritz et al., 2022). Subsequently, four strategies were evaluated across five downstream tasks. The first leverages mBERT, the multilingual variant of BERT; the second and third use a BERT pre-trained on either the entire available Luxembourgish corpus of 130M words or a combination of Luxembourgish and German data; the fourth fine-tunes LuxemBERT, a BERT pre-trained on corpora comprising half Luxembourgish and half German data that have been partially translated into Luxembourgish. An experiment comparing models trained on three different dataset sizes indicates that the data augmentation approach improves performance in downstream tasks. Although LuxemBERT demonstrates superior performance compared to mBERT, the statistical significance of the Wilcoxon test yields a p-value of 0.109, which is not optimal.

After the introduction of ElhBERTeu (Urbizu et al., 2022), a PLM trained on a corpus of 351M words in Basque, a strategy has been implemented using synthetic translated data to improve the corpus size of this low-resource language (Urbizu et al., 2023). Using Spanish as the auxiliary language, a MT Transformer Base

model consisting of 65M parameters has been trained on 8.6M parallel sentences in order to translate a corpus from Spanish to Basque. Subsequently, all their BERT models have been trained using a Unigram tokenizer with a vocabulary size of 50k and a batch size of 256 for 1M steps on a WWM implementation of the MLM task. Evaluated on BasqueGLUE (Urbizu et al., 2022), which comprises nine downstream tasks, the results show that the PLM trained solely on synthetic data is competitive, although it does not outperform the model trained only on a native Basque corpus. Another experiment shows that pre-training a LM on translated data can give comparable results when using the same information by pre-training two LMs on a parallel corpus, one on the Basque part, the other on a synthetic translation in Basque of the Spanish part. Subsequently, to show that the domain and cultural context of the Spanish translated data had an impact on the model performance, the authors pre-trained another LM on a corpus that geographically and culturally filtered text from the Basque Country before translation. The model with a filter performed slightly better than the one without filtered data, even if it was substantially trained on a smaller corpus. Finally, a study that tweaks the native/translated data ratio shows that the addition of synthetic data enhances the native PLM performance.

2.6.3 Synthetic Translated Data for Pre-Training Domain-Specific Generative Language Model in Low-Resource Languages

As they all fall into the NLG group of models, neither GPTs (Brown et al., 2020) nor Text-to-Text Transfer Transformers (T5s) (Raffel et al., 2023) have been discussed in this literature review. However, it is important to note that these models have the potential to address NLU tasks through prompt engineering, which involves framing a task as a text generation problem. In their paper, (Phan et al., 2023) enhance Mtet (Ngo et al., 2022), the current SOTA MT model in the English-to-Vietnamese direction by injecting synthetic biomedical parallel text into its training corpus. Although no details on the size of the MT models are mentioned, their fine-tuned MT system outperforms the models to which it is compared, that is, M2M-100, Google Translate and Mtet, in two translation test sets covering both general and biomedical domains. The resulting translation model is used to generate ViPubmed, a Vietnamese-translated corpus comprising 20M abstracts, as well as ViMedNLI, a benchmark dataset generated by translation of MedNLI (Romanov and Shivade, 2018) and refined with human experts. Subsequently, ViPubmed is used to keep pre-training ViT5 (Phan et al., 2022), the first pre-trained T5 for the generation of the Vietnamese language, while ViMedNLI is used for fine-tuning. ViPubMedT5, the final model, outperformed models including ViT5 in ViMedNLI and acrDrAid, an acronym disambiguation task, while being close second in a summarization task, showing that using artificially translated data can improve model performance.

Chapter 3

Methods

The aim of this chapter is to gather all the common methodologies that will be employed in Chapter 4 and Chapter 5 into one place. Each section in this chapter serves as modules that can be integrated into the subsequent one. It starts with (1) the assembly of a vast life science corpus, which will be incorporated into (2) a translation module, creating TransCorpus, the first fully translated corpus in the life sciences field consisting of 22 millions (M) abstracts translated from English to French. Subsequently, the training of TransTokenizer, TransBERT and cTransBERT (3) will be carried out using (2) TransCorpus. While TransBERT leverages TransTokenizer, which is pre-trained on TransCorpus, cTransBERT utilizes CamemBERT tokenizer. Finally, (4) the fine-tuning datasets and tasks will be presented, where both (3) TransTokenizer and TransBERTs will be plugged in. Despite being the most time-consuming part of this thesis, no results are anticipated from this chapter. However, whenever feasible, an interim results section will be included at the end of a module to evaluate whether the process was executed correctly.

3.1 Biomedical & Life Sciences Literature Corpus

Following a succinct overview of MEDLINE, PubMed, and MEDLINE/PubMed Baseline Repository (MBR), this section briefly justifies the choice of the dataset used to build the training corpus. At the end of it, a comparison will be drawn between the freshly retrieved corpus containing 22M abstracts and the corpora used to train other models.

3.1.1 PubMed & MEDLINE & PubMed Central

MEDLINE is a repository of life sciences references, particularly focusing on biomedicine. This database is essential for researchers, healthcare professionals, and students, providing access to over 30M citations from more than 5,200 journals around the world. MEDLINE's records are meticulously indexed with National Library of Medicine (NLM)'s Medical Subject Headings (MeSH), which significantly improves search precision and efficiency. In addition, it covers a broad spectrum of biomedical disciplines, ranging from fundamental research to clinical practice and public health.

Within this framework, PubMed stands as a search tool created and maintained by NLM's National Center for Biotechnology Information (NCBI). It integrates MEDLINE as its primary component while offering a broader spectrum of information. PubMed gives access to more than 36M[1] citations from life science journals, online books, and MEDLINE. Although it focuses mainly on cataloging journal articles in the fields of biomedicine and health, PubMed also includes references to complete articles accessible through PubMed Central (PMC) and publisher platforms. Accessible to the public since 1996, PubMed has simplified the exploration and retrieval of medical information.

Since 2002, MEDLINE/PubMed Baseline Repository (MBR) offers access to snapshots of the MEDLINE/PubMed database at specific times. These snapshots are static and represent the citation data without updates to the MeSH vocabulary and other revisions that typically occur throughout the year. The baselines are generated at the beginning of each new MeSH indexing year, usually in mid-November, and cover all citations in MEDLINE up to that date. In its last version, it accounts for over 36M citations. The primary purpose of MEDLINE/PubMed Baseline Repository (MBR) is to function as a historical archive, allowing researchers to examine and analyze the data as it was at the time of each baseline creation. This can be especially beneficial for longitudinal studies and for monitoring changes in the medical literature and indexing practices over time. The collection includes MEDLINE references with MeSH, OLDMEDLINE and PubMed-not-MEDLINE entries[2]. Following the creation of the baseline files, daily updates are distributed, including new, updated and removed records.

PMC functions as a free digital archive, currently hosting over 9M freely accessible full-text articles, predominantly from the biomedical and life sciences journal literature. Functioning as the National Institutes of Health (NIH) digital archive for biomedical and life sciences journals, PMC plays an essential role in ensuring that research funded by NIH is universally accessible. While it does not publish, it serves as a repository for journal literature.

3.1.2 Corpus Compilation

In our review of the literature, we discuss the top biomedical Pre-trained Language Models (PLMs), BioBERT, and PubMedBERT, which have been trained using MBR, PMC, or a combination of both. According to (Lee et al., 2019), the performance comparison between a version trained on PMC and another version trained on MBR's abstracts does not reveal any significant gain in performance. Although the citation count in MBR is approximately three times the full-text count in PMC, the latter has about three times more words. In essence, a corpus based on PMC would be more voluminous but would cover fewer citations. Consequently, translating full-text articles would require a substantial increase in resources, with no projected performance benefits. Therefore, the corpus will utilize MBR's abstracts.

For the building of this life sciences corpus, the 2021 MBR baseline, encompassing 31M citations, and updates up until April 2021 was downloaded. These data

[1] `https://pubmed.ncbi.nlm.nih.gov/`

[2] `https://lhncbc.nlm.nih.gov/ii/information/MBR/MEDLINE_Baseline_Repository_Detail_2017.pdf`

repositories, comprising 1,062 and 150 files for the baseline and updates respectively, hold collections of JSON documents that maintain citation details such as title, abstract, journal, authors, PMID, MeSH, among others, as illustrated in Figure A.1 in Appendix A. Subsequently, in order to get a clean set of abstracts, each citation in the dataset that includes a PMID, a title, and an abstract is kept and its raw text is modified by substituting any sequence of one or more whitespace characters with a single space. Figure 3.1 presents an example of a title and abstract after modification, as it would appear prior to translation.

PMID: 44

Title: The origin of the alkaline inactivation of pepsinogen.

Abstract: Above pH 8.5, pepsinogen is converted into a form which cannot be activated to pepsin on exposure to low pH. Intermediate exposure to neutral pH, however, returns the protein to a form which can be activated. Evidence is presented for a reversible, small conformational change in the molecule, distinct from the unfolding of the protein. At the same time, the molecule is converted to a form of limited solubility, which is precipitated at low pH, where activation is normally seen. The results are interpreted in terms of the peculiar structure of the pepsinogen molecule. Titration of the basic NH2-terminal region produced an open form, which can return to the native form at neutral pH, but which is maintained at low pH by neutralization of carboxylate groups in the pepsin portion.

Figure 3.1: Example of a Citation From the MBR Database

A considerable amount of citations lack one of the three essential attributes, i.e. title, abstract, or PMID. For example, citations before 1975 do not include abstracts. Consequently, after applying this procedure to the complete dataset, our corpus comprises 21,567,136 abstracts, amounting to 30.2GB of raw text, 202,190,607 sentences, 4,362,901,244 words, and 6,748,255,011 BERT tokens. Table 3.1 shows a comparison with other models corpora. As mentioned already, despite that both BioBERT and PubMedBERT have a version that also includes PMC full-text articles, only those that use PubMed are displayed for a better comparison.

Despite a few missing unknown values, Table 3.1 provides a comprehensive comparison of our corpus statistics against several models such as BERT, BioBERT, and PubMedBERT. This comparison is crucial for understanding the scale of data that similar models have been trained on, which directly impacts their performance and applicability in various Natural Language Processing (NLP) tasks. In terms of word count, the proposed corpus contains approximately 4.4 billions (B) words, which is closely aligned with BioBERT's 4.5B while exceeding PubMedBERT's 3.1B and BERT's 3.3B. This high word count is indicative of the extensive material that the model will be exposed to after translation. Even though it is in a different language, the statistics of the DrBERT corpus emphasize the scale disparity when compared to English corpora, underscoring the lack of available Domain-Specifics (DSs) open data in foreign languages.

	Corpus Before Translation	BERT	BioBERT	PubMedBERT	DrBERT
Abstracts	22M	N/A	-	14M	N/A
Size	30.2GB	16GB	-	21GB	7.5GB
Sentences	202M	-	-	-	54M
Words	4.4B	3.3B	4.5B	3.1B	1.1B
Tokens	6.7B	-	-	-	-

Table 3.1: Corpus Statistics for Different Models - 'N/A': Not Applicable, '-': Unknown value, Tokens number is computed using a Bidirectional Encoder Representations from Transformers (BERT) cased tokenizer.

3.2 Corpus Translation in French

This section will cover the decision-making steps that guided the translation process, and then it will provide a comprehensive explanation of the procedure undertaken to translate 22M abstracts from English to French. This resulted in the development of TransCorpus, the first fully synthetic life science corpus translated from English to French. Intermediate results are presented to ensure that everything went well and allowing us to move on to the next module.

3.2.1 Translation Approach

As highlighted in the literature review, the M2M-100 model is a State-of-the-Art (SOTA) translation system capable of translating between 98 source and 97 target languages. The model's parameters are fixed and can be obtained from its GitHub repository [3]. This fixed nature provides a significant advantage over online translation services like Google Translate. These online services tend to be more costly and may also implement updates without notice, which can hinder reproducibility.

We have two main factors to evaluate: (1) the model's size, which can be either 418M, 1.2B, or 12B parameters, and (2) the translation techniques, which can be executed either on a sentence-by-sentence basis or on the entire abstract. Each technique has its own pros and cons. Sentence-wise translation might be faster than abstract translation but could be less accurate due to the lack of context a sentence might need. To determine the suitable model size and method, a sample of abstracts was translated using the Fairseq library. The following subsections provide a summary of the Fairseq framework and discuss the analysis that informed our translation approach selection.

3.2.1.1 Fairseq Library

Fairseq is a Python library created by Facebook AI Research, who are the authors of the M2M-100 paper (Fan et al., 2020). Therefore, besides supporting Graphics Processing Units (GPUs) and being built with Pytorch, a widely-used framework in modeling, Fairseq is utilized in the documentation of the M2M-100 project on its

[3] `https://github.com/facebookresearch/fairseq/tree/main/examples/m2m_100`

GitHub page. Although this model can be integrated into a HuggingFace generative pipeline, the fastest method for translating large volumes of text is through the Fairseq generative API, which enables sequences to be processed in parallelizable batches.

First, to translate a list of text sequences, the Fairseq generation API requires that they must be tokenized using the model's SentencePiece encoder. Tokenizing before translation allows sorting the sequences by size, facilitating quick parallel translation by bucket. The tokenized sequences are then transformed into a binary file, which serves as the input for the generative function.

Several key parameters can be adjusted to address memory issues. Specifically, when generating sequences, the batch size is not the only parameter to consider, as the sequence length generated is unpredictable. In our experiments on a V100 GPU, we found that using `--max-tokens=4600` instead of a fixed batch size was effective, as it allowed the batch size to vary depending on the input length. Other relevant parameters that were not modified in the corpus translation are `--max-len-a` and `--max-len-b`, which define the output maximum length as `max-len = max-len-a * source-len + max-len-b`, a linear combination of these two parameters.

3.2.1.2 Model Size & Translation Method Selection

As a reminder, in the M2M-100 paper (Fan et al., 2020) three versions are introduced : a small, base, and large, with 418M, 1.2B, and 12B parameters, respectively. When considering the average enhancement across languages with low, mid, and high resources, the transition from the small to the base version results in an average improvement of 1.9 Biomedical Language Understanding Evaluation (BLUE). Similarly, upgrading from the base to the large version leads to an increase of 1.4 BLUE. However, it is important to note that the first increase multiplies the parameter count by 2.9, while the second one amplifies it by a factor of 10.

Deploying the large model would be computationally prohibitive. Indeed, given its substantial size, it requires four distinct GPUs, which requires inter-GPUs communication, further decelerating the translation process. The combination of high computational costs and minimal performance gains led to the decision to discard this model size for our next translation tests.

The first method for selecting model size and translation techniques is through quantitative analysis. Based on a 1000-abstracts sample, Figure 3.2 compares (a) the input level by examining the number of tokens per sentence or abstract, (b) the translation time per abstract by both model sizes and methods, and lastly, (c) the word distribution after translation for both model sizes compared to the original distribution methods.

Figure 3.2a illustrates that by splitting abstracts into sentences, many small-sized sentences are clustered together, facilitating faster processing through effective batch handling. Conversely, abstracts generally exhibit a more dispersed token distribution, making them less suitable for parallel processing, with many abstracts exceeding the token limit that the model can accommodate.

In the second graph (Figure 3.2b), a noticeable difference between the two methods can be seen. Sentence-wise translation is evidently faster, and an exponential growth is noted when the model size is increased, comparing both methods. This phenomenon occurs because the computational complexity of transformer models,

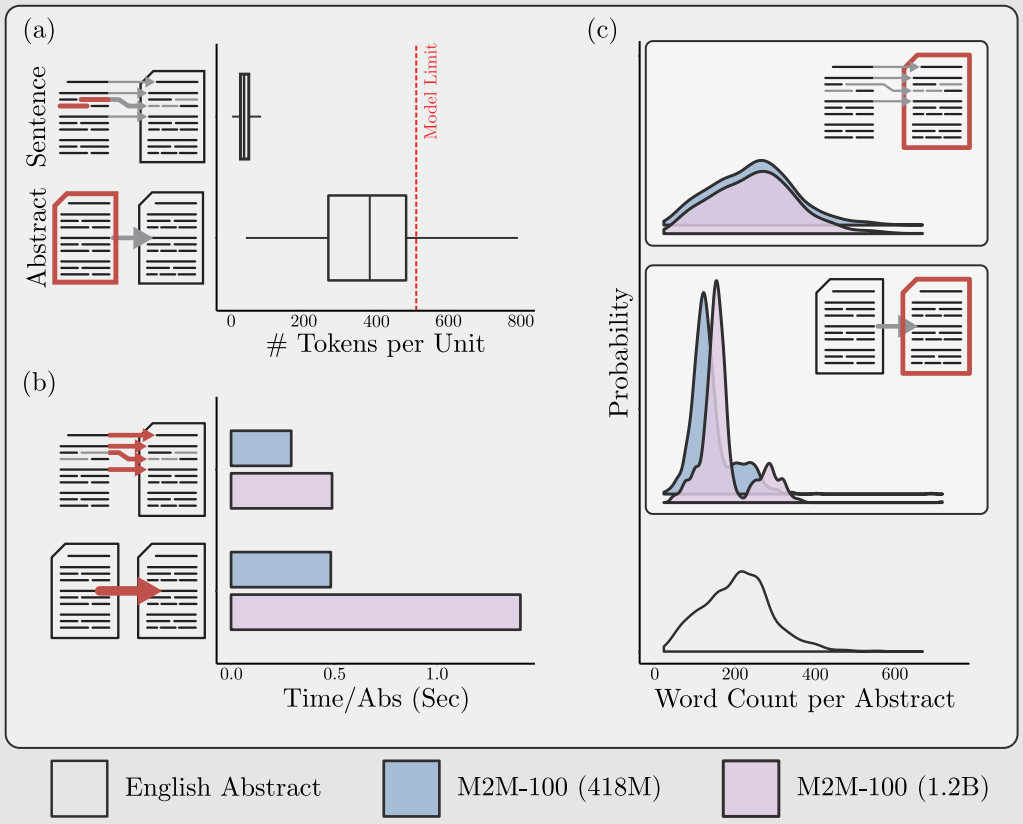

Figure 3.2: Abstract translation analysis on a 1000 abstracts sample - (a) Box plot showing the number of tokens per sentence and abstract, with a red line at 512 tokens representing the maximum token limit that M2M-100 can handle. (b) The average time in seconds to translate each abstract using the 418M and 1.2B model versions, comparing sentence-level and abstract-level translation. (c) Distribution of word count per abstract for both model sizes, displayed with the original English abstract at the bottom when translating by abstract (middle) and by sentence (top). All distributions are normalized to the same scale, so their areas add up to 1.

especially in the self-attention layers, increases quadratically with the sequence length. When the model size grows and longer sequences are used, the translation time per abstract is nearly quintupled.

In Figure 3.2c, when performing translations sentence-by-sentence (top), the distribution is very similar to that of the original abstract in English (bottom). Yet, an irregular word distribution when whole abstracts are translated at once can be observed on the middle distributions of In Figure 3.2c.

Qualitatively, a detailed examination of the translations indicates that the distribution disparity observed in Figure 3.2c is partially due to a 'repetition' problem. Appendix B shows an observed example, all four translations are displayed for comparison. It is worth noting that M2M-100 was trained on sentence pairs and is probably aimed to be used the way it was trained.

Both quantitative and qualitative analyses led to the choice of sentence-wise translation. Following some extrapolations and using multiple V100 GPUs, the use

of the larger model was deemed feasible. In the next stage, we will detail how the entire corpus will be translated with the 1.2B parameters model using sentence-wise translation.

3.2.2 Large Scale Translation Process

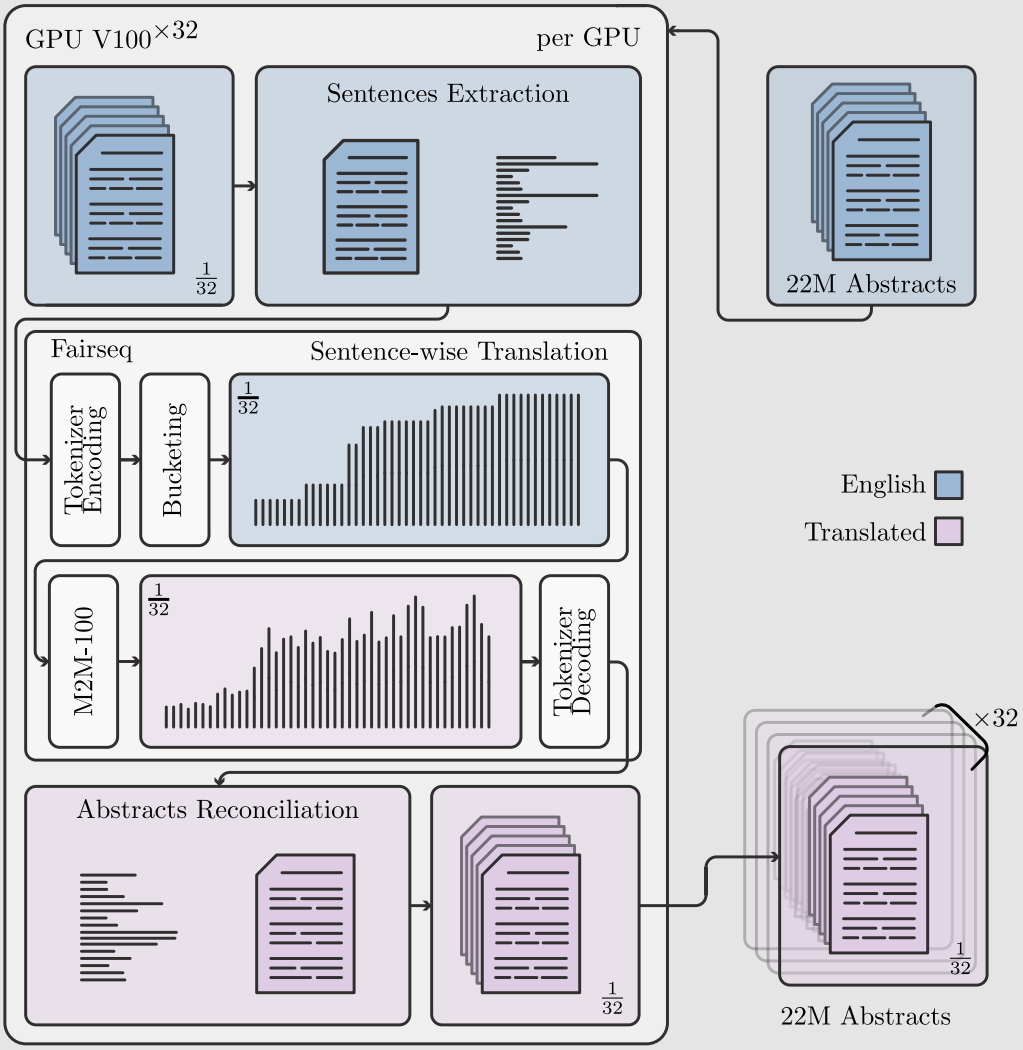

Figure 3.3: Large Scale Translation Workflow - Following the extraction of 22M abstracts from JSON files, the corpus was shuffled to reduce length biases, then divided and allocated across 32×GPUs. Before translating $\frac{1}{32}$ of the corpus, each abstract was broken down into sentences. The Fairseq toolkit encoded each sentence with the model's tokenizer and translated them into batches using bucketing to optimize the process. Once translation was finished, sentences were decoded back into strings and reassembled into abstracts. Finally, all pieces of the translated abstracts were concatenated, completing the translation of the entire corpus.

Translating the entire dataset requires significant resources. Two virtual machines running Ubuntu, each equipped with 80 CPUs and 16× V100 GPUs (totaling

32 V100s), were employed for approximately two weeks, amounting to around 11.52 thousands (k) GPU/hours. Figure 3.3 depicts the method used for the translation process.

First, the 22M abstracts extracted from the JSON files are divided and distributed across the two machines to parallelize the translation process. Each abstract is then split into sentences, with the Fairseq package handling the tokenization and translation of these sentences in batches. Once the sentences are translated, they are matched to their respective abstracts and sentence numbers, and the entire corpus is reconstructed.

Grouping a batch of sentences of the same length allows for bucketing. Though it may seem counterintuitive, there is a considerable increase in speed when translating sentences of the same length simultaneously. This is why it is essential not to translate abstracts individually using the HuggingFace generative pipeline as it would take years to translate a corpus this size.

The splitting of abstracts into sentences is performed using a pre-trained sentence tokenizer using the Natural Language Toolkit (NLTK)(Bird et al., 2009) library. Each sentence must have a minimum of 10 characters; if a sentence is shorter, it is merged with the following one. Appendix C presents an example of split and tokenized abstract. Figure 3.4 shows an example after translation and reconstruction sentence-by-sentence

PMID: 44

Title: L'origine de l'inactivation alcaline du pepsinogène.

Abstract: Au-dessus du pH de 8,5, le pepsinogène est converti en une forme qui ne peut pas être activée en pepsine en cas d'exposition à un pH bas. L'exposition intermédiaire au pH neutre, cependant, renvoie la protéine à une forme qui peut être activée. Des preuves sont présentées pour un changement réversible, de petite conformation dans la molécule, distinct du déploiement de la protéine. Dans le même temps, la molécule est convertie en une forme de solubilité limitée, qui est précipitée à faible pH, où l'activation est normalement observée. Les résultats sont interprétés en termes de la structure particulière de la molécule de pepsinogène. La titration de la région terminale de base NH2 produit une forme ouverte, qui peut revenir à la forme native à pH neutre, mais qui est maintenue à un pH bas par la neutralisation des groupes carboxylés dans la portion de pepsine.

Figure 3.4: Example of Title and Abstract Citation From the MBR Database Translated in French (McPhie, 1975)

3.2.3 Intermediate Results

Considering that our study comprises interconnected modules and our research question seeks to evaluate the performance of the final model, we have opted to add a short section for interim results at the end of each module section. This approach ensures that we can confirm everything went smoothly along the way.

In this section, TransCorpus will be presented, compared with others and a few samples will be analyzed.

3.2.3.1 TransCorpus

After translation, the resultant raw text file is 36.4GB, containing 221M sentences and 5.25B words. Table 3.2 compares TransCorpus with the only two French life science corpora leveraged for pre-training. The comparison reveals that DrBERT, although it utilizes the largest corpus until now, is about five times smaller than TransCorpus.

	TransCorpus	**DrBERT**	**CamemBERT bio**
Size	36.4GB	7.5GB	$\sim 4GB^*$
Sentences	221M	54M	-
Words	5.25B	1.1B	413M

Table 3.2: Corpus Statistics After Translation Compared to DrBERT's - '-': Unknown value, '*': Number obtained by extrapolation because only the size in GB for a given proportion are disclosed.

Additionally, both DrBERT and CamemBERT bio have diverse types of sources for their corpora, which might be confusing for a Language Model (LM) during the pre-training phase. For instance, CamemBERT bio includes both scientific literature, drug leaflets and clinical cases/leaflets. If a provided sequence is too short for the model to deduce a context helping it identify the kind of document it is receiving, this may cause confusion, potentially resulting in ineffective learning. In such scenarios, it would be wise to indicate the source at the beginning of the sequence with a special token. This approach is similar to what MLM applies to facilitate language translation, where the first token signifies the language, thereby aiding the model in avoiding confusion.

While the corpus size is important, its quality must also be closely monitored. Our source for the corpus is already considered a benchmark of quality for the training we plan to perform as it is used by BioBERT and PubMedBERT. However, it remains crucial to assess the quality of our translations to make sure that everything has been conducted properly.

To assess the quality of translation, we will employ the same approach used in the assessment of translation methods. In particular, Figure 3.5 illustrates the distribution of words per abstract, which raised concerns when reviewing Figure 3.2c, as a kind of binomial distribution appeared when entire abstracts were translated. Thankfully, Figure 3.5 shows the same distribution shape as observed in our previous sample study for sentence-by-sentence translation, which also centered on about 250 words per abstract.

3.2.3.2 Translation Comparison to True French

Although DrBERT and CamemBERT bio do not provide examples of the raw data used to train their models, the nature of our research methodology necessitates showcasing a few instances. While evaluating the translation quality through

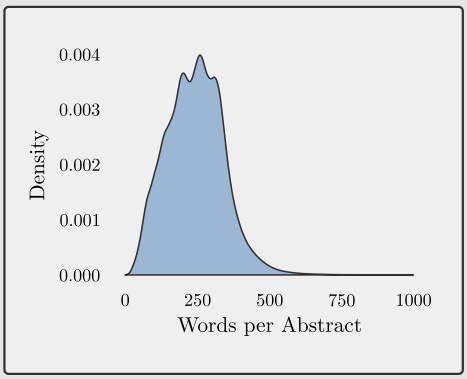

Figure 3.5: Distribution of the Number of Words per Abstract -

manual assessment would have been feasible, deriving a metric such as Bi-Lingual Evaluation Understudy (BLEU) score without any points of reference would seem useless.

Luckily for us, there are authentic French abstracts in PubMed that have already been translated in TransCorpus. In fact, PubMed allows setting filters to display original articles in French. After acquiring a French abstract, we can verify whether the associated PMID has been translated. Figure 3.6 presents an article (Lauby-Secretan et al., 2019) found in PubMed that already had been translated. Additional examples are available in Appendix D, obtained from PubMed through three diverse queries aimed at creating a comprehensive lexicon: *psychology*, *chemical drugs*, and *molecular genetics*. Even though a French speaker might observe minor discrepancies in certain writing styles or find acronyms incorrectly recomposed, the translations are predominantly precise.

Original:
La prévalence du surpoids et de l'obésité est en augmentation dans le monde depuis plusieurs décennies, chez les hommes comme chez les femmes. En France, la prévalence du surpoids chez les adultes atteint 49 % en 2015 (54 % des hommes et 44 % des femmes), dont 17 % d'obèses. D'après la dernière évaluation réalisée par le CIRC en 2017, le surpoids et l'obésité sont des facteurs de risque établis pour 13 localisations de cancers avec un risque de cancer chez les obèses variant fortement en fonction des localisations cancéreuses. En 2015 en France, on estime que 5,4 % des cancers étaient attribuables à l'excès de poids soit 18 600 cas, dont 3400 cancers du côlon, 2600 cancers du rein, 4500 cancers du sein et 2500 cancers de l'endomètre. L'obésité est aussi associée à un moins bon pronostic pour certains cancers, en particulier les cancers du sein et du côlon. L'obésité chez les enfants et les adolescents, en augmentation dans de nombreux pays, a également été associée à une augmentation du risque de cancer à l'âge adulte. L'obésité a pour origine principale un déséquilibre de la balance énergétique et est favorisée par un régime alimentaire riche en produits transformés, viande rouge, acides gras trans et saturés, boissons et aliments sucrés et pauvres en fruits et légumes, légumineuses et céréales complètes. Les principales recommandations nationales et internationales en matière de réduction de la prévalence de l'obésité préconisent donc de pratiquer une activité physique et d'avoir une alimentation équilibrée.

Translated:
Au cours des dernières décennies, la prévalence de l'obésité et du surpoids a augmenté dans le monde entier, tant chez les hommes que chez les femmes. En France, la prévalence du surpoids chez les adultes était de 49% en 2015 (54% chez les hommes et 44% chez les femmes), dont 17% chez les adultes obèses. Selon la dernière évaluation réalisée par l'IARC en 2017, le surpoids et l'obésité sont des facteurs de risque établis pour 13 sites de cancer avec des estimations de risque par 5 kg/m2 qui varient en grande partie en fonction du site de cancer. En 2015, en France, 5,4% des cas de cancer pouvaient être attribués à l'excès de poids, correspondant à 18 600 cas, dont 3400 cancers du côlon, 2600 cancers du rein, 4500 cancers du sein et 2500 cancers de l'endomètre. L'obésité est également liée à un mauvais pronostic pour certains cancers, en particulier les cancers du sein et du côlon. L'obésité chez les enfants et les adolescents, également en hausse dans de nombreux pays, a également été associée à une augmentation du risque de cancer chez l'adulte. Une cause majeure de l'obésité est un déséquilibre dans l'équilibre énergétique favorisé par un régime riche en aliments transformés, viande rouge, acides gras trans et saturés, aliments et boissons sucrés et pauvres en fruits et légumes, légumes et céréales entières. Les principales recommandations nationales et internationales pour réduire la prévalence de l'obésité sont d'avoir un régime alimentaire équilibré et une activité physique régulière.

Figure 3.6: Comparison of Translation Original True French - Translation and original of [Lauby- Secretan et al., 2019] (PMID: 31227175)

3.3 Language Model Training

This section elaborates on the training process for a tokenizer and two language models. It supports the recommendation of using SentencePiece as the tokenizer for enhanced multilingual compatibility and describes the training of TransTokenizer, a Unigram tokenizer trained on 10M translated abstracts. For LMs, two models are pre-trained using TransCorpus: TransBERT, which employs TransTokenizer, and cTransBERT, which uses the CamemBERT tokenizer trained on native French corpus. The pre-training of each model adheres to the Robustly optimized BERT approach (RoBERTa) setup and spans approximately 90 days on three A100 GPUs per model.

3.3.1 Tokenizer Training

In contrast to PLMs, there is no established consensus on the most effective tokenizer, as subword segmentation algorithms aim to split words optimally using probability. For the purposes of this thesis, considering the potential addition of more languages in the future, choosing a tokenizer capable of handling specific linguistic features could prove beneficial. As noted in the literature review, SentencePiece treats whitespaces as regular characters rather than relying on them, which means that it is suited for all kinds of languages.

As SentencePiece tokenizers require a considerable amount of RAM to run, 10M translated abstracts were randomly selected in order to train a DS tokenizer based on our synthetic translated corpus. The original SentencePiece implementation[4] (Kudo and Richardson, 2018) is used to train a Unigram tokenizer with a vocabulary size of 32,000 and a character coverage set to 0.9995 (default values). It took approximately four hours to train the model on a machine with 600GB of RAM.

3.3.2 Language Model Training Settings

Based on the literature review LMs comparisons, the choice in the pre-training setting was straightforward. It basically features RoBERTa's hyperparameters, which is a BERT base architecture featuring 12 hidden layers, each with 12 attention heads of dimension 768 and an extensive batch size of 8k. The Adam Optimizer with default settings is used, along with 24,000 warmup steps and a learning rate of 6e-4. To achieve this batch size, we accumulate 28 gradient steps on each batch of 96 per GPU, resulting in an effective batch size of 8,064 sequences.

Two $\text{BERT}_{\text{base}}$ model architectures were trained: one with the tokenizer trained on the synthetic data (see Section 3.3.1) and the other with the CamemBERT tokenizer (Martin et al., 2020), which was trained on a native French corpus. Using these hyperparameters, each model's training process lasted approximately 90 days on 3×A100s on Baobab High-Performance Computing (HPC)[5] of the University of Geneva, totaling roughly 6.48k GPU/hours for the 500,000 steps. To assess the models and mitigate overfitting, set_{dev} and set_{test}, both containing 100,000 abstracts, were evaluated: set_{dev} every 10,000 steps and set_{test} on the model with the lowest $loss_{dev}$ at the end of the training loop.

[4] `https://github.com/google/sentencepiece`

[5] `https://www.unige.ch/eresearch/en/services/hpc/`

Following the RoBERTa implementation, the Next Sentence Prediction (NSP) task is excluded, focusing solely on the Masked Language Model (MLM) task. By employing an adaptive on-the-fly dataloader, both RoBERTa's dynamic masking strategy and full-sentence input format are emulated. This ensures that each input consists of complete sentences sequenced from one or multiple abstracts, capped at a maximum length of 512 tokens. As described in the referenced paper, inputs may span across document boundaries. When an abstract ends, sentences from the subsequent abstract are incorporated, with an additional separator token between abstracts.

3.3.3 Intermediary Results

As already mentioned, an intermediary results section which evaluates a modular milestone is necessary. If the intrinsic value of pre-training a LM can only be observed on a downstream task, there are two aspects that need to be reviewed before going to the next stage. TransTokenizer, which has been trained on TransCorpus should be evaluated and compared with DrBERT and CamemBERT. Finally, the MLM task should be assessed and compared with the same models using the Pseudo-Perplexity, which will be presented in the following sections.

3.3.3.1 TransTokenizer

In Chapter 5, various comparisons between CamemBERT and TransTokenizer will be presented. A common metric for evaluating tokenizers is the number of tokens generated for a sequence. Generally, fewer tokens are preferable, as they typically result in reduced noise within vector representations. An examination of the number of tokens produced by CamemBERT and TransTokenizer across various DS Named Entity Recognition (NER) datasets revealed that TransTokenizer uses significantly fewer tokens to encode the same entities. Specifically, while CamemBERT almost doubles each entity with a tokenization rate of 1.99, TransTokenizer increases it to 1.65, which represents a delta of 20%. Figure 3.7 illustrates how TransTokenizer processes a named entity consisting of three words with three tokens, whereas CamemBERT decomposes it into smaller segments. Although subword tokenization enables the tokenization of any words, the representation of '__infarctus' is likely more accurate than the combined vector representation of '__inf', 'arc', 'tu', 's'.

Entity: ['infarctus', 'du', 'myocarde,'] (3 words)
TransTokenizer: ['__infarctus', '__du', '__myocarde', ','] (4 tokens)
CamemBERT: ['__inf', 'arc', 'tu', 's', '__du', '__my', 'oc', 'arde', ','] (Δ+5)

Figure 3.7: CamemBERT Vs TransTokenizer Sample - An example of tokenization shows that the tokenizer of TransBERT (i.e., TransTokenizer) requires less tokens than the tokenizer of CamemBERT to encode the same sequence.

Considering that TransTokenizer was entirely trained on a corpus which was itself relying on the translation model tokenizer, concerns arose about its effectiveness on actual DS datasets. Despite using 20% fewer tokens than CamemBERT,

understanding how TransTokenizer compares to an authentic DS tokenizer was critical. Consequently, the same experiment was conducted with the DrBERT tokenizer, which is trained on its DS corpus. DrBERT yielded a ratio of 1.55, indicating that TransTokenizer needs 6% more tokens to represent the same sequence. Nonetheless, it is crucial to highlight that DrBERT's tokenizer primarily focused on clinical text, akin to the type used in this assessment. Yet, this experiment demonstrates that the training of TransTokenizer has been effective. Appendix I presents examples of tokenizations as detailed in Chapter 5 illustrating differences from both CamemBERT and DrBERT.

3.3.3.2 TransBERT and cTransBERT

Two LMs have been pre-trained on TransCorpus: TransBERT and cTransBERT, which utilize TransTokenizer and CamemBERT's tokenizer, respectively. Throughout the pre-training phase, an assessment was performed using different translated abstracts on the MLM task to prevent overfitting. However, since this set of abstracts consisted solely of synthetically translated data, it was not used to compare different tokenizers. To gain an understanding of how the models perform relative to others, Pseudo-Perplexity (PPPL) was calculated across the models using a sample of 50 authentic French abstracts. To comprehend PPPL as outlined in (Salazar et al., 2020), one must understand the concept of Pseudo-Log-Likelihood scores (PLLs). This metric assesses the likelihood of a sentence according to the model. To derive such a score, one must determine the probability of each word in a sentence given the surrounding words.

$$\mathrm{PLL}(W) = \sum_{t=1}^{|W|} \log P(w_t | w_{\setminus t}) \tag{3.1}$$

Given W as the set of words forming a sentence, w_t as the t^{th} word, and $w_{\setminus t}$ as all the other words. The Pseudo-Perplexity (PPPL) is then represented by:

$$\mathrm{PPPL}(\mathbb{W}) = \exp\left(-\frac{1}{N}\sum_{W\in\mathbb{W}} \log \mathrm{PLL}(W)\right) \tag{3.2}$$

Where $\mathbb{W}$ denotes the set of sentences of the corpus to evaluate and N represents the number of tokens that the corpus comprises. Although computing this metric is relevant to all models, comparing their value across models with different tokenizers is complex. Typically, (Salazar et al., 2020) argues that, when models use a different tokenizer, using the number of words as a normalizer (instead of N) offers a better comparison, even if it is not a perfect fit. Table 3.3 presents the evaluation results for the 50 French abstracts retrieved. A simple whitespace separation was used to separate the words in the text.

This table corroborates our previous tokenizer analysis, which indicated that CamemBERT's tokenizer requires significantly more tokens to encode the same sequence, whereas DrBERT requires slightly fewer. It should be noted that we are far from the 20% discrepancy observed in our results, as the text used for those figures mainly consisted of medical named entities, which are more complex than abstracts. Focusing on the only two models directly comparable due to their

	TransBERT	cTransBERT	CamemBERT	DrBERT
PPPL_{token}	6.00	4.14	**174.42**	8.30
PPPL_{word}	11.71	8.59	**2474.88**	17.55
$n_{sentence}$	376			
n_{word}	9,204			
n_{token}	12,640	13,934	13,934	12,459

Table 3.3: Pseudo-Perplexity Comparison Across Models - Pseudo-Perplexity across models, with the highest uncertainty highlighted in bold.

use of the same tokenizer, cTransBERT scores 4.14, while CamemBERT scores 174.42. This suggests that even tested on genuine French text, a model pre-trained on translated data can effectively learn life science terminology better than a model trained on native French non-DS corpus. For comparison of other models, PPPL_{word} serves as a reference metric; however, with adjustments, this metric is only indicative. The main takeaway is that CamemBERT operates on a different scale, while LMs trained on life science corpora are comparable.

3.4 Language Model Fine-Tuning

To evaluate the model on multiple tasks, an extensive adaptation of DrBenchmark (Labrak et al., 2024) is implemented to improve consistency and robustness. First, cross-validation is applied to each dataset, followed by Hyperparameter Optimization (HPO) for each task, ensuring that each model is evaluated with its optimal hyperparameter configuration. This approach ensures a fair comparison between models while increasing data size through cross-validation, allowing for proper statistical testing. This section finishes with an examination of each dataset/task of the benchmark, showcasing data samples and fundamental dataset statistics for each case.

3.4.1 DrBenchmark: An Adaptation

As previously noted in the literature review, foreign language DS datasets are challenging to procure, particularly within the biomedical domain where data privacy presents a significant hurdle. Released in May 2024, DrBenchmark (Labrak et al., 2024) stands as the first publicly accessible French biomedical language comprehension benchmark, featuring 20 varied datasets in tasks such as NER, Part-Of-Speech (POS), Semantic Textual Similarity (STS), and classification. The paper is complemented with a GitHub[6] repository comprising the used datasets and the code that have been run in order to evaluate the benchmark on 8 SOTA PLMs. In their paper, the authors conclude that there is no evidence a model is better than others across all the tasks. In the upcoming sections, adjustments to the benchmark methodology will be introduced to enhance training performance using HPO as well as statistical testing through a 5-fold cross-validation procedure.

[6] `https://github.com/DrBenchmark/DrBenchmark`

3.4.1.1 5-Folds Cross-Validation Implementation

In the benchmark's paper (Labrak et al., 2024), to achieve a more precise assessment of PLM performance during fine-tuning, each experiment is performed four times, with the final layer being randomly re-initialized each time. While being broadly used in the field, this method helps to reduce all random effects, e.g. model initialization, and batch order, without providing extra insights about the model's advantage on a particular dataset.

To address this issue, we will merge the sets set_{train}, set_{dev}, and set_{test}, divide each dataset into five subsets and train the model five times as described in Section 2.1.1.6. Therefore, each fold will expose entirely unseen observations, thereby partially preserving the independence assumption among the sets. However, it can be argued that within a k-fold training schema, some intersection between training sets is inevitable when $k > 2$. To streamline the procedure and acknowledging that the model assesses completely fresh observations in each fold, we will relax the independence assumption.

It is worth noting that labels that are very sparse and do not appear at least once in every fold will be excluded from the dataset reporting and statistical testing.

3.4.1.2 Hyperparameters Optimization Implementation

Model performance on a particular dataset is greatly impacted by hyperparameters. However, There are no mention as of how hyperparameters were fixed; one can only deduce that certain groups of hyperparameters appear to be fairly uniform for various types of tasks. In the original DrBenchmark's implementation, each model receives identical sets of hyperparameters, whereas each model can have an optimal set of hyperparameters for a given task, impacting deeply the potential performance of each PLM.

To ensure an unbiased selection of hyperparameters, HPO has been incorporated into our existing fine-tuning framework using the RayTune[7] library. Utilizing Sequential Model-Based Optimization (SMBO) as mentioned in Section 2.1.1.7, we assigned a range of hyperparameters to each task based on the time needed to complete it. In our experiments, some tasks took almost a full day using 4×GPUs to be trained, we decided to decrease HPO hyperparameters range so the training time would not be prohibitive.

Appendix E details the hyperparameters ranges set to keep training time under five hours. This decision stems from the fact that the optimization time scales with the number of models to be fine-tuned, the number of folds, which is five in our case, and the number of tasks, which is 20 in the original DrBenchmark paper. As later, our experiments will focus on fine-tuning a total of four PLMs (4 GPUs × 20 tasks × 4 PLMs × 5 hours), the maximum fine-tuning time is capped at 1,600 GPU/hour in our setup.

The set of hyperparameters is restricted to the following: (1) batch size, which can be adjusted by changing the actual batch size or altering the number of gradient accumulation steps, (2) learning rate, (3) number of epochs, (4) weight decay, (5) warmup ratio, (6) dropout, (7) the number of evaluations before deciding to drop a trial, and (8) the number of trials. To shorten the training duration, we primarily

[7] `https://docs.ray.io/`

decreased the number of trials and epochs as hyperparameters. For tasks like NER and POS, where a small batch size typically yields optimal results, dropping gradient accumulation steps was considered when training time was still longer than five hours.

In our preliminary tests, HPO enhanced metrics up to more than 10 points using the same models, data splits, and training code. More importantly, metrics improved consistently across all the tasks.

3.4.1.3 Multiple Training Repetition Re-implementation

After a model has completed a training iteration with HPO on all tasks, it will undergo four additional rounds of retraining using the previously optimized hyperparameter sets on a freshly initialized model. This extra process, as initially introduced in the original DrBenchmark paper, helps to prevent a fortunate initialization from unfairly enhancing a model's performance for a specific dataset or task. Consequently, each model will be trained and evaluated over five folds, five times, totaling 25 runs per task or dataset. Although this fine-tuning repetition was already implemented, this time, it takes into account the newly optimized set of hyperparameters set in the first round.

It is important to note that this will only serve to modify training randomness and will not enhance statistical power during testing. This is why, in order to evaluate significance at the class level, the iterations of models must be aggregated at the prediction phase, prior to evaluation. The key concept is that if one model misses a classification decision, for instance, while the other four rounds capture it, the combined predictions will consider these minor errors and adjust them to reflect what a particular PLM would typically predict.

3.4.1.4 Sparse Dataset or Method Improvement

Each dataset was examined for duplicates, and empty entries. Tasks that are divided into sub-tasks will be combined if feasible and required. In Section 3.4.3, each dataset will be described, and any alterations to the original code will be highlighted.

For NER, sequences without labels were excluded from the entire dataset as it may confuse the model at training if getting batches with little to no label. Although an alternative could have been to eliminate such sequences exclusively from the training set, the decision to remove them from the whole dataset was driven by the cross-validation setting, which aims to provide consistent support across each fold. While empty sequences can help evaluate models for false positives, it is worth noting that NER sequences are usually sparse, with labeled tokens comprising a minor fraction, thereby false positives are inherently assessed by design.

Since DEFT2020 was under a license, the dataset could be accessed with the authors' permission. After identifying an issue with downloading DEFT2021, the download code remains unsuccessful, so this task will not be considered in our analysis.

Additionally, minor enhancements will be applied to the training methods when needed, such as randomizing the order of sentences in the STS task. Whenever such a modification is made, it will be mentioned in the relevant dataset section.

3.4.2 Downstream Tasks & Metrics

The downstream tasks and their corresponding metrics have already been presented in Section 2.1.1. Unless mentioned otherwise in the subsequent section, the original training codes were employed. The primary metric utilized will be the F_1-Score, unless specified differently.

3.4.3 Datasets

The following sections will present each of the available datasets of the benchmark by task. It will start with a small description, followed by a data sample and basic statistics to get the intuition on what to expect from it. When required, a clarification about data/model modification will be given. Table 3.4 shows an overview of all the 15 retained datasets with a few details such as the number of instances per label and the type of data source.

Name	Task	Instance	Label	Source	Section
CAS	POS	86,805	30T	CC	Section 3.4.3.3.1
CLISTER	STS	1,000	0 to 5	CC	Section 3.4.3.4.1
DEFT-2020	STS	1,009	0 to 5	CC, encyclopedia & drug	Section 3.4.3.4.2
	CLS	1,100	3C		Section 3.4.3.1.1
DiaMed	CLS	726	15C	CC	Section 3.4.3.1.2
E3C/Clinical	NER	3,270	1E	CC	Section 3.4.3.2.1
E3C/Temporal		5,756	5E		
ESSAI	POS	150,269	29T	Clinical Trial Protocols	Section 3.4.3.3.2
FrenchMedMCQA	CLS	3,102	5C	Pharmacy Exam	Section 3.4.3.1.3
MantraGSC	NER	879	7E	Biomedical, Drug & Patent	Section 3.4.3.2.2
MorFITT	CLS	5,115	12L	Biomedical	Section 3.4.3.1.4
PxCorpus	NER	11,465	30E	Drug	Section 3.4.3.2.3
	CLS	1,727	4C		Section 3.4.3.1.5
QUAERO/EMEA	NER	6,001	10E	Drug & Biomedical	Section 3.4.3.2.4
QUAERO/Medline		6,765			

Table 3.4: DrBenchmark Adaptation: Data & Tasks Summary - By alphabetical order - Overall, every model tested will be evaluated using cross-validation on 15 distinct datasets covering a broad range of tasks. In the Label column, `C` indicates a class within a multi-class framework, while `L` denotes the count of potential labels in a multi-label classification, `T` tag and `E` entity. The instance count reflects the number of positive `C`, `L`, `T` or `E`. In the source column `CC` stands for Clinical Cases.

3.4.3.1 Classification

This section presents four classification datasets with three multi-class and one multi-label.

3.4.3.1.1 DEFT-2020/Task 2

DEFT-2020 (Cardon et al., 2020) includes clinical cases, encyclopedic entries, and drug descriptions. Introduced during the 2020 edition of the annual French Text Mining Challenge known as DÉfi Fouille de Textes (DEFT), this dataset is annotated for two different tasks.

Entailing a classification problem, the second task of the challenge consists of determining which sentence, out of three options, most closely resembles a provided source sentence. A manual review of the dataset reveals that life science-related sentences are sparse; thus, the results sections will emphasize DS entries, as this is the crucial area for model comparison. Figure G.4 shows an example of a source sentence that is linked to another among two other sentences that are somehow related to the source topic while not being similar to the source sentence. The first task of this challenge utilizes the same dataset, but rather than performing a classification, it conducts a STS task, which will be detailed in the STS section.

3.4.3.1.2 DiaMed DiaMed is a dataset that has been launched along with DrBenchmark's paper. It comprises 739 new French clinical cases collected from an open-source journal (The Pan African Medical Journal). The cases have been manually annotated by several curators, one of which is a medical expert, into 22 chapters of the International Classification of Diseases, 10th Revision (Organization, 2015). These chapters provide a general description of the type of injury or disease. Upon analyzing the dataset, 'External causes of morbidity' lacks any observations, effectively reducing the dataset to 21 classes. Furthermore, several classes offer very limited support, which prevents achieving a minimum of one observation per fold. Once these classes are removed, the dataset comprises 726 sequences distributed among 15 classes.

Figure G.6 shows an example of such a classification. In the original code, the "Title" was not taken into account whereas it usually contains essential information in classification tasks, it has therefore been added to the classification pipeline, which is now `title + clinical_case` → `label`. Table F.2 presents the label distribution after adjustments, highlighting an unbalanced dataset where the first three labels comprise over 55% of the total. Initially, within the 21 classes observed, 14 had fewer than 30 observations, but this count now pertains to only 8 classes.

3.4.3.1.3 FrenchMedMCQA

FrenchMedMCQA (Labrak et al., 2022) is a Multiple-Choice Question Answering (MCQA) dataset designed for the biomedical field. It comprises 3,105 questions sourced from actual exams of the French medical specialization diploma in pharmacy, featuring both single and multiple correct answers. In DrBenchmark's paper, the dataset is subdivided into two tasks: (1) create a model that automatically determines the correct answers from the five options provided for a given question and (2) identify the number of answers (ranging from 1 to 5) believed to be correct

for a given question. Figure G.10 shows a question example. During the data-cleaning process, a formatting error (the appearance of \xa0) was fixed. There are 21 identical questions, which are due to typical questions such as 'Which one of these answers is correct?'. We discovered three identical questions with the same set of answers and removed the duplicates.

The final dataset consists of 3,102 unique sets of questions and possible answers. Figure 3.8b illustrates that most questions have only one answer, while the rarest scenario involves five correct answers, where every answer is correct. Interestingly, there are more questions with three correct answers than with two, indicating that the frequency does not decrease strictly with the number of correct answers.

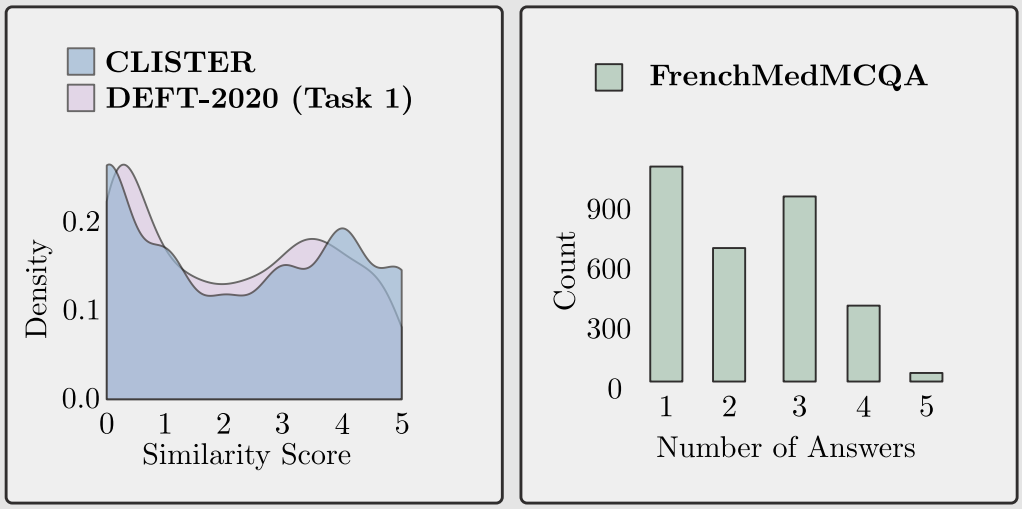

Figure 3.8: Data Distribution for CLISTER, DEFT-2020 (Task 1) and FrenchMedMCQA - Figure (a) depict the similarity score distribution of the CLISTER and DEFT-2020 datasets in blue and pink, respectively. Figure (b) illustrates a histogram of the number of answers of the FrenchMedMCQA

MCQA - Aborted - The first task involves predicting the correct answer(s) for a given question. In tackling the task, the original code uses a model with a softmax layer connected to all possible cases ('a', 'b', ..., 'ab', 'ac', ..., 'abcde'), which seems like a workaround solution to adapt a multi-class model to a multi-label problem. Since this choice is not justified while getting an average score of only 3% on their main metric, a multi-label model is implemented instead for each potential answer. For this task, the Exact Match Ratio (EMR), essentially the ratio of correctly answered questions, is used as the main metric (see Equation 2.9). To increase diversity in answer order, training permutations are applied, and sampling is used during inference. For each question/answer pair prediction, the maximum sample probability is taken, with the default set to $n_{permutation} = 10$. The input format remains unchanged: `question + answers[a+b+c+d+e]` $\rightarrow$ `labels`. Under this new configuration, the results remain on par with the original method (EMR=$\sim$3%).

Further task analysis reveals that treating it as a retrieval problem returning the top-k answers, where k is the number of correct answers for a given question,

boosts the average EMR to ∼16%, a fivefold increase. Even though this result appears to be impressive, randomly providing k answers for each question actually outperforms this method with a EMR of approximately 18%. This indicates that (1) the model has not learned anything in the training phase, and (2) knowing the exact number of correct answers is crucial to enhancing this metric.

A final test with a binary classifier that reranks each question/answer pair independently shows better performance (EMR=∼24%), but this comes at a high cost, requiring seven hours to run a single fold for one model and one trial, which is prohibitive for the potential gain. Additionally, returning the top-k answers usually involves knowing the number of answers to return which is basically the second task of this dataset.

In conclusion, conventional methods do not seem to affect performance, indicating that modeling does not contribute to solving this task. Therefore, it has been decided to remove it from the benchmark to avoid redundant analysis in the upcoming chapters. Notably, even a generative model with 65B parameters such as LLaMa, which has about 590 times more parameters than BERT, struggles with an EMR of 33.76% (Labrak et al., 2023a). The only competitive model tested on this task is GPT-4, achieving an EMR of 72.83%(OpenAI et al., 2024) with its 1,760B parameters, approximately 27 times more than LLaMa, or about 16,000 times more than BERT.

Classification - In this subsequent task, the same dataset is employed, but this time the model is fine-tuned to estimate the count of correct answers. For this purpose, the original code uses the format `question + answers[a+b+c+d+e]` → `number of correct answers`, by linking the PLM to a feed-forward layer followed by a softmax function.

3.4.3.1.4 MorFITT MorFITT (Labrak et al., 2023c) is a dataset aimed at multi-label classification, consisting of 3,624 biomedical abstracts sourced from PMC Open Access and annotated with 12 medical specialties, resulting in 5,116 annotations. After cleaning and identifying one duplicate, the dataset comprises 3,623 sequences and 5,115 labels overall. Figure G.14 illustrates an abstract annotated with two labels: Genetics and Veterinary. Table F.7 presents the distribution of the 12 labels, which are fairly balanced, with the least occurring label appearing 152 times and the most frequent 824 times, making up 16.11% of the dataset. For the classification, the input is tokenized up to the maximum length of 512 tokens. By using a binary cross-entropy loss function, a multi-label model is fine-tuned, which enables the prediction of multiple classes concurrently for a single abstract. The 0.7 threshold, which was previously hardcoded without explicit justification, has now been revised to an arbitrarily established threshold of 0.5.

3.4.3.1.5 PxCorpus/Task 2 PxCorpus (Kocabiyikoglu et al., 2022) is a dataset for spoken language understanding in the context of medical drug prescription transcripts. It comprises 4 hours of transcribed and annotated dialogues (1,981 recordings) related to drug prescriptions. The recordings were manually transcribed and semantically annotated. Our analysis routine of the dataset reveals 177 duplicate sequences which are removed, resulting in a dataset size of 1,727 sequences for both tasks.

A significant portion of the sequences consists of only a few words, with counts such as 80 sequences of length 1, 52 of length 2, and 48 of length 3, averaging 11.9 words per sequence. Figure G.15 illustrates an example from the dataset, where each sequence is used for both NER (Task 1) and classification (Task 2).

The second task of the PxCorpus dataset involves the classification of each sequence into one of four specific intent categories: medical prescription, replace, negate, or none. Since there are no additional data modifications, Table F.9 illustrates how imbalanced the distribution of labels is, with Medical Prescription making up 91.14% of the sequences.

3.4.3.2 Named Entity Recognition

This section introduces four multi-class NER datasets. Both E3C and QUAERO are divided into two sub-datasets each, resulting in a total of six NER datasets for the benchmark. Metrics are computed at the entity level in a strict manner, meaning that the model must predict the exact boundaries of the entity to be considered correct.

3.4.3.2.1 E3C E3C (Magnini et al., 2020) is a multilingual collection of clinical case reports annotated for NER. The dataset includes two categories of annotations : (1) clinical entities such as diseases, (2) temporal details, and factual descriptions such as events. Although the dataset encompasses five languages, only the French section is utilized in this work. As both sub-tasks are NER, an effort has been made trying to merge the two datasets. However, there were 695 shared sequences that could not be combined because some words had conflicting labels. This occurs because the labeling schemas are not mutually exclusive across the datasets. By excluding multi-label word sequences, the combined dataset would result in 1,763 unique sequences, which is significantly less than the 3,048 sequences anticipated from summing both datasets. As a result, the decision was made to abandon merging the two subtasks.

Clinical - The original subset includes 3,498 sequences. We identified 10 duplicates and 1,556 sequences without labels (i.e., sequences full of 'None' labels). Given that nearly half of the sequences lack labels, we chose to remove these entries. This results in a dataset containing 1,941 sequences comprising 57,370 words. Figure G.7 shows an example of CLINENTITY tagging.

By eliminating the entries without labels, the proportion of unlabeled words decreases from 94% to 90%, which still indicates a sparse dataset. Table F.3 shows the dataset of 3,270 entities with only one entity to detect. Same way as POS tagging, models are trained by classifying the first word's token to either '0', 'B-CLINENTITY', or 'I-CLINENTITY', then metrics are computed by named entity i.e. 'CLINENTITY'.

Temporal - By addressing the issues of duplicate and unlabeled sequences, we reduced the dataset from 1,109 to 1,107 sequences. A temporal NER example is displayed in Figure G.8 while Table F.4 presents another unbalanced dataset of 28,726 words comprising 5,756 entities spread across five named entities, with the least frequent appearing 333 times.

3.4.3.2.2 MantraGSC

Merged - The MantraGSC dataset (Kors et al., 2015) is annotated for biomedical NER and covers multiple languages, from which only the French subset is used in this benchmark. This dataset is compiled from three different sources which were initially partitioned to be evaluated with two annotation schemes: Medline (11 classes), and EMEA and Patents (10 classes). These sources encompass various types of documents including biomedical abstracts/titles, drug labels, and patents. Given the extremely small size of the sub-datasets (100 sequences each for EMEA and Medline, and 50 for Patents), they were merged to form a single dataset. After eliminating duplicates and sequences without labels, the combined dataset consists of 238 sequences for 10 classes. An example from each subset is depicted in Figure G.11, G.12 and G.13, while Table F.6 shows the distribution of the combined dataset, showing only three entities occur over 100 times, highlighting the necessity of merging the data.

3.4.3.2.3 PxCorpus/Task 1 The first task of the already described PxCorpus dataset focuses on NER, classifying each word in a sequence into one of 37 categories, including drug, dose, and mode. After data-cleaning, 95 sequences without labels are eliminated, resulting in a NER subset comprising 1,640 sequences spread across 30 named entities. Table F.8 presents the named entity distribution, revealing a substantial number of classes and a highly unequal dataset distribution among named entities. Specifically, 15 out of the 30 entities have fewer than 100 occurrences, and the top five classes represent more than half the label occurrences.

3.4.3.2.4 QUAERO QUAERO (Névéol et al., 2014) provides annotated entities and concepts for NER tasks. The dataset encompasses two genres of text: drug leaflets and biomedical titles, comprising a total of 103,056 words derived from EMEA or Medline.

EMEA - After removing empty sequences, duplicates, and "no-label" sequences, the EMEA dataset accounts for 1130 sequences. Figure G.16 shows an example of a sequence while Table F.10 shows the distribution of named entities. Even if the entities are not distributed evenly, there is a lot more data than in the MantraGSC dataset, which refers to exactly the same entities. This time, only two entries appear less than 100 times against four in the merged MantraGSC dataset (Table F.6).

Medline - With even more data, Medline has not even a single entity appearing less than 100 times. After a data cleaning process, the dataset goes from 2,498 to 2,386 sequences. Figure G.17 and Table F.11 display both an example and the named entities distribution, respectively.

3.4.3.3 Part-of-Speech Tagging

This section presents the two POS datasets of the benchmark. In this task, as each word gets a label, metrics are computed at the word level.

3.4.3.3.1 CAS

The CAS dataset (Grabar et al., 2018) comprises 3,790 clinical cases annotated for POS tagging with 31 different classes. After going through a few data-cleaning steps, 35 duplicate entries were identified and excluded, resulting in 3,753 unique

sequences. Upon dividing the dataset into 5 folds, one class was left out, leading to 86,805 words being spread across 30 classes. Figure G.1 presents an example of POS tagging in a clinical case sentence, where medical terms like "hypogastriques," which is classified as an adjective, are shown. The distribution of tags shown in Table F.1 indicates an imbalanced distribution of tags, with the top five tags accounting for over 65% of the tags. To address POS tagging, each word is tokenized using the model's tokenizer, and the first token representation of each word is classified into one of the 30 classes.

3.4.3.3.2 ESSAI

The ESSAI dataset (Dalloux et al., 2021) includes 7,247 clinical trial protocols annotated with 41 POS tags using an automatic tagger. A data analysis revealed 333 duplicate entries, 13 of which had different labels. Consistent with previous datasets, these duplicates were removed, resulting in a final dataset of 6,068 documents with a total of 150,240 words distributed across 35 classes, 10 of which occur less than 100 times. Figure G.9 shows a sequence extracted from a clinical trial with POS tagging. After splitting the dataset into 5 folds, ESSAI ends up with 150,269 spread across 29 tags, Table F.5 shows the classes distribution of the corrected dataset.

3.4.3.4 Semantic Textual Similarity

This section presents the two STS datasets of the benchmark. Since this task involves forecasting a bounded number, the main metric is the R^2.

3.4.3.4.1 CLISTER CLISTER (Hiebel et al., 2022) is a French dataset designed for STS that focuses on clinical cases. It includes 1,000 pairs of sentences, each rated by multiple annotators with similarity scores ranging from 0 to 5. These scores were then averaged to yield a single floating-point value representing the overall similarity. The aim of this dataset is to develop models that can predict similarity scores matching the reference score based on the given sentence pairs. After further data cleaning, the dataset itself remains unchanged.

Figure G.2 shows an example of a pair of sentences which display a medical lexicon. A modification in the model fitting code has been made in order to allow permutation of the sentences when inputting into the regression model. Indeed, as the similarity score is insensitive to the order of sentences, switching order 50% of the cases is a kind of free data augmentation that can help generalize at prediction. In other words, the model used to receive data in the original order: $\texttt{text}_1$ `+` $\texttt{text}_2$ $\rightarrow$ `Score` and now, randomly, half the time, the model receives $\texttt{text}_2$ `+` $\texttt{text}_1$ $\rightarrow$ `Score`.

Figure 3.8a shows the distributions of the similarity scores across two datasets. In the blue distribution, which refers to CLISTER, we can see that there is a pick frequency at 0.0 (no similarity) and at 4.0 (high similarity), overall, the labels seem to be well distributed.

3.4.3.4.2 DEFT-2020/Task 1 The first task of the DEFT-2020 consists of assigning similarity scores of sentence pairs, ranging from 0 (least similar) to 5 (most similar). With one duplicate, it has a final size of 1009 sentence pairs.

Figure G.3 illustrates two sentences with a similarity score of 3.7. Four out of five evaluators strongly agree on the high similarity of the sentences, resulting in an average score of 4.1. However, one evaluator rates the similarity lower at 2.0, which brings the overall similarity score down to 3.7. This example highlights the significance of having multiple curators to provide an objective assessment that reflects individual opinions. Similar to the CLISTER task, sentence order is reversed 50% of the time. Figure 3.8a shows a distribution pattern resembling the CLISTER task (Section 3.4.3.4.1), with a high frequency of sentence pairs having a similarity score around 0 and a small peak near 3-4.

Chapter 4

TransBERT: A Synthetically Translated Language Model

Following a brief introduction and the presentation of our hypothesis, this chapter focuses on developing a framework that is built on the final module from the previous chapter to conduct an experiment. The goal of this experiment is to validate the hypothesis that the current quality of Machine Translation (MT) supports the creation of a Language Model (LM) pre-trained on an automatically translated corpus, while still remaining competitive with State-of-the-Art (SOTA) models. Using various statistical tests, a reporting system will be set up to evaluate the performance of each model across all datasets in our DrBenchmark's adaptation. After assessing the performance of TransBERT, CamemBERT, and DrBERT on each dataset, an aggregation will be deployed allowing statistical testing at the task level to deduce the competitiveness of TransBERT on genuine French downstream tasks.

4.1 Introduction

This section briefly outlines the context, motivation, and central hypothesis of the research, laying the groundwork for the experimental framework that follows.

4.1.1 Motivation

For many years, computers have been employed to support healthcare professionals in their research. As mentioned in our literature review, from classification to Information Retrieval (IR), life science researchers are actively enhancing their respective fields assisted by advancing technologies. Recently, Pre-trained Language Models (PLMs) significantly impacted Natural Language Processing (NLP), leading to numerous model variations based on different methodology, language and training corpus. In the meantime, the advent of Transformer-based models has greatly improved automatic translation, rendering it possible to translate extensive corpora in an efficient timeframe.

Given that training frameworks require data, Language Models (LMs) are predominantly centered around the English language. Although a few French models like CamemBERT exist, locating a Domain-Specific (DS) model trained in another language is generally challenging. Earlier this year, DrBenchmark (Labrak

et al., 2024) was launched, it includes a collection of French/Life sciences datasets that encompass classification, Named Entity Recognition (NER), Part-Of-Speech (POS), and Semantic Textual Similarity (STS) tasks. Despite the paper's mixed results regarding an overall superior model, their model, DrBERT, appears to be on of the most competitive.

4.1.2 Hypothesis

Taking into account the limited availability of life science data in French and the advancements in automatic translation along with DrBenchmark's publication, it becomes feasible to (1) translate a substantial life science dataset, (2) train a LM entirely on synthetically translated data, and (3) test and evaluate this model against other models such as DrBERT. The goal of this chapter is to address our first hypothesis, which is:

> **The current state of Machine Translation (MT) enables the development of a Language Model (LM) trained entirely on an automatically translated corpus, maintaining competitiveness with State-of-the-Art (SOTA) models in the field.**

The subsequent section will describe the experimental framework formulated to rigorously evaluate our hypothesis.

4.2 Experimental Setting

This section provides an overview of the experimental framework constructed to thoroughly assess the hypothesis detailed in the Introduction. It contains a brief summary of the methodology, model comparison, statistical testing, and reporting strategies used in the research. The section is organized to guide the reader through the experimental process, from selecting models to interpreting results, ensuring a full grasp of the research approach.

4.2.1 Model Comparison

To evaluate our hypothesis with competitive models, we decided the top performing models of each kind, a general French model, to see at least how our model compares with a general model and a DS model. Given that the general French models produced similar results in DrBenchmark, we selected the most downloaded one, CamemBERT (Section 2.4.3.3). For the DS model, the highest performing one, DrBERT (Section 2.4.3.4), was picked. In the previous chapter, TransBERT, our LM entirely trained on synthetic translated data, was developed. TransBERT was compared to two State-of-the-Art (SOTA) models for two main reasons: (1) adding another model would significantly increase computation time, and (2) there is no evidence suggesting that an additional model would outperform these two SOTA models.

4.2.2 From Fine-Tuning to Results

Each model will be fine-tuned over five distinct rounds or iterations as described in Chapter 3. It is important to highlight that metrics are only evaluated at the fold level and are not combined across different rounds, as doing so would compromise our independence assumption required for statistical purposes. To establish this essential framework, for every tested dataset, fold, or observation, the combined predictions from all rounds for a particular model will be considered. This means predictions will be averaged over the rounds. Metrics will be calculated at the fold level and can then be aggregated with other metrics to derive new measurements or used for statistical analysis.

For instance, metrics such as macro and weighted averages are composite measures derived from the performance of a model's label on a specific dataset fold. Since each dataset is divided into five folds, it will generate five metric sets per dataset. Thus, for every label, there will be five metrics, and if a fold has several labels, there will also be five micro, macro, and weighted averages. Treating each fold separately is crucial for the later inter-dataset testing, which results in somewhat unusual reporting. Although this method differs from the conventional approach in the community, where folds are typically averaged or concatenated, independence across fold is considered essential for validating a statistical test that could validate our chapter hypothesis.

4.2.3 Statistical Testing

As described in Section 3.4.1.1, the current setup of DrBenchmark has been modified to increase the data range available. By employing a 5-fold cross-validation approach for each dataset, the metrics for each dataset will essentially be calculated five times. By assuming each dataset's fold as being independent, this results in models being assessed across an effective total of 75 datasets (5 folds $\times$ 15 datasets). This separation aids in performing statistical tests as detailed in Section 2.1.1.8. Various statistical tests will be carried out in the following sections, taking each dataset and fold into account.

During the prediction phase for tasks such as classification, NER, and POS tagging, we will assess the statistical significance for each distinct class/label, named entity, and POS tag. To statistically evaluate model differences in a multi-class/label scenario, we need to treat each class/label as a separate binary problem. After binarization, each class/label will be analyzed using the commonly employed McNemar test to detect disparities between models. Still at the fold level, this binary test will also be applied for the micro-average metric, where all labels are treated as binary. As stated in Section 2.1.1.8, since the McNemar test is a pairwise test, a Bonferroni correction will be applied.

For the STS task, we will analyze the residuals to check for consistency across various models. Given that residuals typically follow a normal distribution, the repeated measures Analysis Of Variance (ANOVA) will be performed on the residuals. If the hypothesis that all groups are identical is rejected, a paired t-test with Bonferroni correction will be conducted. Given that normality and equal variances cannot be assumed, the Shapiro test and Levene test will be performed accordingly. Should these assumptions be compromised, the Friedman test will be performed, followed by the Nemenyi test for pairwise comparisons.

Until now, the only test purpose is to identify differences among models in a specific dataset fold, which can only lead to the conclusion that a model behaves differently. The determination of it being superior is achieved through metric measurement. When a model achieves the best metrics with statistical significance, it can be deemed the best and attributes its success to its features rather than randomness. To draw a comparison, it is akin to winning a 100-meter sprint by 0.1 seconds versus several seconds. However, although obtaining statistical significance for a particular class within a dataset is challenging, it only shows excellence in a specific context. Continuing with the analogy, if an athlete wins all of its races by 1/10 second, at one point, this pattern cannot be attributed to randomness. In other words, excelling in one dataset doesn't prove much, but comparing models across various datasets provides a more accurate representation of their overall performance.

According to (Demšar, 2006), employing the Friedman test succeeded by the Nemenyi test is a recommended practice for comparing metric rankings to assess model differences over multiple datasets. Hence, for these tests, three levels of abstraction can be evaluated: (1) at each fold, by comparing the metrics for each label that make up both the macro and weighted averages, (2) at the dataset level, by comparing the metrics for each label of each fold that make up the macro and weighted averages averaged across all folds, and (3) at any logical level of aggregation, such as the weighted aggregation of all binary classification F_1 scores, which will be covered in Section 4.4. This approach not only provides a metric to compare models but also statistically demonstrates differences between models, thereby rejecting the idea that metric values are due to randomness.

To summarize, at the dataset level, three elements will not undergo statistical testing: (1) each label metric averaged across folds, (2) the micro average averaged across folds, and (3) the R^2 averaged across folds. The reason for this is straightforward: while it would be insightful to test the averaged metrics, the Friedman test would not yield conclusive results with only five values. An alternative approach, such as using the McNemar test or combining STS testing on the concatenated folds, was dismissed because it would essentially involve testing a different metric. Indeed, although concatenating folds might have allowed for statistically significant testing, it would not pertain to the metric we are evaluating. In essence, considering each fold as an independent dataset implies that testing averaged metrics across five values is not viable.

4.2.4 Reporting

To reflect independence across folds, while avoiding table repetition per fold, a special reporting that takes into account each dataset common features will be introduced. Table 4.1 provides an example of how to capture each model's performance for each fold in a single table preserving dataset's label/metric uniformity.

Each value in the table represents the corresponding metric averaged over all folds, with bold text indicating the first place and underlined text indicating the second place. Although mainly indicative, this value reflects a key distribution parameter: the mean. Additionally, to provide a quick overview of per-fold performance, a medal reward system has been adopted. This system assigns a gold medal to the top model for each metric/label pair, a silver medal to the second-best,

and a bronze medal to the third. Similar to the Olympic Games, in the event of a tie, the highest rank is assigned to each model sharing that tie, and the subsequent rank takes into account the number of ties. For instance, if two models tie for first place, the following model is ranked third. Red medals indicate a tie with a metric of 0.00. From left to right, the first medal represents the first fold, and so forth.

Classes	**CamemBERT**			**DrBERT**			**TransBERT**			**Support**
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Neoplasms	85.41	90.12	87.50	84.62	**91.36**	87.83	**89.34**	91.09	**90.10**	242
...										
Blood Disorders	0.00	0.00	0.00	**20.00**	10.00	13.33	16.67	**30.00**	**21.33**	7
Macro avg	45.25	46.95	43.64	52.60	49.30	49.33	**60.77***	**60.21**	**58.34***	726
Micro avg (Accuracy)	←	65.98	→	←	70.09	→	←	**75.88**	→	726

Table 4.1: Example of a Dataset Detailed Model Evaluation - This is a reporting example using a modified table of DiaMed.

Pastel colors provide insights into the rankings for a specific metric or label, while vibrant colors indicate ranks with statistical significance. As discussed in the previous section, various levels of testing will be conducted, each corresponding to its suitable test. At the label level, the McNemar test is used to assess each fold by comparing the models binary predictions with the true labels. Still at the fold level, this test is similarly applied for micro averaging or accuracy in this example. Both macro and weighted averages are evaluated at two levels: (1) indicated by vibrant colors at the fold level, and (2) marked with an asterisk across folds to highlight the given metric. Both levels are tested using the Friedman test followed by the Nemenyi test; (1) compares a specific metric for each label within a fold, while (2) applies the same test across all folds.

At first, this reporting system might seem hard to interpret, but it thoroughly details and visualizes each metric well. Let's focus on Table 4.1 and examine the insights one can draw with and without the medal system. Looking at CamemBERT's Recall for the "*Neoplasms*" category, the raw figures suggest it has the lowest metric among all models, indicating it performs the worst in retrieving "*Neoplasms*" sequences. However, when considering the medals, it becomes evident that CamemBERT achieved the highest Recall in three out of five folds but the lowest in the remaining two, probably by a larger margin. Since there was no major difference among the five medals, these variations were likely minor.

In terms of macro average, TransBERT achieves the highest F_1 score across every fold. Despite a considerable 9-point difference from the second-best model, none of the macro folds were statistically significant. This is likely because the test's nature is derived from the label metrics, which shows that the macro level significance is greatly influenced by the number of labels in the dataset. Fortunately, evaluating the same metric across various folds, encompassing a quintuple comparison of label metrics, ultimately confers statistical significance upon the aggregate measurement.

As disclosed in Section 4.2.3, the $F_{1_{micro}}$ averaged across folds will not be tested. In our example, TransBERT consistently ranks first, once achieving statistical significance. On the other hand, CamemBERT often ranks in the opposite position

ranking third, once with statistical significance.

As mentioned in Section 3.4.1.1, labels that are very sparse and do not appear at least once in each fold were excluded from the dataset reporting and statistical evaluation. Consequently, in certain cases of multi-class classification, the micro average may show different figures for Recall and Precision, when this is not the case, only one number will be displayed, which is an equivalent of the accuracy metric.

Since DEFT-2020/Task 2 is addressed in a way that makes any class aggregation a useless metric, the main metric used will be $F_{1_{micro}}$ for this dataset. To maintain consistency, Table 4.2 will use this same metric for comparing all datasets, including classification, NER, and POS. Given that the only meaningful metrics for the entire dataset, which get statistically tested at the dataset level, are the macro and weighted averages averaged across the folds, many dataset conclusions will rely on those metrics. In reports such as Table 4.2, a Confidence Interval (CI) can be provided for information purposes, as the normality is never implied.

4.3 Model Performance Overview

In the subsequent sections, each task will be individually evaluated to ascertain whether any model offers unique knowledge supporting task completion. It is important to highlight that all models possess the same architecture and undergo fine-tuning via an identical methodology, differing solely in the pre-training phase. As a Table of Content of the results, with link going back and forth each section, Table 4.2 illustrates the main task metric for each dataset, averaged over five folds for each model. A 95% CI is included to depict the variability of performance across folds, only for information purposes, as normality will not be assumed in our statistical tests. For ease of navigation, links to detailed results sections for each task are included within the table, and each section begins with a reference to this table.

As an overview of the detailed task analysis provided in subsequent sections, this introduction will briefly highlight the findings presented in Table 4.2. Classification and NER are the most represented tasks in our benchmark adaptation, each with five and six datasets, respectively, and are likely the most popular tasks in the life sciences field as it is used in various applications. Meanwhile, POS and STS are each represented by two datasets. The table presents the main metrics for each task which are the $F_{1_{micro}}$ for POS, NER and CLS tasks and the R^2 score for STS task.

Examining classification tasks, CamemBERT achieves the highest average main metric across folds in three of the five datasets, whereas TransBERT excels in two. However, evaluating from a fold-wise perspective, CamemBERT, DrBERT, and TransBERT rank first in 6, 2, and 18 instances out of 25 folds, respectively, with a tie in a DEFT-2020/Task 2 fold. For the NER task, the main metric averaged over folds is led by TransBERT, while a tie is observed in the POS tagging datasets. In the two STS datasets, while the average R^2 across folds seems dominated by CamemBERT, an insight into fold-wise ranking shows a tie, with both models achieving the best scores three and two times for one dataset each.

Task	Dataset	CamemBERT	DrBERT	TransBERT
CLS	DEFT-2020/Task 2	**98.91±0.48**	97.55±1.09	98.82±0.96
	DiaMed	65.98±2.22	70.09±2.34	**75.88±1.96**
	FrenchMedMCQA	**60.22±2.54**	56.48±0.53	59.38±1.57
	MorFITT	73.55±1.04	73.30±0.60	**75.74±1.01**
	PxCorpus/Task 2	**96.47±0.71**	95.66±1.26	95.77±1.07
NER	E3C/Clinical	74.91±1.39	75.46±1.16	**76.83±1.25**
	E3C/Temporal	85.46±0.47	83.91±0.62	**85.73±0.59**
	MantraGSC/Merged	60.33±3.47	58.20±3.63	**63.24±3.28**
	PxCorpus/Task 1	93.63±1.19	93.31±2.46	**95.26±0.73**
	QUAERO/EMEA	84.87±0.90	84.86±0.50	**85.72±0.59**
	QUAERO/Medline	62.38±0.84	60.97±0.86	**64.29±0.95**
POS	CAS	97.69±0.23	97.58±0.08	**97.76±0.22**
	ESSAI	**98.67±0.04**	98.55±0.04	98.65±0.03
STS	CLISTER	**82.80±1.86**	75.44±1.24	82.62±2.18
	DEFT-2020/Task 1	**83.95±3.21**	71.69±4.06	83.46±2.39

Table 4.2: Summary of Model/Dataset Results - The table shows the main metric of each task averaged across for the 15 datasets. The main metrics are the $F1_{weighted}$ for POS, NER and CLS tasks and the R^2 score for STS task. Bold and underline formatting are used to highlight the best and second-best results, respectively.

4.3.1 Classification Task

In this section, each of the five classification tasks will be thoroughly examined. For each task, after exploring the specifics of the data using tools such as Precision/Recall curves, confusion matrices, or class/label-specific performance, when deemed necessary, a conclusion will be provided.

4.3.1.1 DEFT-2020/Task 2

As explained in Section 3.4.3.1.1, the current task involves determining which of the three sentences is most similar to a given source sentence. The variable that indicates the sentence number acts as a placeholder, making weighted and macro-aggregations irrelevant. The key metric used is $F1_{micro}$, which measures accuracy in this multi-class setting. Table 4.3 outlines average fold accuracy, with CamemBERT leading by 0.09 points. Across all folds, the models' accuracy spanned a narrow range between 95.91% and 100.00%, underscoring their competitiveness. DrBERT shone in the third fold, achieving the highest performance, but was the least effective in the other four folds, illustrating how tightly grouped model performances make it possible to move from the lowest to the highest position. In four folds, CamemBERT ranked second-best and shared the top position with TransBERT, which held the top spot four times, despite finishing last in the third fold. With such a small room for improvement, there were no space for statistical significance. This dataset illustrates that poor performance in a single fold can decrease the overall metric average. Indeed, even without achieving the top score in any individual fold (once with a tie), CamemBERT attained the highest average $F1_{micro}$.

Classes	CamemBERT			DrBERT			TransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Micro avg (Accuracy)	←	**98.91**	→	←	97.55	→	←	<u>98.82</u>	→	1,100

Table 4.3: Detailed Model Evaluation for DEFT-2020/Task 2 - The table shows model main metric averaged over all five folds. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. No statistical significance in this dataset. For easy navigation, Table 4.2 shows the main results.

As described in Section 3.4.3.1.1, given that the DEFT-2020 dataset is unique in mixing life science content with encyclopedia source, a manual review uncovered a substantial presence of non-life science data. Since the primary research goal is assessing a life science DS LM, and the accuracy is fairly high on this 1,100 instances dataset, we manually curated all misclassified cases to either life science or encyclopedia categories. Figure 4.1 depicts the number of instances where each model made at least one error across all five training rounds. It is worth nothing that because the total misclassifications are calculated over all five rounds, these figures do not reflect overall accuracy, the selection as only been made in an effort to see the life science proportion of misclassification. Taking a look at the figure, it's noteworthy that approximately 94% of CamemBERT's errors pertain to life science instances, while TransBERT and DrBERT exhibit lower rates of about 84% and 82%, respectively. For illustration, an example of a misclassification outside the life sciences domain is provided in Figure G.5.

In summary, the second task of DEFT-2020 does not reveal any model that is definitively the best or worst in statistical terms. TransBERT achieves the highest main metric in four folds, as opposed to only once for the other two models, but has a marginally lower average main metric across folds. Conversely, DrBERT, which has the lowest average metric, shows the poorest performance in four out of five instances. However, since no statistical significance is found, no definitive differences among the models for this dataset can be concluded, with the average performance difference spanning 1.36 points. Additionally, for this specific dataset utilizing non-life science data, around 17% of misclassifications are related to encyclopedia topics. Examining misclassifications within models indicates that CamemBERT's misclassifications are 94% related to life sciences, whereas TransBERT and DrBERT exhibit a significantly lower rate of about 83%.

4.3.1.2 DiaMed

DiaMed involves categorizing clinical cases into one of 15 possible classes. Table 4.4 presents the Precision, Recall, and F_1 per class, along with micro, macro, and weighted average aggregations.

Focusing on the class level, TransBERT consistently outperformed the other models in most classes, achieving the top scores in Precision, Recall, and F_1 for the majority of the categories, with statistical significance for "*Cong. Malform.*". Achieving mostly second-place results across various classes, DrBERT appears to be the next best model, frequently providing strong performances, especially in

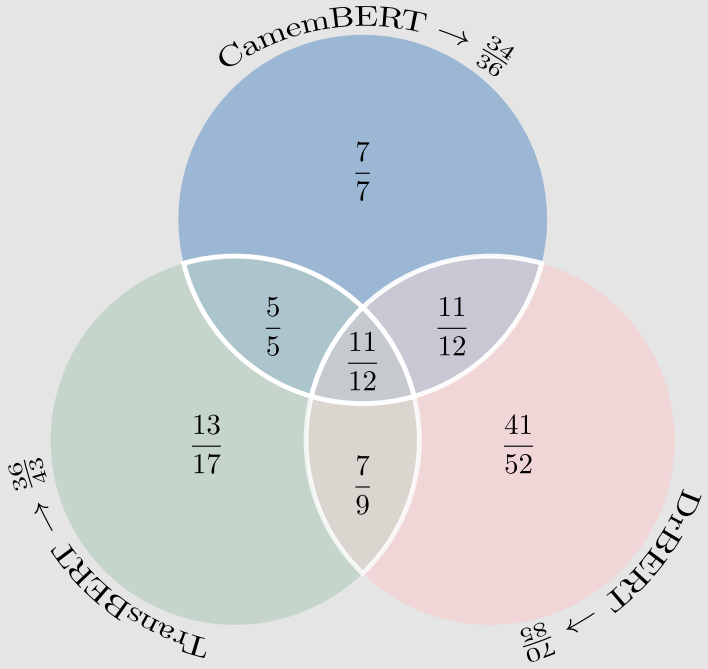

Figure 4.1: DEFT-2020/Task 2 - Error Analysis Venn Diagram - This figure illustrates the number of instances where each model made at least one misclassification across all the five training iterations. The ratio represents the proportion of life science misclassifications. As this dataset's source is blent with encyclopedic data, the highest the life science proportion misclassification, the less effective a model is at classifying what is relevant for this research.

the "*Neoplasms*" and "*Musculoskeletal*" categories. It was recognized for having the highest Recall in several cases, highlighting its strength in identifying relevant instances. In contrast, CamemBERT finishes last in the majority of the categories, except for "*Infectious*" where it secures the second position, while ranking first in the final two folds.

Looking at the accuracy, CamemBERT typically underperformed compared to the other two models. Although it never statistically ranked last at the label level, it significantly achieved the lowest $F_{1_{micro}}$ score in the first fold. In contrast, TransBERT achieved the highest accuracy across all folds, with statistical significance in the second fold, where it also had significance in classifying "*Cong. Malform.*". It is important to note that some of the less supported classes had multiple folds where the models failed to perform, resulting in an accuracy of 0.00.

Figure 4.2 displays the confusion matrix for the two best-performing models, revealing that DrBERT's predictions are heavily skewed towards the most common class "*Neoplasms*". This tendency explains the high Recall for this class, coupled with poor Precision. On the other hand, TransBERT's confusion matrix shows that misclassifications are mainly restricted to the three least represented classes: "*Nervous*", "*Respiratory*", and "*Blood Disorders*", with 13, 10, and 7 instances,

Classes	CamemBERT			DrBERT			TransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Neoplasms	85.41	90.12	87.50	84.62	**91.36**	87.83	**89.34**	91.09	**90.10**	242
Infectious	74.64	81.15	77.02	72.38	82.29	75.40	**78.34**	**85.44**	**81.34**	89
Injury	75.18	61.06	66.43	70.16	68.98	69.18	**78.40**	**83.86**	**80.32**	74
Cong. Malform.	43.00	32.29	36.08	46.66	39.64	41.99	**70.92**	**61.89**	**63.57**	55
Musculoskeletal	57.71	64.01	60.00	58.66	**70.53**	63.79	**67.74**	68.68	**65.86**	52
Circulatory	47.64	46.15	46.36	64.89	56.31	58.32	**73.81**	**58.27**	**62.16**	43
Digestive	48.57	**66.94**	54.16	**68.83**	56.94	58.63	58.11	61.94	**59.17**	34
Endocrine	43.33	31.33	33.71	51.67	32.67	38.70	**65.95**	**72.67**	**66.39**	24
Pregnancy	63.71	57.33	51.87	72.00	69.00	69.16	**78.67**	**81.00**	**77.11**	23
Eye	51.24	60.33	54.34	54.50	**63.67**	**57.71**	**58.33**	59.67	57.67	21
Genitourinary	41.90	46.57	38.33	48.00	38.43	41.67	**79.33**	**70.57**	**70.78**	20
Skin	41.33	47.00	40.76	56.67	49.67	50.85	**66.00**	**51.33**	**55.97**	19
Nervous	5.00	**20.00**	8.00	0.00	0.00	0.00	**10.00**	**20.00**	**13.33**	13
Respiratory	0.00	0.00	0.00	**20.00**	**10.00**	**13.33**	**20.00**	6.67	10.00	10
Blood Disorders	0.00	0.00	0.00	**20.00**	10.00	13.33	16.67	**30.00**	**21.33**	7
Weighted avg	66.17	65.98	64.70	69.87	70.09	68.86	**77.39***	**75.88**	**75.31***	726
Macro avg	45.25	46.95	43.64	52.60	49.30	49.33	**60.77***	**60.21**	**58.34***	726
Micro avg (Accuracy)	←	65.98	→	←	70.09	→	←	**75.88**	→	726

Table 4.4: Detailed Model Evaluation for DiaMed - The table shows model metrics averaged over all five folds for the 15 classes. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table F.2 shows the task statistics and Table 4.2 the main results.

respectively, across the entire dataset. As outlined in Table 4.4, these classes amassed multiple 0.00 metrics, marking them as the lowest-performing classes across models. The only non-diagonal entries for TransBERT indicate that the model tend to confuse "*Respiratory*" with "*Injury*" and "*Nervous*" with "*Musculuoskeletal*".

The analysis of the DiaMed results table reveals that TransBERT stands out as the most effective model, demonstrating superior performance in various medical categories and all aggregated measures of Precision, Recall, and F_1 metrics. It persistently achieves the best results in each fold on the main metric, with one instance of statistical significance. This consistency across multiple classes and folds has been recognized with a statistically significant overall best performance, as

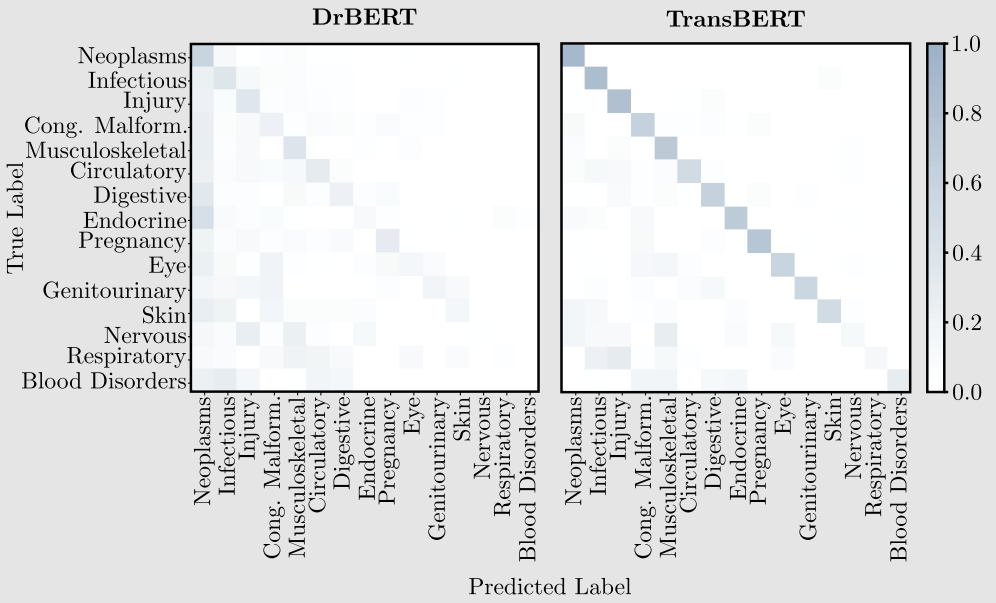

Figure 4.2: DrBERT & TransBERT - Confusion Matrices for DiaMed - To plot this confusion matrix, the test sets from all folds were combined, providing better insight into the overall class confusion for a given model.

evidenced by the weighted and macro averages computed across all folds. In contrast, CamemBERT frequently underperforms, often delivering the worst outcomes in several classes and emerging as the least effective model four out of five times, once with statistical significance. Summarizing, the comparison of models on this dataset reveals a distinct overall best-performing model, and another that is notably worse in performance. By inference, and without any statistical differentiation, DrBERT consistently places in the middle, without notably excelling or underperforming in any specific area.

4.3.1.3 FrenchMedMCQA

The FrenchMedMCQA dataset entails determining the count of correct responses in a Multiple-Choice Question Answering (MCQA) dataset. As indicated by Table 4.2, CambemBERT achieves the highest average score, being 0.84 points ahead of TransBERT, which is in the second place, and also shows the largest CI in the classification task. DrBERT, trailing 2.9 points behind the runner-up, falls short in this overall average dataset metric.

Referring to Table 4.5, CamemBERT excels in predicting single correct answers, achieving the highest F_1 score for the most supported class across all five folds. Although it attains the best Precision for identifying "*2 Correct Answers*" questions with significance at one fold, it can also perform the worst in retrieving them, displaying the lowest Recall and F_1 with statistical significance at one fold. Moreover, DrBERT yields the worst results among the top-3 most supported classes, recording the lowest metric for predicting "*2 Correct Answers*" at the second fold with statistical significance. TransBERT, on the other hand, performs moderately,

securing the highest F_1 on four folds for "*2 Correct Answers*" while ranking second for the other top-3 supported classes on the same folds. With an F_1 of 20.21, DrBERT becomes the best "*4 Correct Answers*" classifier with significance on one occasion. All models face difficulties in identifying "*5 Correct Answers*" except for the fourth fold where DrBERT identified a few instances.

Classes	CamemBERT			DrBERT			TransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
1 Correct Answer	**97.09**	89.59	**93.17**	93.02	89.62	91.28	93.19	**90.46**	91.79	1,079
2 Correct Answers	**37.77**	26.53	28.39	35.66	36.76	35.82	36.38	**37.43**	**35.90**	670
3 Correct Answers	45.98	**74.02**	**56.33**	42.99	50.76	46.47	**47.65**	63.19	53.74	929
4 Correct Answers	19.05	8.47	11.52	**25.61**	**17.29**	**20.21**	21.49	9.25	12.64	381
5 Correct Answers	0.00	0.00	0.00	**10.00**	**2.00**	**3.33**	0.00	0.00	0.00	43
Weighted avg	**58.11**	**60.22**	56.94	56.37	56.48	56.01	57.24	59.38	**57.25**	3,102
Macro avg	39.98	39.72	37.88	**41.45**	39.29	**39.42**	39.74	**40.07**	38.82	3,102
Micro avg (Accuracy)	←	**60.22**	→	←	56.48	→	←	59.38	→	3,102

Table 4.5: Detailed Model Evaluation for FrenchMedMCQA - The table shows model metrics averaged over all five folds for the 5 classes. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table 4.2 the main results.

The aggregated metrics reveal an interesting comparison, with each of the three models excelling in different aspects. CamemBERT achieves the highest average Precision and Recall when class weights are taken into account, whereas TransBERT secures the best $F_{1_{weighted}}$. If equal weight is assigned to each class, DrBERT obtains the top $F_{1_{micro}}$ twice. While perspectives may vary, it is crucial to note that, for the main metric, , despite the absence of a definitive best model on this dataset, TransBERT delivers the best accuracy across most folds, while CamemBERT reaches that level twice with the highest average accuracy, finally, DrBERT performs the worst in four out of five instances, twice with statistical significance.

4.3.1.4 MorFITT

MorFITT is a dataset designed to categorize biomedical abstracts into 12 distinct labels. As delineated in Table 4.2, a 2.19 points disparity is observed between TransBERT and CamemBERT, identified as the highest and second-highest performing models concerning the averaged $F_{1_{micro}}$, respectively.

Table 4.6 provides a detailed comparison of the three models' performance across all 12 of MorFITT's labels. Examining the bold metric by label, DrBERT

has the highest Precision for most labels but also the lowest Recall for 10 of them. CamemBERT exhibits the lowest Precision in 11 out of 12 labels, TransBERT obtains the best Recall and F_1 scores in 11 and 12 labels, respectively. In contrast to earlier datasets, this consistency in label performance is partially due to the substantial support, even for the less frequently occurring labels. Analyzing the F_1 score for each label per fold, TransBERT achieved the highest score approximately 63% of the time, whereas DrBERT did so around 13%. This likely explains why both macro and weighted averages are predominantly characterized by five top rankings.

Labels	**CamemBERT**			**DrBERT**			**TransBERT**			**Support**
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Veterinary	80.12	88.59	84.10	**81.80**	85.63	83.61	81.02	**90.54**	**85.49**	824
Etiology	63.95	**70.71**	67.05	67.26	63.63	65.34	**69.38**	68.75	**68.90**	741
Psychology	84.04	87.54	85.67	84.47	86.90	85.59	**85.60**	**87.67**	**86.58**	608
Surgery	79.72	86.43	82.91	**81.63**	85.82	83.61	81.58	**86.84**	**84.04**	549
Genetics	**77.09**	76.33	76.52	76.89	75.00	75.83	75.41	**78.91**	**77.04**	505
Physiology	67.12	51.83	58.27	64.28	47.95	54.89	**68.57**	**54.10**	**60.36**	490
Pharmacology	67.44	65.61	66.34	**73.94**	60.48	66.23	70.18	**69.41**	**69.45**	299
Microbiology	69.18	70.63	69.31	**71.87**	72.29	71.44	71.36	**76.24**	**73.53**	273
Immunology	64.86	63.62	62.94	**69.96**	60.40	64.49	68.09	**67.43**	**67.21**	262
Chemistry	67.67	46.69	54.00	65.08	47.71	54.39	**69.80**	**54.88**	**60.57**	212
Virology	69.61	70.92	70.11	**72.17**	69.02	70.29	69.95	**73.59**	**71.25**	200
Parasitology	60.97	64.78	61.78	65.58	62.18	63.36	**69.21**	**75.71**	**72.27**	152
Weighted avg	73.30*	74.11*	73.16	74.87	71.48*	72.74	**75.32**	**76.12***	**75.36***	5,115
Macro avg	70.98*	70.31*	69.92	72.91	68.09*	69.92	**73.35**	**73.67***	**73.06***	5,115
Micro avg	73.13	74.11	73.55	75.26	71.48	73.30	**75.38**	**76.12**	**75.74**	5,115

Table 4.6: Detailed Model Evaluation for MorFITT - The table shows model metrics averaged over all five folds for the 12 labels. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table F.7 shows the task statistics and Table 4.2 the main results.

Figure 4.3 presents the Precision/Recall curves of TransBERT for each label, along with the micro and macro averages. The categories of Veterinary and Psychology excel with high Area Under the Cruve (AUC) of 86.37 and 86.16 respectively, demonstrating a strong balance between Precision and Recall. Conversely, categories such as Chemistry and Physiology exhibit lower performance, with AUC

values of 56.78 and 60.63. The macro average and micro average, with respectively AUC of 71.99 and 75.56 offer an overall view of performance, with the micro average indicating slightly better performance when assessing all instances equally. Most of the curves show that Precision is well-maintained as Recall increases, particularly for the highest-performing categories. However, categories like Chemistry and Physiology experience a quicker drop in Precision as Recall grows, highlighting difficulties in sustaining accuracy while capturing all positive instances in these fields. In contrast to previous tasks, the labels with fewer instances do not exhibit the lowest performance results. It is, however, noteworthy that in absolute terms, the label with the least support, "*Parasitology*", comprises 152 instances, whereas DiaMed's class "*Blood Disorders*" is represented by a mere 7 instances.

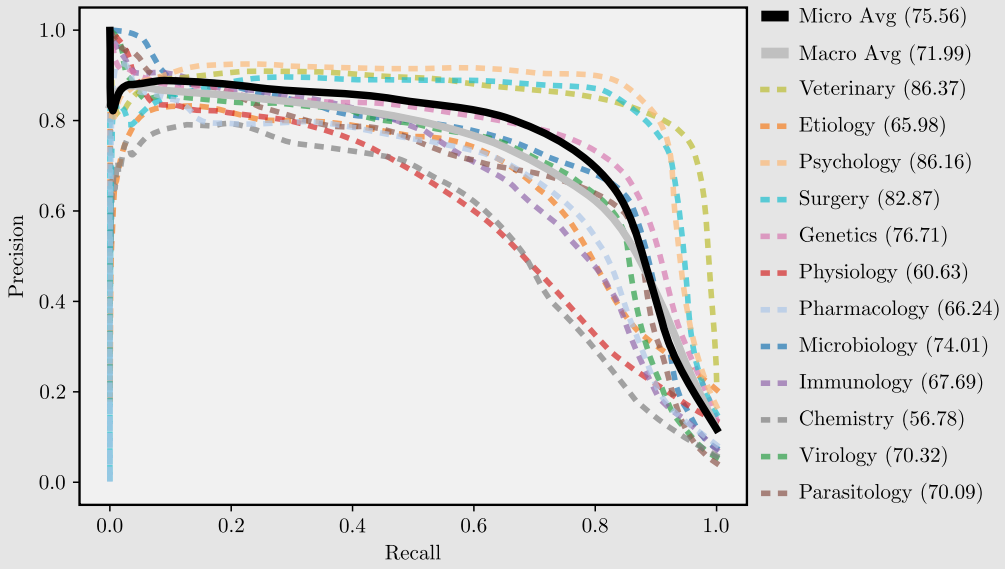

Figure 4.3: TransBERT - Precision/Recall Curves for MorFITT - To provide a broader perspective, the micro and macro averages are displayed as solid lines, and individual labels are indicated with dashed lines. The numbers in parentheses denote the area under each respective curve.

In summary, TransBERT achieves the highest weighted and macro averages for Recall and F_1 scores across folds, with statistical significance, by obtaining the highest averaged Recall and F_1 scores for 11 and 12 labels, respectively. Despite being the most precise model on average and in three folds, it does not achieve statistical significance for Precision. Conversely, it's challenging to determine the second-leading model between CamemBERT and DrBERT. CamemBERT statistically outperforms DrBERT in Recall averaged across folds, yet it is statistically the least precise model with statistical significance aggregated and in one given fold.

4.3.1.5 PxCorpus/Task 2

The PxCorpus's second task includes a dataset of medical drug prescription transcripts that were manually categorized into four classes, with a considerable imbalance as the most prevalent class comprises roughly 91% of the dataset. Table 4.2 shows the main metric averaged across folds where CamemBERT, TransBERT, and DrBERT rank in first, second, and third places, respectively. Nonetheless, the narrow metric range of 0.81 points illustrates a very competitive dataset.

Table 4.7 presents the dataset outcomes. In the most supported class, "*Medical Prescription*", CamemBERT achieves the highest Precision and F_1 scores on average across folds, and it secures the top rank in Recall the most frequently. The "*None*" class highest ranks are shared by CamemBERT and DrBERT, while TransBERT achieves the highest score only in the first fold. The second most supported class is largely dominated by CamemBERT, which ranks first in at least three different metrics and shows statistical significance in one of them. In the same fold, it obtains the best Recall and F_1 in four and three classes, respectively. Additionally, CamemBERT achieves the best performances in the two least supported classes, where a few 0.00 values among models can be spotted.

Classes	**CamemBERT**			**DrBERT**			**TransBERT**			**Support**
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Medical Prescription	**97.87**	99.05	**98.45**	97.49	98.67	98.07	97.02	**99.24**	98.11	1,574
None	**83.21**	**77.29**	**79.73**	75.28	72.98	73.67	80.77	68.59	73.84	115
Negate	66.00	**45.00**	**51.55**	65.00	41.67	50.48	**66.67**	41.67	50.48	21
Replace	**60.00**	**40.00**	**47.67**	41.67	34.67	35.24	36.67	18.67	24.43	17
Weighted avg	**96.39**	**96.47**	**96.31**	95.28	95.66	95.35	95.28	95.77	95.34	1,727
Macro avg	**76.77**	**65.33**	**69.35**	69.86	61.99	64.36	70.28	57.04	61.71	1,727
Micro avg (Accuracy)	←	**96.47**	→	←	95.66	→	←	95.77	→	1,727

Table 4.7: Detailed Model Evaluation for PxCorpus/Task 2 - The table shows model metrics averaged over all five folds for the 4 classes. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table F.9 shows the task statistics and Table 4.2 the main results.

Achieving the top scores on the majority of folds across all metrics and classes, CamemBERT achieves three Recall and F_1 values for each aggregation method, with statistical significance in one instance. In contrast, both DrBERT and TransBERT secure the remaining top scores in a single fold across each metric's highest aggregation, lacking statistical significance.

4.3.2 Named Entity Recognition Task

An examination of Table 4.2 reveals that the analysis exhibits a consistent pattern with TransBERT demonstrating superior performance across all NER tasks. This section will scrutinize each task result independently by assessing Precision, Recall, and F_1 metrics for each entity, taking into account the balance and origin of each dataset.

4.3.2.1 E3C/Clinical

In this first NER dataset, which focuses on identifying a single entity called "*Clinical Entity*", Table 4.2 shows that TransBERT leads in performance across four out of five folds for all metrics while getting the highest metrics averaged across folds. No statistical significance is noted at the entity level, and the singularity of the dataset prevents testing at any aggregation level. CamemBERT appears to underperform with the lowest F_1 score averaged across folds and ranking lowest in four out of five folds. Once again, since no statistical significance is observed in the task, any performance improvements could be attributed to random events.

Named Entities	**CamemBERT**			**DrBERT**			**TransBERT**			**Support**
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Clinical Entity	74.75	75.08	74.91	<u>75.50</u>	<u>75.44</u>	<u>75.46</u>	**76.80**	**76.89**	**76.83**	3,270

Table 4.8: Detailed Model Evaluation for E3C/Clinical - The table shows model metrics averaged over all five folds for the only named entity. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table F.3 shows the task statistics and Table 4.2 the main results.

4.3.2.2 E3C/Temporal

Originating from a clinical case as well, E3C/Temporal involves the identification of five distinct time-related named entities. An initial examination of Table 4.2 reveals that TransBERT, CamemBERT, and DrBERT rank first, second, and third when considering the primary metric averaged across folds, with a gap of 0.27 points separating the top two positions. In this dataset, the most represented entity accounts for approximately 66% of the instances, with the remaining four entities being comparatively uniformly distributed.

Analyzing Table 4.9, high performance appears to be divided among entities between CamemBERT and TransBERT. Specifically, TransBERT achieves the best outcomes in the two most prevalent entities and also in the "*Time Expression*" category, whereas CamemBERT excels in "*Lab Result*" and "*Actor*". Despite the absence of a statistical test differentiating the two models best performing models,

DrBERT ranks among the lowest for almost each entity and fold, with statistical significance at one instance on "*Body Part*".

Named Entities	CamemBERT			DrBERT			TransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Event	86.13	88.27	87.18	85.91	87.13	86.50	**86.83**	**89.04**	**87.91**	3,836
Body Part	75.60	76.22	75.80	70.74	72.50	71.48	**75.87**	**76.66**	**76.14**	654
Lab Result	**80.19**	**85.44**	**82.70**	77.75	82.53	80.04	79.03	82.76	80.83	507
Actor	**89.98**	**92.05**	**90.99**	88.64	89.20	88.92	88.94	91.45	90.13	426
Time Expression	**79.88**	82.53	81.09	76.10	79.43	77.68	79.00	**84.07**	**81.43**	333
Weighted avg	84.39	86.60	85.45	83.16*	84.79*	83.93*	**84.63**	**86.94**	**85.74**	5,756
Macro avg	**82.36**	**84.90**	**83.55**	79.83*	82.16*	80.92*	81.93	84.80	83.29	5,756
Micro avg	84.36	86.60	85.46	83.06	84.79	83.91	**84.57**	**86.94**	**85.73**	5,756

Table 4.9: Detailed Model Evaluation for E3C/Temporal - The table shows model metrics averaged over all five folds for the 5 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table F.4 shows the task statistics and Table 4.2 the main results.

In conclusion, it appears that even though they distribute the best outcomes between themselves, no model was consistent in any metric across all entities for a single fold, thus, while being better than DrBERT, we can't tell which is the best among CamemBERT and TransBERT. On the other hand, by being consistent with its ranking, DrBERT attains statistical significance in every aggregated metric averaged across folds. Additionally, it achieves the poorest results in each fold for each metric of the micro aggregation, once with statistical significance.

4.3.2.3 MantraGSC

MantraGSC/Merged represents a NER dataset that centers on biomedical abstracts/titles, drug labels, and patents. It possesses the smallest dataset size among all others, which is probably why Table 4.2 CIs suggests the highest expected variability. An overview of our summary results table indicates that TransBERT outperforms CamemBERT by 2.91 points, with CamemBERT ahead of DrBERT by 2.13 points.

Examining Table 4.10 at the entity level indicates that some metrics have fluctuated significantly across folds. Specifically, in the cases of "*Disorder*" and "*Chemical/Drugs*", the two most dominant classes, although CamemBERT frequently achieves the top rank with statistical significance, TransBERT attains the highest scores for all metrics averaged across folds. On the other hand, DrBERT is ranked lowest for approximately half of the entities in each metric.

Named Entities	CamemBERT			DrBERT			TransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Disorders	66.06	61.29	63.36	63.04	64.18	63.57	**66.74**	**65.01**	**65.73**	288
Chemical/Drugs	63.00	66.93	63.78	59.29	61.88	59.78	**64.36**	**69.78**	**66.52**	236
Procedures	49.46	**59.15**	53.48	54.59	56.17	**54.46**	**54.84**	53.29	53.44	129
Living Beings	**74.64**	**74.82**	**74.39**	69.13	65.76	66.58	70.93	71.27	70.71	91
Anatomy	51.97	60.02	55.04	46.25	38.21	41.40	**69.57**	**61.90**	**65.18**	66
Physiology	30.78	**32.67**	**31.35**	**46.42**	22.44	26.86	29.27	24.44	26.22	44
Objects	**52.00**	19.33	24.38	46.67	31.33	31.71	42.00	**58.15**	**38.42**	25
Weighted avg	61.55	61.08	60.14	59.94	57.24*	57.38	**63.87**	**62.76**	**62.56**	879
Macro avg	55.42	53.46	52.25	55.05	48.57*	49.19	**56.82**	**57.69**	**55.17**	879
Micro avg	59.70	61.08	60.33	59.24	57.24	58.20	**63.82**	**62.76**	**63.24**	879

Table 4.10: Detailed Model Evaluation for MantraGSC/Merged - The table shows model metrics averaged over all five folds for the 7 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table F.6 shows the task statistics and Table 4.2 the main results.

In conclusion, the analysis of the most supported entities' results presents a perplexing picture. On one hand, CamemBERT achieves the best results with statistical significance on three separate occasions, while on the other hand, TransBERT consistently scores higher on the same metrics, although without any statistical significance. However, when examining the micro aggregation, the story becomes clearer. Specifically, while CamemBERT achieves the top Recall score twice, with statistical significance on one occasion, its performance in terms of aggregated F_1 scores is less impressive, achieving significance only twice when considering macro aggregation on this highly unbalanced dataset. In contrast, considering entity distribution shows that TransBERT achieves the highest F_1 scores four times across both micro and weighted averages, with statistical significance achieved in one fold. DrBERT lags behind in most folds, metrics, and aggregations, suffering notably from poor Recall, evidenced by the worst results in one fold for both weighted and macro aggregations. This consistency renders it the worst in aggregated Recall across folds, with statistical significance.

4.3.2.4 PxCorpus/Task 1

Comprising 30 named entities, PxCorpus/Task 1 is probably the most unbalanced dataset, with one-third of the entities representing close to 90% of the total

instances. In Table 4.2, the primary metric is led by TransBERT at 1.63 points above CamemBERT, the second best. DrBERT lags behind CamemBERT, with a 0.32 point difference and high CIs compared to the other two models.

With its 30 entities, Table 4.11 is quite extensive, but a few key points can effectively summarize each model's performance. TransBERT stands out with a clear majority of top scores across all metrics and folds, achieving the best results for most entities. Identifying the second best per entity is more complex. When looking at the F-1 score, which combines Precision and Recall, DrBERT exhibits high variability, with some of the lowest and highest statistically significant scores. However, CamemBERT shows strong results in several of the most supported entities when considering the aggregate performance across folds.

In conclusion, both CamemBERT and DrBERT exhibit a mix of statistically significant high and low results, making it challenging to assess their performance without aggregating the entities. Conversely, TransBERT consistently performs well, frequently achieving top rankings, though statistical significance is noted only twice. Aggregating results provides a clearer evaluation: TransBERT attains the best scores for both weighted and macro averages across all metrics and folds, with statistical significance. It also outperforms in micro metrics, securing the top spot three out of five times with statistical significance. DrBERT secures second place in weighted and macro average Precision across folds with statistical significance and tends to perform better in less supported entities compared to CamemBERT, showing an average over folds five-point higher $F_{1_{macro}}$ score. On a fold level, both models are the worst performers at least once per metric with statistical significance, but DrBERT ranks first twice, with one instance being statistically significant.

4.3.2.5 QUAERO/EMEA

QUAERO/EMEA is a NER dataset comprising 10 entities derived from texts related to marketed drugs. Table 4.2 shows that TransBERT outperforms CamemBERT, the second-best model, by 0.85 points and is just 0.01 point ahead of DrBERT.

Table 4.12 demonstrates the performance of models for each entity, with TransBERT achieving the highest F_1 score for 8 out of 10 entities. It is notably consistent in entities with low support, obtaining the best results in 53%, 93%, and 73% of the folds for Precision, Recall, and F_1, respectively. In contrast, both CamemBERT and DrBERT have diverse competitive results, with CamemBERT excelling in classifying "*Disorders*" and DrBERT in "*Physiology*", with one fold showing statistical significance.

TransBERT attains the highest values for all aggregated metrics averaged across folds. By achieving top Precision in 8 out of 10 entities, it achieves statistical significance in both aggregated averages. More importantly, by consistently attaining high F_1 scores across the entities, it reaches statistical significance with the highest weighted and macro averages averaged across folds. Without any statistical significance in any aggregation metric, there is not much difference between DrBERT and CamemBERT. However, it is noteworthy that DrBERT achieves substantially better $F_{1_{macro}}$, indicating more consistent F_1 scores across entities, being awarded four second places and one first place compared to five last ranks for CamemBERT in this specific metric.

Named Entities	CamemBERT			DrBERT			TransBERT			Support
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1	
dos_val	93.40	96.14	94.73	95.08	96.77	95.91	**96.08**	**97.07**	**96.56**	1,600
dos_uf	93.32	94.76	94.03	94.50	94.65	94.57	**96.24**	**96.24**	**96.24**	1,513
rhythm_tdte	99.05	99.79	99.42	98.72	99.73	99.22	**99.22**	**99.93**	**99.57**	1,320
dur_val	96.86	**99.67**	98.24	95.70	99.67	97.59	**98.13**	99.58	**98.85**	1,208
dur_ut	96.61	99.75	98.14	95.89	**99.75**	97.76	**98.06**	99.67	**98.85**	1,205
drug	89.10	**91.38**	**90.20**	86.62	88.15	87.36	**90.62**	88.81	89.67	935
d_dos_val	95.52	96.68	96.09	95.56	95.90	95.72	**96.05**	**97.03**	**96.53**	849
d_dos_up	**97.10**	**98.79**	**97.92**	96.76	97.60	97.18	96.96	98.78	97.86	822
inn	**82.60**	77.70	79.67	76.15	77.40	76.54	79.03	**85.84**	**82.07**	380
cma_event	**82.46**	81.46	**81.90**	77.59	76.73	77.13	78.42	**82.44**	80.33	313
d_dos_form	85.33	92.00	88.47	85.61	92.12	88.63	**90.21**	**93.90**	**92.00**	280
rhythm_perday	88.86	97.10	92.41	90.29	91.96	91.02	**95.02**	**97.15**	**95.91**	241
dos_cond	80.26	**87.24**	83.26	**89.13**	81.53	84.45	82.92	86.79	**84.51**	134
rhythm_hour	89.56	96.40	92.69	89.01	87.27	88.00	**95.20**	**98.00**	**96.46**	112
freq_ut	83.30	89.49	84.55	91.96	86.77	89.14	**94.53**	**98.22**	**96.30**	109
d_dos_form_ext	72.77	61.57	64.49	52.64	51.13	51.85	**92.60**	**81.60**	**85.69**	66
A	76.18	63.96	68.21	**95.32**	76.08	**82.52**	85.18	**80.99**	79.81	52
roa	64.44	72.89	67.62	75.00	57.36	59.79	**82.78**	**91.57**	**85.28**	46
freq_int_v1	44.29	45.00	43.38	60.00	54.44	57.01	**87.78**	**88.33**	**87.42**	31
qsp_val	57.50	53.78	55.27	60.00	60.00	60.00	**100.00**	**100.00**	**100.00**	29
rhythm_rec_ut	41.00	56.00	46.48	54.44	52.78	53.10	**90.00**	**89.44**	**89.00**	29
max_unit_val	0.00	0.00	0.00	40.00	33.33	36.00	**80.00**	**62.67**	**69.29**	28
qsp_ut	35.56	35.56	35.00	60.00	60.00	60.00	**96.00**	**100.00**	**97.78**	28
freq_int_v1_ut	51.67	60.00	55.32	80.00	51.11	58.82	**83.43**	**84.44**	**80.41**	26
rhythm_rec_val	45.09	56.00	48.06	75.00	58.29	60.48	**87.67**	**96.00**	**90.88**	24
freq_int_v2	60.00	53.33	56.00	80.00	63.33	68.18	**100.00**	**90.00**	**94.18**	20
freq_val	20.00	13.33	16.00	53.33	46.67	49.33	**93.33**	**76.67**	**82.67**	19
fasting	60.00	60.00	60.00	56.67	56.67	56.36	**100.00**	**80.67**	**84.85**	18
max_unit_uf	0.00	0.00	0.00	36.00	32.00	32.78	**66.00**	**56.67**	**59.11**	18
freq_int_v2_ut	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	10
weighted avg	92.08*	93.94	92.88	92.26*	93.26	92.62	**94.82***	**95.72***	**95.17***	11,465
macro avg	66.06*	67.66	66.25	74.57*	70.64	71.55	**87.72***	**86.62***	**86.27***	11,465
micro avg	93.33	93.94	93.63	93.36	93.26	93.31	**94.82**	**95.72**	**95.26**	11,465

Table 4.11: Detailed Model Evaluation for PxCorpus/Task 1 - The table presents model metrics averaged across five folds for 30 entities. Bold and underline highlight the best and second-best outcomes. Medals show metric ranks per fold, with gold for the top model and red for the Null metric. Pastel medals depict absolute ranking, and vibrant medals indicate statistical significance at $\alpha = 0.05$. Micro average and class-level evaluations used the McNemar test with Bonferroni correction. Macro and weighted averages' significance were assessed using the Friedman test followed by the Nemenyi test. For easy navigation, Table F.8 shows the task statistics and Table 4.2 the main results.

4.3.2.6 QUAERO/Medline

Just like its *EMEA* equivalent, QUAERO/Medline includes the same 10 entities, but this time it centers on the titles of research articles indexed in the Medline database.

Named Entities	CamemBERT			DrBERT			TransBERT			Support
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1	
Chemical/Drugs	91.37	92.53	91.93	90.92	92.43	91.67	**91.58**	**92.61**	**92.08**	2,167
Disorders	**82.59**	**83.03**	**82.80**	80.25	81.59	80.90	81.02	82.88	81.93	1,286
Procedures	82.38	81.57	81.95	81.99	81.58	81.76	**84.64**	**82.65**	**83.61**	835
Living Beings	90.58	91.36	90.95	90.21	91.53	90.86	**91.92**	**93.51**	**92.70**	722
Physiology	60.17	67.33	63.08	**71.66**	**68.11**	**69.52**	67.19	67.79	67.41	300
Anatomy	75.11	**72.95**	73.72	74.80	70.16	72.22	**76.36**	72.18	**73.79**	265
Objects	65.92	**74.06**	69.52	**70.08**	69.74	69.54	69.94	70.19	**69.83**	162
Devices	**88.30**	80.38	84.01	85.25	81.67	83.15	86.99	**83.04**	**84.91**	144
Geo. Areas	77.66	83.67	80.48	87.73	84.45	85.94	**88.52**	**87.67**	**87.95**	64
Phenomena	38.57	18.17	24.52	67.29	47.22	54.99	**70.68**	**54.63**	**61.34**	56
Weighted avg	84.56	85.09	84.71	84.77	84.89	84.75	**85.61**	**85.85**	**85.67***	6,001
Macro avg	75.27	74.51	74.30	80.02	76.85	78.05	**80.88**	**78.72**	**79.55***	6,001
Micro avg	84.65	85.09	84.87	84.84	84.89	84.86	**85.59**	**85.85**	**85.72**	6,001

Table 4.12: Detailed Model Evaluation for QUAERO/EMEA - The table shows model metrics averaged over all five folds for the 10 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table F.10 shows the task statistics and Table 4.2 the main results.

It's essential to highlight that this source of data is primarily what TransBERT was pre-trained on. Table 4.2 shows TransBERT outperforming CamemBERT, the second-best model, by 1.91 points, which in turn is 1.41 points ahead of DrBERT.

A glance at the averaged results across folds at the entity level in Table 4.13 reveals that TransBERT holds the top value for the majority of the entities for each metric, particularly excelling at the top-3 supported entities, with statistical significance in two cases. Conversely, CamemBERT excels in identifying "*Living Beings*", "*Geo. Areas*" and "*Objects*" for Precision, Recall, and F_1. This result aligns with expectations from a non-DS LM; namely, the DS model excels in DS-related named entities, while the general model outperforms in broader entities. Having the lowest F_1 four times in the most supported entity, twice with statistical significance, DrBERT shows the lowest metrics overall, with a few instances of being the runner-up.

When evaluating the micro averages, TransBERT achieves the highest value five, four, and five times for Precision, Recall, and F_1 respectively, with statistical significance on two folds. It also excels in the entity aggregation, especially in the weighted Precision where it obtains the highest score each time, once with

Named Entities	CamemBERT			DrBERT			TransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Disorders	66.21	62.87	64.49	63.25	62.60	62.91	**67.31**	**64.85**	**66.04**	2,115
Procedures	61.68	64.53	63.06	61.56	64.82	63.11	**65.12**	**67.57**	**66.28**	1,528
Chemical/Drugs	68.48	71.17	69.70	66.98	70.21	68.47	**72.48**	**72.17**	**72.27**	819
Living Beings	**75.03**	**74.31**	**74.64**	71.37	70.61	70.96	74.42	73.87	74.11	777
Anatomy	55.62	50.48	52.84	53.89	52.40	53.09	**58.84**	**53.55**	**55.97**	744
Physiology	37.64	**39.74**	38.53	40.33	35.28	37.46	**41.11**	39.45	**40.17**	353
Geo. Areas	**81.99**	**82.90**	**82.33**	70.99	67.77	69.18	77.63	78.88	77.97	126
Phenomena	**33.20**	22.46	25.35	32.63	21.02	25.23	33.08	**23.01**	**26.56**	123
Devices	36.38	34.84	35.26	35.90	29.83	32.32	**45.00**	**38.95**	**41.07**	97
Objects	**47.20**	**34.22**	**37.92**	29.60	26.06	26.08	36.80	32.46	33.10	83
Weighted avg	62.89	61.87	62.22	60.87	60.78*	60.70*	**64.87**	**63.50**	**64.05**	6,765
Macro avg	56.34	53.75	54.41	52.65	50.06*	50.88*	**57.18**	**54.47**	**55.35**	6,765
Micro avg	62.90	61.87	62.38	61.16	60.78	60.97	**65.10**	**63.50**	**64.29**	6,765

Table 4.13: Detailed Model Evaluation for QUAERO/Medline - The table shows model metrics averaged over all five folds for the 10 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using a McNemar test with Bonferroni correction ($\alpha_{\text{Bonferroni}} = \frac{\alpha}{n_{test}} = \frac{0.05}{3}$). Macro and weighted averages significance were evaluated using a Friedman test followed by a Nemenyi test on labels metrics. For easy navigation, Table F.11 shows the task statistics and Table 4.2 the main results.

statistical significance. At the bottom of the table, all averaged aggregated metrics ranks appear to be segregated by model. While CamemBERT shows no significant difference from the others, DrBERT ranks last in the majority of aggregated metrics, with one fold featuring the worst $F_{1_{macro}}$ with statistical significance. Being consistent across entities, DrBERT also shows the worst macro and weighted average Recall and F_1 with statistical significance.

4.3.3 Part-of-Speech Tagging Task

POS tagging is a key task in NLP that entails identifying the grammatical categories (e.g., nouns, verbs, adjectives) for each word within a sentence. This task is crucial for revealing the linguistic structure of a text, assisting in the comprehension of syntactic relationships among words. The findings shown in Table 4.2 indicate that the models have already reached near-perfect $F_{1_{micro}}$ scores, with the lowest averaged score being above 97% and with relatively small CI. This outstanding performance showcases the models' proficiency in identifying these grammatical

categories, regardless of whether the lexical content is medical. With minimal room for improvement, determining the most outstanding model overall becomes challenging.

4.3.3.1 CAS

CAS is a POS tagging dataset centered on clinical cases, comprising a total of 86,805 words spread across 30 tags. As depicted in Table 4.2, all three models attain fairly high $F_{1_{micro}}$ scores with relatively tight CI. TransBERT secures the highest score, surpassing CamemBERT by just 0.07 points. Due to its notably low variance, DrBERT lags by 0.15 points, making it the poorest performing model.

Table 4.14 presents a detailed performance analysis of each POS tag across all models. Due to high support, each tag has a significant number of instances in training, leading to average metrics reaching up to 99.66% in "*Pers. Pr.*" Recall, which ranks the lowest. With regard to the F_1 score averaged across labels, CamemBERT, and DrBERT and TransBERT achieve the highest values for 13, 6, and 11 tags, respectively. While the best metrics of CamemBERT and DrBERT tend to be found in tags with lower support, TransBERT achieves its best results in tags with the highest support. A few zeroes appear in the "*Subjunctive Imperfect Verb*", the least supported tag, with DrBERT achieving favorable results in three folds.

Although the average metric across folds is marginally higher for TransBERT, it excels in four instances compared to one for CamemBERT. For the weighted and macro aggregation, perspective is crucial; a statistical test based on the F_1 rank across all tags and folds highlighted DrBERT's significance. However, DrBERT tends to achieve the worst results on the most supported metric and the best on the least supported ones. This gives it the lowest overall $F_{1_{weighted}}$ but the highest $F_{1_{micro}}$. While the statistical results mirror DrBERT's dataset performance, the narrow competitiveness makes the situation quite paradoxical. Looking at the micro aggregation, CamemBERT and TransBERT both obtain the top results with statistical significance in two distinct folds each, while DrBERT achieves the lowest outcomes with the same significance in two. This aligns with our previous observations that stated that DrBERT had poor results for well-supported tags, whereas TransBERT frequently had the best results.

4.3.3.2 ESSAI

ESSAI is a POS tagging dataset focused on clinical trial protocols. It represents the largest dataset, containing 150,269 words divided into 29 tags. As shown in Table 4.2, on average, CamemBERT leads TransBERT by 0.02 points, which in turn leads DrBERT by 0.10 points. All models achieve over 98% in this metric, with low variability across folds, evidenced by CIs of up to 0.04 points. The tag distribution is highly skewed, with the top-3 tags accounting for more than 50% of the instances.

Referencing Table 4.15, the competition is intense as evidenced by the lowest mean F_1 score for "*Subjunctive Imperfect Verb*" at 99.72%. The figures start to become intriguing midway through the table, where despite an average exceeding 98%, many tags see CamemBERT securing the highest F_1 scores. Notably, some tags exhibit perfect ties in terms of metrics and folds, including "*Sentence*", "*Poss.*

POS Tags	CamemBERT			DrBERT			TransBERT			Support
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1	
Noun	**97.34**	96.63	96.98	96.94	96.72	96.83	97.30	**96.88**	**97.09**	20,052
Pers. Pr.	99.38	99.71	99.54	99.43	**99.71**	99.57	**99.61**	99.66	**99.64**	11,049
Adjective	**95.47**	95.20	**95.33**	95.10	94.61	94.85	95.35	**95.23**	95.29	9,179
Article	99.64	99.89	99.77	99.55	**99.91**	99.73	**99.72**	99.90	**99.81**	9,085
Punctuation	**99.95**	**99.86**	**99.90**	99.91	**99.86**	99.88	99.94	99.75	99.84	7,500
Number	98.01	**99.05**	98.53	98.06	99.00	98.53	**98.43**	99.02	**98.72**	4,298
Sentence	99.95	**99.92**	99.94	**100.00**	99.90	**99.95**	99.97	99.87	99.92	3,883
PastP Verb	**95.54**	**97.20**	**96.35**	95.13	96.58	95.85	95.14	96.56	95.84	3,114
Conjunction	**98.12**	98.33	**98.22**	98.09	98.20	98.14	98.04	**98.40**	98.21	2,655
Present Verb	**96.96**	96.97	96.96	96.22	96.93	96.57	96.60	**97.45**	**97.02**	2,485
Adverb	**97.90**	96.99	97.44	97.85	96.82	97.33	97.64	**97.66**	**97.65**	2,468
Poss. Pr.	99.77	99.87	99.82	99.63	**99.91**	99.77	**99.78**	**99.91**	**99.84**	2,233
Imperfect Verb	**99.71**	**99.80**	**99.76**	99.39	99.61	99.50	99.53	99.71	99.62	2,117
Pers. Pr.	98.94	**98.74**	**98.84**	98.94	98.41	98.67	**99.32**	98.30	98.80	1,583
Proper Noun	81.35	**86.88**	83.95	**83.47**	83.30	83.37	82.42	85.72	**84.02**	1,446
Inf. Verb	**98.33**	97.74	**98.02**	97.81	97.35	97.56	97.93	**98.06**	97.97	567
PresP Verb	**95.94**	**96.17**	**96.04**	94.26	95.40	94.78	95.33	94.79	95.00	512
Abbreviation	74.63	68.84	71.20	81.16	**73.22**	76.79	**82.30**	73.02	**77.09**	471
Poss. Det.	99.04	**99.57**	99.30	98.82	**99.57**	99.18	**99.36**	**99.57**	**99.46**	428
Demon. Pr.	98.98	**100.00**	99.49	99.25	**100.00**	**99.62**	**99.48**	99.75	99.61	397
Relative Pr.	97.82	94.87	96.29	97.80	**96.67**	**97.22**	**98.08**	95.64	96.81	320
Indef. Pr.	97.78	**100.00**	**98.87**	97.93	98.52	98.21	**98.35**	98.46	98.40	263
Quot. Punct.	99.69	**99.58**	**99.63**	98.92	97.07	97.93	**100.00**	98.13	99.05	232
Symbol	**99.55**	**99.30**	**99.41**	95.39	96.16	95.56	95.09	99.26	97.02	210
Past Verb	75.71	64.14	67.04	**83.04**	69.48	**75.43**	80.04	**70.11**	74.53	130
Future Verb	**87.52**	50.22	**62.47**	79.33	49.98	60.38	79.68	**51.65**	61.11	46
Cond. Verb	**90.29**	**83.33**	**85.93**	89.33	70.00	76.77	81.07	**83.33**	81.44	26
SubjP Verb	52.00	35.00	38.91	**89.29**	**67.62**	**73.29**	81.67	60.95	61.72	22
Interjection	60.00	32.00	41.67	90.00	65.33	75.29	**100.00**	**70.33**	**81.48**	18
SubjI Verb	0.00	0.00	0.00	**40.00**	**24.67**	**30.00**	13.33	8.00	10.00	16
Weighted avg	97.70	97.66	97.66	97.60	97.55	97.56*	**97.78**	**97.73**	**97.74**	86,805
Macro avg	89.51	86.19	87.19	**93.00**	88.68	**90.22***	92.02	**88.84**	89.73	86,805
Micro avg	97.73	97.66	97.69	97.61	97.55	97.58	**97.79**	**97.73**	**97.76**	86,805

Table 4.14: Detailed Model Evaluation for CAS - The table shows metrics averaged over five folds for 30 POS tags. Bold and underline highlight the best and second-best outcomes. Medal colors rank each fold: gold for the top model, red for Null metrics, pastel for absolute ranking, and vibrant for statistical significance ($\alpha = 0.05$). Micro average and individual class levels were evaluated using a McNemar test with Bonferroni correction. Macro and weighted averages used a Friedman test followed by a Nemenyi test. For easy navigation, Table F.1 shows the task statistics and Table 4.2 the main results.

Pr.", "*Demon. Pr.*", "*Quot. Punct.*" and *Symbol*". For these tags, the F_1 scores averaged across folds reach up to 100.00%. However, the tag "*Symbol*", being among the least supported, has four tied folds where no model accurately predicted

any instance.

POS Tags	CamemBERT			DrBERT			TransBERT			Support
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1	
Noun	98.55	98.34	98.45	98.43	98.32	98.37	**98.56**	**98.42**	**98.49**	39,279
Pers. Pr.	99.66	99.83	99.74	99.55	**99.89**	99.72	**99.66**	99.86	**99.76**	22,261
Article	99.79	99.89	99.84	99.79	**99.90**	**99.85**	**99.81**	99.88	99.84	18,404
Adjective	96.65	**95.77**	96.21	96.53	95.24	95.88	**96.69**	95.75	**96.21**	11,056
Punctuation	99.99	**100.00**	**99.99**	99.98	**100.00**	99.99	**100.00**	99.92	99.96	9,272
Sentence	**99.98**	**99.98**	**99.98**	**99.98**	**99.98**	**99.98**	**99.98**	**99.98**	**99.98**	6,016
Conjunction	**98.83**	98.78	**98.81**	98.54	**98.87**	98.70	98.78	98.82	98.80	5,653
Number	99.01	99.30	99.15	99.06	99.26	99.16	**99.06**	**99.33**	**99.20**	5,530
Poss. Pr.	**99.93**	**99.91**	**99.92**	**99.93**	**99.91**	**99.92**	99.93	99.91	99.92	5,480
PastP Verb	**97.12**	**97.86**	**97.49**	96.59	97.57	97.07	96.86	97.67	97.26	4,821
Present Verb	**98.83**	**97.81**	**98.32**	98.47	97.53	97.99	98.72	97.58	98.15	3,556
Adverb	98.16	**98.55**	**98.36**	**98.35**	97.81	98.08	98.06	98.19	98.12	3,490
Proper Noun	**88.46**	91.58	**89.97**	86.30	91.37	88.73	88.02	**91.99**	89.93	2,622
Future Verb	**99.54**	**99.57**	**99.55**	99.54	99.49	99.52	99.53	99.53	99.53	2,562
Inf. Verb	**99.22**	99.51	**99.36**	99.18	**99.51**	99.35	99.10	99.51	99.30	2,442
Demon. Pr.	**99.83**	**100.00**	**99.92**	**99.83**	**100.00**	**99.92**	**99.83**	**100.00**	**99.92**	1,796
PresP Verb	98.26	**98.68**	**98.47**	98.14	98.56	98.35	**98.32**	98.62	98.47	1,661
Indef. Pr.	**99.32**	**99.66**	**99.49**	99.00	99.34	99.17	99.07	99.34	99.20	1,210
Pers. Pr.	98.35	**97.31**	**97.82**	**98.76**	96.30	97.52	98.32	96.84	97.57	1,089
Relative Pr.	99.27	**99.24**	**99.25**	**99.56**	98.22	98.88	99.27	98.18	98.71	672
Abbreviation	62.91	**65.93**	**64.17**	64.27	56.56	59.75	**64.72**	61.39	62.72	325
Poss. Det.	99.69	**99.67**	99.68	99.72	**99.67**	**99.69**	**100.00**	99.35	99.67	312
Quot. Punct.	**100.00**	**100.00**	**100.00**	99.27	97.89	98.52	**100.00**	**100.00**	**100.00**	212
Noun Sing./Mass	94.71	97.61	96.07	95.66	**98.86**	**97.21**	**95.66**	98.75	97.14	161
Symbol	**100.00**	**98.75**	**99.35**	93.92	97.70	95.70	**100.00**	**98.75**	**99.35**	156
Cond. Verb	**100.00**	**92.61**	**96.05**	98.82	90.52	94.33	96.72	90.52	93.39	90
SubjP Verb	85.32	**59.17**	**68.98**	**87.50**	50.79	63.47	76.67	52.05	60.33	53
Past Verb	0.00	0.00	0.00	**20.00**	**2.00**	**3.64**	0.00	0.00	0.00	46
Imperfect Verb	**97.78**	**89.84**	**93.27**	93.50	81.62	86.56	96.36	83.21	87.84	42
Weighted avg	**98.67**	**98.66***	**98.66***	98.55	98.54	98.53*	98.65	98.63	98.64*	150,269
Macro avg	93.42	**92.25***	**92.68***	**93.73**	91.13	91.90*	93.02	91.49	92.03*	150,269
Micro avg	**98.68**	**98.66**	**98.67**	98.56	98.54	98.55	98.66	98.63	98.65	150,269

Table 4.15: Detailed Model Evaluation for ESSAI - The table presents average model metrics across five folds for 29 POS tags. Bold and underline highlight the best and second-best results. Medal colors indicate ranking per fold: gold for top model, red for Null metrics, pastel for absolute ranking, and vibrant colors for significant metrics ($\alpha = 0.05$). Micro and class-level evaluations used McNemar test with Bonferroni correction. Macro and weighted averages were assessed with Friedman and Nemenyi tests. For easy navigation, Table F.5 shows the task statistics and Table 4.2 the main results.

In conclusion, CamemBERT achieves the highest rank for the micro aggregation averaged across fold for all metrics, with statistical significance noted in one instance. In the same aggregation, TransBERT attains the top rank in two folds,

while DrBERT consistently ranks lowest, once with statistical significance. For both weighted and macro aggregations, CamemBERT demonstrates the best Recall and F_1 scores, securing the top rank in most folds and twice achieving statistical significance, once in first place and once in second. DrBERT displays a more complex performance, showing the lowest weighted average across all metrics and folds, although it yields the best weighted average Recall and second-best F_1 with significance in the same fold where it had the worst average. TransBERT's performance is intermediate, showing improved results in aggregations that consider support distribution. Despite slight differences in macro and weighted F_1 averaged across folds for all models, the high sparsity rank across tags and folds allows CamemBERT, TransBERT, and DrBERT to secure first, second, and third places, respectively, with statistical significance. CamemBERT also achieves the best Recall averaged across folds with statistical significance.

4.3.4 Semantic Textual Similarity Task

STS involves performing a regression task to determine the similarity or dissimilarity between two given sentences. For this task, DrBenchmark provides two datasets. As observed in Table 4.2, there is a slight difference between CamemBERT and TransBERT, while DrBERT shows significantly lower results when considering the average R^2 over the folds. The subsequent sections will delve into the analysis of both datasets.

4.3.4.1 CLISTER

CLISTER is a STS dataset that deals with sentence pairs derived from clinical cases. Table 4.2 illustrates that CamemBERT and TransBERT occupy the first and second positions respectively, with a difference of 0.18 points on the R^2 averaged across folds.

Figure 4.4 shows how the model predictions correlate with the actual scores. Even though the similarity scores are shown as floating-point numbers, three vertical lines are placed between each integer to show the level of detail possible when averaging annotations from two annotators. Therefore, to make the plots clearer, each model is plotted individually with the predictions of other models in the background. Larger deviations from the dashed line imply worse predictions, whereas points on this line represent perfect classification.

Examining the actual labels reveals an evident pattern along the lines with reduced variability at both ends, likely due to the predicted scores being restricted between 0 and 5. When inspecting the graphs of CamemBERT and TransBERT, traces of DrBERT's model predictions are noticeable in the background, aligning with its performance. It is worth noting that DrBERT seems to struggle more at the extremes, displaying smaller dots at the corner points (0, 0) and (5, 5), and showing slightly more outliers in various areas. There are no significant differences between CamemBERT and TransBERT; when analyzing their graphs, only a few dots appear to spill over into the background of the other.

Beyond this superficial analysis, the graph does not provide much information about why there is such a wide variance around the dashed line. However, both the R^2 and the clustering around that line indicate that the models have learned something, which is crucial. An examination of the worst prediction errors for

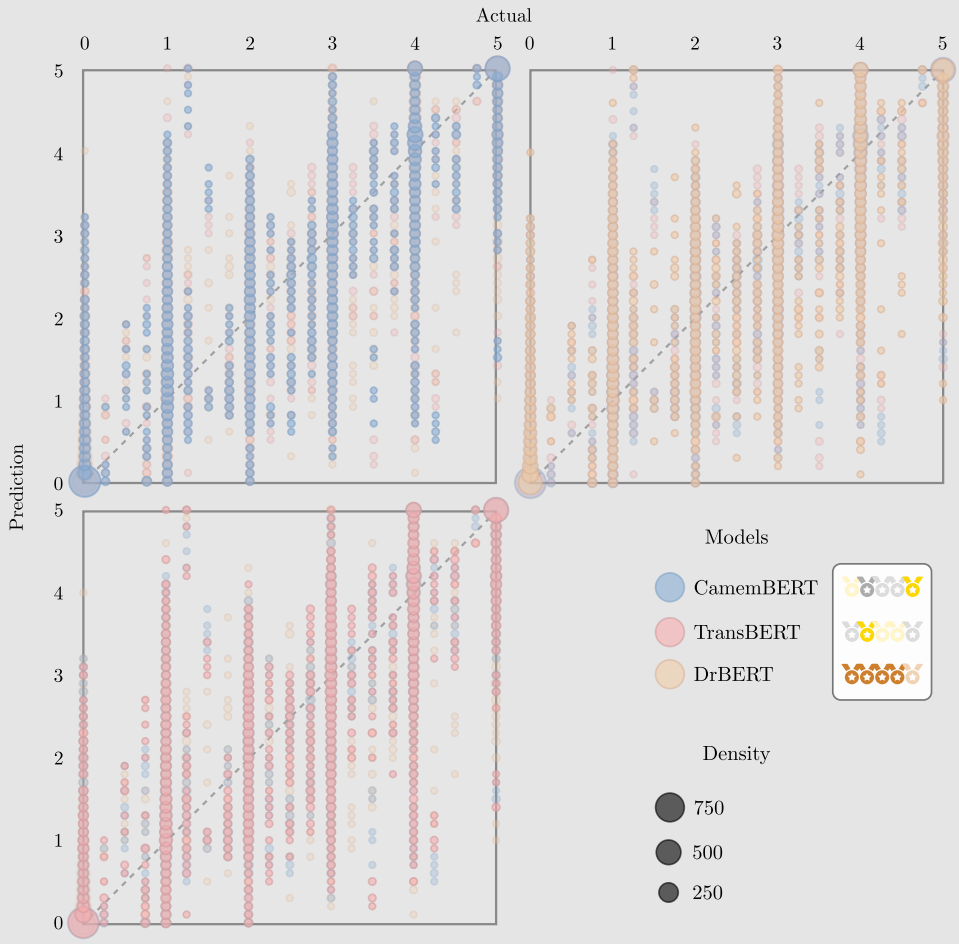

Figure 4.4: CLISTER Semantic Textual Similarity Scatter Plot - The figure illustrates scatter plots where each model's prediction is represented by a dot in relation to the actual values. All folds and rounds have been included, with larger dots indicating a higher concentration of observations.

two similar and dissimilar sentences is presented in Figure 4.5. In both instances, DrBERT is the model that deviates substantially. For the 0.0 similarity score case, while the outcome is clear to a human, even the two best models found it difficult to differentiate the sentences. On the other hand, the 5.0 similarity case appears to be predicted quite accurately by both CamemBERT and TransBERT, as the sentences use almost identical wording. Here, it is unexpected to observe DrBERT making such an inaccurate prediction.

Figure 4.4 also shows results per fold along with statistical testing. Although it was previously evident that DrBERT received the lowest scores, it is now indisputable as it ranked last in each fold with significance in four out of five cases. CamemBERT achieves the highest R^2 averaged across the folds, securing the first place twice, once with significance. TransBERT obtained the best results in three out of five folds; however, once with significance too.

CamemBERT: 2.16 / **DrBERT: 3.95** / **TransBERT:** 2.13

Similarity Score: 0.0

Sentence 1: Le testicule gauche est normal.

Sentence 2: Le toucher rectal est normal.

CamemBERT: 4.71 / **DrBERT: 1.0** / **TransBERT**: 4.37

Similarity Score: 5.0

Sentence 1: 'On réalisait une urétérotomie sur la sténose permettant de placer un endoprothèse double J Ch.7 siliconée.

Sentence 2: te. Une urétérotomie sur la sténose fut réalisée permettant de met tre en place une endoprothèse double J Ch. 7 silic

Figure 4.5: CLISTER - Highest Error Prediction Sample - Highest error for both extremities of the graph.

4.3.4.2 DEFT-2020/Task 1

The last STS dataset is the first task of the DEFT-2020 competition, with similarity scores now determined by averaging ratings from five annotators, rather than two. Similar to the classification task in this competition, the dataset comprises encyclopedic information. As seen with the CLISTER results, Table 4.2 shows that CamemBERT is leading by 0.49 points of the R^2 averaged across folds.

Figure 4.6 depicts the similarity scores in relation to the model predictions. By averaging across five annotators, we observe a dispersed vertical line effect. Despite having the same width as Figure 4.4, the absence of vertical alignment gives the impression that the models fit better. Nevertheless, the conclusions are similar to the CLISTER case. Notably, DrBERT is remarkably wider than the other two models, and its predictions at both extremes deviate compared to the others. When contrasting CamemBERT and TransBERT, there are a few instances where one model's data appears in the background of the other's, but the frequency seems balanced.

Across folds, DrBERT consistently delivers the lowest results with statistical significance. However, CamemBERT and TransBERT alternate between the first and second positions with the highest R^2 values, three for CamemBERT and two for TransBERT, neither of which demonstrates statistical significance.

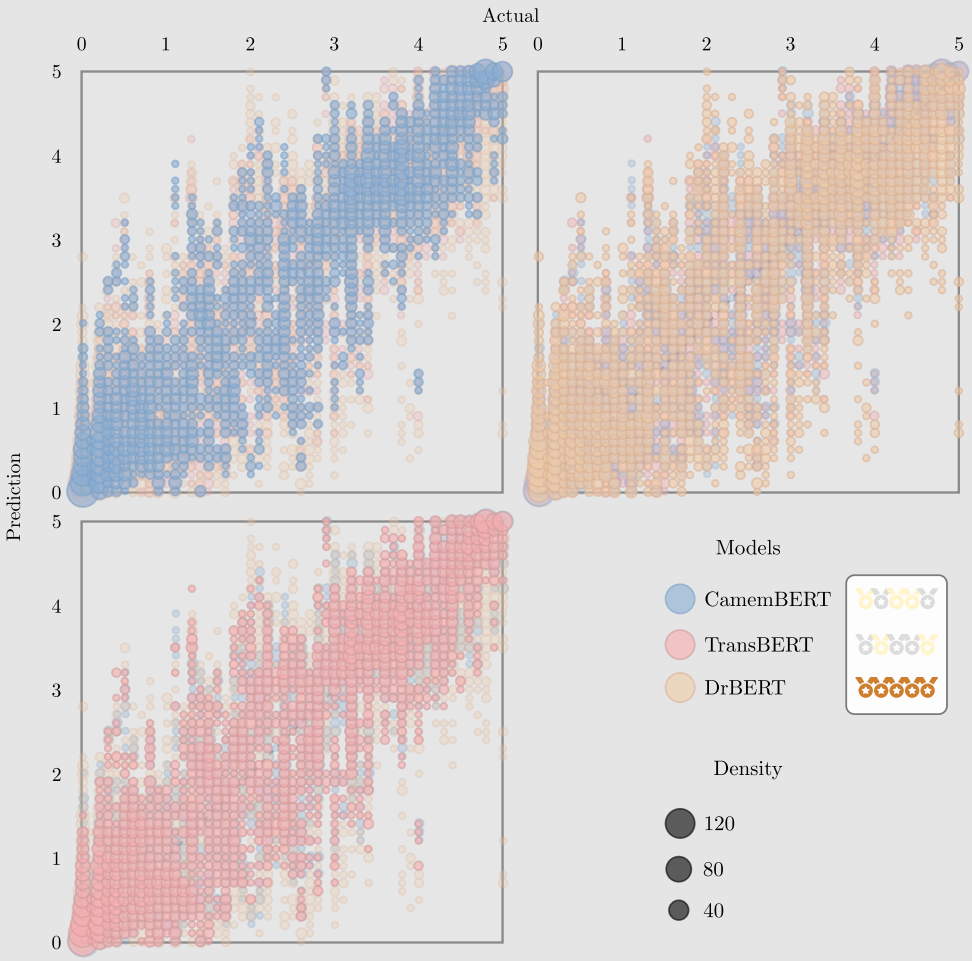

Figure 4.6: DEFT-2020 Semantic Textual Similarity Scatter Plot - The figure depicts a scatter plot showing each model's prediction as a dot against the actual value. Since the actual labels are not continuous, a systematic shift has been introduced to distinguish each model's predictions, CamemBERT to the left, TransBERT to the right. All folds/rounds have been taken into account, the bigger the dot, the more concentration of observation there are.

4.4 Performance Analysis Aggregation

In Section 4.2.3, we demonstrated that any form of abstraction could be utilized to conduct a statistical evaluation of our performance. We previously illustrated the usefulness of this approach across various datasets by testing models on labels metrics ranking, both by fold and across folds, which consists of a weighted mean of both macro and weighted averages. These tests provided us with valuable insights, enabling us to complete our dataset analysis with more robust metrics and knowledge. While identifying the best model for a specific dataset was significant, applying this method at the task level would yield even more informative results.

To conduct this testing, we must adhere to the schema used so far. This involves considering any class/label, entity, and tag metric, respectively, for any

classification, NER, POS, dataset. The next section will provide insightful metrics for each model across all datasets task by task. It is important to note that the statistical testing will rank each metric, potentially leading to different conclusions when examining overall aggregated metrics.

Our task analysis will be founded on (1) the weighted average of each task metric by the support of each label/dataset/fold, (2) a macro average of the same values, which inherently does not account for support, (3) an assessment of the rankings per metric across label/dataset/fold (4) a Normalized Ranking Average (NRA) serving as a ranking score, and (5) results of statistical significance derived from the rankings of the evaluated metrics.

The Nemenyi test has provided the rankings for each metric. To calculate the NRA, it is essential to consider that the rankings' average spans from 1 to the total number of models minus one, with the lowest metric assigned the top rank. Consequently, after normalization, a higher score is more favorable. Once the average ranking for a particular model is computed, the NRA can be calculated as follows:

$$\text{NRA} = \frac{\text{RA} - 1}{n_{model} - 1} \tag{4.1}$$

n_{model} represents the number of models under comparison and RA denotes the Ranking Average, ranging from 1 to n_{model} with 1 being the worst rank.

4.4.1 Classification Task Analysis

For the classification task, 185 class/labels across five datasets and folds are analyzed. For DEFT-2020/Task 2, it is worth nothing that the micro average will be used since classes serve as placeholders in this dataset. Prior to delving into Table 4.16, let us briefly revisit the conclusions from our task-by-task analysis. For DEFT-2020/Task 2, no definitive conclusion was reached as all models demonstrated competitive performance. In DiaMed, the analysis indicated TransBERT as the most effective model, showing statistical significance. In FrenchMedMCQA, DrBERT underperformed with statistical significance noted for accuracy on two folds. For MorFITT, TransBERT achieved the highest weighted and macro averages for Recall and F1 scores across folds, also with statistical significance, achieving the highest average Recall and F1 scores for 11 and 12 labels, respectively. Lastly, in PxCorpus/Task 2, CamemBERT showed statistical significance in one fold for accuracy, while achieving the highest accuracy in three folds.

Table 4.16 presents the main aggregated statistics for the classification task. Firstly, an overview of the ranking indicates that TransBERT secures the highest Precision, Recall, and F_1 scores by a substantial margin, achieving almost twice as many wins as the other two models. Examining both the weighted and macro averages, TransBERT attains the highest scores across all metrics. The Friedman and Nemenyi tests demonstrated that its average ranking is significantly different, even with an $\alpha = 0.01$ threshold. The NRA highlights the disparity in rankings, which is unsurprising given the ranking distribution. On the other hand, both CamemBERT and DrBERT exhibit similar performance, with CamemBERT showing considerably better weighted Recall and DrBERT being competitive in macro Precision.

	CamemBERT			DrBERT			TransBERT		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	74.65	75.54	74.17	74.81	73.42	73.73	**75.82****	**76.69****	**75.71****
Macro Avg	57.74	56.94	55.66	60.76	56.71	57.60	**64.06****	**62.55****	**61.93****
🥇 / 🥈	**2**/57	**1**/82	**1**/61	1/68	**1**/65	**1**/52	1/**107**	**1**/**124**	**1**/**113**
🥈 / 🥉	**0**/51	**0**/53	**0**/58	**0**/**65**	**0**/**62**	**0**/**65**	**0**/55	**0**/40	**0**/50
🥇 / 🥉	**1**/**74**	**2**/47	**2**/63	**1**/50	1/**56**	1/**66**	0/22	0/20	0/21
NRA	37.70	45.81	42.57	46.76	39.32	39.05	**65.54****	**64.86****	**68.38****

Table 4.16: Model Evaluation for the Classification Task - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different classes/labels for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Friedman test followed by Nemenyi post-hoc tests. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

4.4.2 Named Entity Recognition Task Analysis

Within the NER task, our benchmark covers 315 entities distributed over six datasets and five folds. Summarizing our task by task analysis findings, in E3C/Clinical, TransBERT secured 80% of the highest metrics, yet with only one entity, no statistical significance was achieved. For E3C/Temporal, DrBERT reached statistical significance by getting the lowest performance in a consistent manner. For MantraGSC, DrBERT consistently had the lowest results, making it the worst in aggregated Recall across folds, with statistical significance. In PxCorpus/Task 1, TransBERT recorded the highest scores averaged across folds for both weighted and macro averages across and gets statistical significance for that. In QUAERO/EMEA, TransBERT consistently achieved high F_1 scores across entities, resulting in statistical significance. In QUAERO/Medline, DrBERT demonstrated the lowest macro and weighted average Recall and F_1 averaged across folds, resulting in statistical significance for its overall ranking.

Referring to Table 4.17, the ranking distributions align with our previous analysis summary. DrBERT generally achieves the lowest results, whereas TransBERT consistently attains the highest metrics. While TransBERT shows the highest averages overall, it outperforms the other two models in the macro aggregation by quite a substantial amount that goes up to approximately 11 points. TransBERT demonstrates statistical significance with a p-value < 0.01 for Precision, Recall, and F_1. While there is only a minor difference in Precision between CamemBERT and DrBERT, they notably differ in Recall, with CamemBERT showing statistically significant results based on the metric ranking. The increase in Recall enhances its F_1 considerably, although not sufficiently to be deemed significant.

4.4.3 Part-of-Speech Tagging Task Analysis

With two datasets and five folds, POS tagging was tested on 295 tags, showing substantially high scores. Synthesizing the conclusions derived from the datasets, within the CAS dataset, DrBERT recorded the lowest F_1 scores across tags, thereby achieving statistical significance with the lowest weighted averages. On the other

	CamemBERT			DrBERT			TransBERT		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	81.23	82.13**	81.55	80.74	81.27**	80.88	**83.03****	**83.46****	**83.15****
Macro Avg	66.23	66.45**	65.60	70.22	66.90**	67.62	**77.72****	**76.75****	**76.45****
🥇 / 🥉	2/90	5/141	**4**/89	**4**/113	**7**/98	3/92	**4/177**	4/**210**	**4/179**
🥈 / 🥈	**2/107**	**0/94**	**0/113**	0/77	**0**/80	**0**/84	0/95	**0**/78	**0**/97
🥉 / 🥇	7/**107**	6/69	7/102	**22**/99	**19/111**	**23/113**	0/39	0/23	0/35
NRA	41.35	48.89**	43.17	42.62	34.44**	38.17	**66.03****	**66.67****	**68.65****

Table 4.17: Model Evaluation for the Named Entity Recognition Task - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different entities for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Friedman test followed by Nemenyi post-hoc tests. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

hand, in the ESSAI dataset, CamemBERT achieved the highest Recall and F_1 averaged across folds, reaching statistical significance. However, TransBERT and DrBERT also showed statistical significance, ranking second and third, respectively, at the macro level for the F_1 metric.

Table 4.18 shows the results for the POS task. Upon validating our analysis across datasets, TransBERT achieves the highest Precision, Recall, and F_1 scores although almost identical to CamemBERT, whereas DrBERT exhibits the lowest weighted metrics. The NRA reveals a substantial ranking difference between CamemBERT and TransBERT however, this distinction lacks statistical significance. On the other hand, DrBERT Precision and F_1 show statistically significant differences concerning their ranking. In terms of Precision, although the macro value is the highest among all, the weighted value is the lowest.This paradox underscores the complexity of statistical tests based on ranks. Although the tests identify DrBERT's ranks as the most inferior on average and corroborate this statistically, DrBERT achieves high results in critical tags, where other models score near zero. This high performance in isolated cases contributes to DrBERT's statistical significance. Therefore, the test, while directional, also aims to eliminate randomness. Essentially, our testing methodology concludes that a model achieving its metrics in a non-random manner, which indicates both exceptional performance in one scenario and poor performance in another. In other words, the metrics we deemed non-random are the input of the linear combination that constructs both weighted and macro averages.

4.4.4 Semantic Textual Similarity Task Analysis

The analysis of the STS task is straightforward. It involves two datasets, each divided into five folds with roughly equal data allocation. Table 4.19 shows that both CamemBERT and TransBERT achieved the highest R^2 in five occurrences, resulting in a tie in NRA while CamemBERT gets a slightly higher R^2. However, DrBERT consistently produced the poorest performance, with statistical significance.

	CamemBERT			DrBERT			TransBERT		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	98.31	98.29	98.29	98.20**	98.18	98.18**	**98.33**	**98.30**	**98.31**
Macro Avg	91.43	89.17	89.89	**93.36****	89.88	**91.04****	92.51	**90.14**	90.86
🥇 / 🥈	0/**171**	0/**209**	0/**181**	0/131	0/147	0/103	**2**/164	**2**/165	**2**/144
🥈 / 🥉	**1**/65	**0**/41	**0**/66	0/63	**0**/72	**0**/71	0/**74**	**0**/**83**	**0**/**89**
🥉 / 4	1/57	2/43	2/46	**5**/**96**	**5**/**71**	**5**/**116**	1/54	1/44	1/59
NRA	**54.49**	**58.47**	**60.76**	41.53**	41.78	36.27**	53.98	49.75	52.97

Table 4.18: Model Evaluation for the Part-of-Speech Tagging Task - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different tags for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Friedman test followed by Nemenyi post-hoc tests. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

4.4.5 Overall Aggregation

Upon reflection, aggregating everything into a single overall assessment seemed overly simplistic. Indeed, even before beginning, questions arise: How can class/label, entities, and tags be combined into one ranking? How does the POS Precision compare with classification Precision? Is the current task distribution representative of typical user experiences? These considerations led to the realization that creating a single aggregation was impractical. Therefore, readers are encouraged to examine results per task and create their own linear combinations to explore the outcomes.

	CamemBERT	DrBERT	TransBERT
	R^2	R^2	R^2
Weighted Avg	**83.38**	73.56**	83.04
Macro Avg	**83.38**	73.57**	83.04
🥇 / 🥈	**1**/4	0/0	**1**/4
🥈 / 🥉	**1**/4	0/0	0/**5**
🥉 /	0/0	**9**/1	0/0
NRA	**75.00**	0.00**	**75.00**

Table 4.19: Model Evaluation for the Semantic Textual Similarity Task - This table presents the weighted and macro aggregations for R^2 for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Friedman test followed by Nemenyi post-hoc tests. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

4.5 Conclusion & Discussion

This chapter covered a significant amount of information. To begin with, we introduced a framework for conducting experiments at the dataset level and developed a specialized reporting system that effectively presents the data relevant to our analysis. The results of each dataset were then thoroughly examined using our new reporting system, and the pre-established framework helped to identify various conclusions of the dataset through different statistical tests. Finally, according to (Demšar, 2006), a novel setup was subsequently created for testing the model across datasets, enabling us to draw some task related conclusions.

This chapter initially started with the statement of our first hypothesis, which is:

> **The current state of Machine Translation (MT) enables the development of a Language Model (LM) trained entirely on an automatically translated corpus, maintaining competitiveness with State-of-the-Art (SOTA) models in the field.**

In Chapter 2, we explored the current state of NLP by evaluating the latest SOTA models. We reviewed recent advancements in Machine Translation (MT), which informed our methodology in Chapter 3 in which we translated an extensive corpus from English to French and pre-trained a LM on the newly created large corpus. This chapter addressed the final step, which involved integrating our PLM with two French SOTA models in a modified version of DrBenchmark and proceeding to fine-tuning before analyzing their results.

From this chapter's experiment, we can derive several intriguing insights, though it is essential to begin with some of its limitations. Firstly, our hypothesis centers on comparing a model which derives from the previously defined methods across the largest possible number of tasks and datasets. However, as already discussed, sourcing non-English DS datasets within the life science domain can be difficult. Although adapting DrBenchmark offers a sound representation of typical tasks for life science models, there are countless ways and perspectives to compare

models. We chose this benchmark because it aligns with the common practices of the community, as discussed in the literature review with benchmarks such as General Language Understanding Evaluation (GLUE). In addition, there are various approaches to statistical testing; our methods were selected based on perceived effectiveness given the amount of datasets. Although there are numerous ways to aggregate results, we opted for the most straightforward and transparent choices to minimize arbitrariness.

Common LM benchmarks in life sciences are predominantly biomedical or clinical, such as the already mentioned Biomedical Language Understanding & Reasoning Benchmark (BLURB) and Biomedical Language Understanding Evaluation (BLUE). To compare our adaptation, BLUE comprises 10 datasets categorized into four tasks, including two classification tasks, three NER, two STS, and three Relation Extraction (RE). BLURB includes 13 datasets spread over six tasks, including one classification, five NER, one STS, a PICO, which is a framework for extracting evidence-based medical information, three RE, and two Question Answering (QA). Consequently, both benchmarks share classification, NER, STS, and RE tasks, while neither includes POS tagging. It is worth noting that QA can be somewhat related to NER as it is addressed with similar token representations. While it would indeed be intriguing to compare TransBERT in both RE and QA, our benchmark encompasses a substantial number of datasets for most tasks, including the addition of POS.

The DrBenchmark paper (Labrak et al., 2024) included an experiment that evaluated performance using training subsets of varying sizes: 25%, 50%, 75%, and 100%. This exploration provides important insights into a model's ability to utilize limited data and assess its scalability when increasing the training set size. Although exploring different number of total folds within a cross-validation setting could have yielded interesting insights, it was not conducted due to the high computational resources this side experiment would have required. However, the chosen setting mirrors real-world scenarios and standard machine learning methodologies, with models being typically trained on a fixed dataset and evaluated on a test set that usually consists of approximately 20% of the data. This is the same ratio used in several datasets within the DrBenchmark, including PxCorpus, MantraGSC, CAS, and ESSAI.

In the field, it is common to perform statistical tests over multiple training iterations using the same test set. Although performing a 5-folds cross-validation is a significant improvement, it can be argued that sharing parts of the training data among different folds introduces a degree of dependence, violating one of the assumptions of the statistical testing. Upon reviewing rankings categorized by class, label, entity, or tag within a data table, it is not unusual for models that usually perform poorly to sometimes secure the top scores. This actually underscores the metric variability across different folds, which reflects the disparity between folds of the dataset. Among the numerous metrics across folds, datasets, tasks, ups and downs, it is impossible to visually disociate which model is effectively better or worse. In that context, statistical testing highlights consistency. Although the independence assumption is somewhat violated, possibly increasing the likelihood of a Type I error, hypothesis testing functions as an autonomous component linked to the experiment. It remains uninvolved with the aggregation metrics per dataset or task, whose results still lead to the same conclusions.

As we initially aimed for our experiment to compete with SOTA models, our final section presented substantial improvement validating our hypothesis. We discovered that our model not only competes with the widely-used general French LM but also considerably outperforms it, along with the recently introduced French DS PLM. Our analysis of the classification task indicates that TransBERT achieved the highest aggregated Precision, Recall, and F_1 scores using two of the most common aggregation methods, ranking on average significantly better across all classification datasets. In the NER task, which is also highly regarded in the community, results on the tested dataset showed that TransBERT outperformed by a substantial margin of about 10 points when taking a macro average of all tested entities. It also showed a statistically significant higher average ranking compared to both DrBERT and CamemBERT. In the POS task, although there was no improvement compared to CamemBERT, it was statistically significantly better than DrBERT in ranking, achieving the best weighted Precision, Recall, and F_1 scores. Lastly, in STS, TransBERT obtained a better R^2 and ranked statistically better than DrBERT while being on par with CamemBERT.

Although each of the models were similarly fine-tuned, their primary distinction lay in their pre-training. A detailed examination of their respective papers reveals that the three models differ mainly in their corpora, which vary in several respects. CamemBERT utilizes a General-Domain corpus, whereas DrBERT and TransCorpus draw from life sciences corpora. In terms of corpus size, CamemBERT, DrBERT, and TransCorpus comprise 138GB, 7.5GB, and 36GB, respectively. Additionally, both CamemBERT and DrBERT are composed of native French data, while TransCorpus consists solely of data synthetically translated from English to French. By comparing CamemBERT to TransBERT, our experiments have indirectly demonstrated that having 138GB of native General-Domain data can be less advantageous than 36GB of DS data. Lastly, when comparing DrBERT with TransBERT, our results indicate that pre-training a model on a 7.5GB of native DS corpus does not exceed the performance of LM pre-trained on a 36GB of synthetically translated DS corpus.

It is crucial to note that our hypothesis was tested under specific conditions with respect to corpus size and translation quality. Although computational constraints prevented us from deeper exploration, future studies could benefit from examining the two core elements of the corpus: (1) corpus size and (2) translation quality. Evaluating the corpus size would involve pre-training several LMs and tokenizers with only a subset of the abstracts, which would reveal the data quantity necessary, for a given translation quality, to develop a life science LM and could serve as a cost-effective baseline for testing another DS domain. In addition, for a given corpus size, altering the quality of the translation would provide insight into how factors, such as the BLEU score of a translation model, influence the performance of the downstream task. This inquiry would clarify the repercussions of decreased translation quality, especially when tackling more challenging language pairs or low-resource languages. Although important, both research paths necessitate substantial computational resources, which can make them extremely costly, especially based on the experiment's granularity, e.g., each experiment would require about three months of training.

Chapter 5

The Impact of Domain-Specific Tokenization on Pre-trained Language Models Performance

This chapter expands upon the experiment from the previous chapter to evaluate TransBERT against cTransBERT. The primary objective of this evaluation is to verify whether the community's belief that training a Domain-Specific (DS) tokenization improves the performance of Pre-trained Language Models (PLMs) on specific downstream tasks holds. The significant adjustments in the experimental setup consist of altering the statistical tests, as the tests required for comparing two models differ from those used for comparing three models. To minimize redundancy with the last chapter's dataset-based results analysis, we will directly review aggregated results per task, enabling us to focus on tokenization when necessary. Detailed results tables for each dataset can be found in Appendix H for reference.

5.1 Introduction

This section outlines the research context, motivation, and hypothesis, preparing the reader for the experimental framework that follows. It emphasizes the importance of this study in addressing a gap in current research and potentially challenging a widely held assumption in the field of Natural Language Processing (NLP) for Domain-Specific (DS) tasks.

5.1.1 Motivation

Despite the scarcity of research that compares the effectiveness of tokenizers in DS tasks, the belief that DS tokenizer is more efficient is widely accepted within the community. This agreement probably stems from the observation that DS Language Models (LMs) built from scratch frequently outperform those fine-tuned from a Pre-trained Language Model (PLM) on a DS corpus. An example of this is the claim made by PubMedBERT in contrast to BioBERT (Gu et al., 2021) or even this sentence in the Limitation Section of DrBERT's paper, which mentions this hypothesis: "it would be wise to evaluate the impact of the tokenizer on

the performance of the models to ensure that this is not the main reason for the observed performance gains".

One approach to mitigating the impact of tokenizers on downstream applications is to conduct the same experiment twice from scratch. This entails replicating the pre-training process with different tokenizers. To our knowledge, no studies of this nature exist yet, since pre-training two PLMs on the same corpus is quite labor-intensive. Typically, researchers pre-train a model for comparison with others to evaluate the overall method's improvement. As mentioned earlier in Chapter 3, to achieve this goal, a LM has been pre-trained on the same machine-translated corpus using the CamemBERT tokenizer, which was trained on a non-DS corpus.

5.1.2 Hypothesis

Given the lack of rigorous evidence proving that the use of a DS tokenizer affects the measurement of downstream task performance, the hypothesis of this chapter is articulated as follows:

> **Domain-Specific (DS) tokenization enhances the performance of Pre-trained Language Models (PLMs) on specialized downstream tasks.**

The next section describes the experimental setup carefully crafted to thoroughly test our hypothesis. It is important to note that this setup mainly mirrors the methods and datasets in the previous chapter.

5.2 Experimental Setting

This section outlines the experimental framework designed to test the second hypothesis, comparing the performance of TransBERT and cTransBERT models.

5.2.1 Model Comparison

To address our second hypothesis, this experiment will compare TransBERT with cTransBERT. The latter is named after the combination of TransBERT and CamemBERT as it is methodologically identical to TransBERT but utilizes the CamemBERT tokenizer instead.

5.2.2 Mirrored Experiment

This chapter replicates the previous experiment by using the same datasets. The folds, model training procedures, and aggregation are executed exactly in the same way. Only two aspects will change (1) as there is a change related to the number of models to be tested, some testing procedures will require an adjustment, details will be discussed in the subsequent section. (2) In order to avoid redundancy and lengthy analysis and as we already have an in-depth understanding of TransBERT's performance across all datasets, the analysis of the results will be presented at the task level directly for conclusion. However, all datasets comparison tables can be found in Appendix H.

5.2.3 Statistical Testing

In Chapter 4, all statistical tests involved comparing three models, which implied considering the variance between groups through an adhoc test or by applying the Bonferroni correction to pairwise tests. Now that we are comparing only two models, a single test per comparison will be sufficient, necessitating changes to some tests and rendering the Bonferroni correction unnecessary. In essence, all previous rank-based tests that used the Friedman test to account for variability between groups will be replaced by the Wilcoxon signed-ranks test, as recommended by (Demšar, 2006). In summary, rank-based tests that were formerly used to evaluate macro and weighted averages at both the fold and dataset levels will now be performed using the Wilcoxon signed-ranks test. In the Semantic Textual Similarity (STS) task, should both normality and sphericity be validated through prior testing methods, the paired t-test will be conducted, otherwise, the Wilcoxon test will be utilized to determine significance.

5.3 Performance Analysis Aggregation

This section presents a comprehensive analysis of the performance comparison between TransBERT and cTransBERT across the four tasks: Classification, Named Entity Recognition (NER), Part-Of-Speech (POS), STS.

5.3.1 Classification Task Analysis

Starting with the classification task, Table 5.1 presents an aggregated summary of the main results of the task's datasets. TransBERT achieved for both aggregation methods the highest Precision, Recall, and F_1. While being small in general, the Precision difference between TransBERT and cTransBERT is slightly larger when considering the macro average, indicating a greater disparity for labels with lower support. Even though TransBERT exhibits a considerably increased Normalized Ranking Average (NRA), the difference lacks statistical significance. Therefore, we cannot assert that pre-training a LM with an adhoc tokenizer enhances classification performance.

	TransBERT			**cTransBERT**		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	**75.82**	**76.69**	**75.71**	75.10	76.05	74.70
Macro Avg	**64.06**	**62.55**	**61.93**	60.58	61.08	59.31
🥇 / 🥈	**9/113**	5/**128**	**6/114**	2/98	**6**/105	5/91
🥉 / 🏅	2/61	**6**/46	5/60	**9/76**	5/**69**	**6/83**
NRA	**55.95**	**55.95**	**56.49**	44.05	44.05	43.51

Table 5.1: Model Evaluation for the Classification Task (Tokenizer Analysis) - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different classes/labels for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Wilcoxon test. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

5.3.2 Named Entity Recognition Task Analysis

In the NER task, Table 5.2 shows that TransBERT achieves better weighted and macro averages for Precision, Recall, and F_1. While it used to be only about Precision in the classification task, once again, there is a notable difference in the macro average this time across all metrics, indicating that TransBERT performs even better on entities with low support.

	TransBERT			**cTransBERT**		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	**83.03****	**83.46****	**83.15****	81.02**	82.13**	81.44**
Macro Avg	**77.72****	**76.75****	**76.45****	67.16**	67.13**	66.51**
🥇 / 🥈	**38/184**	**38/196**	**40/185**	10/133	17/159	8/125
🥉 / 🏅	10/83	10/71	8/82	**38/134**	**31/108**	**40/142**
NRA	**62.54****	**59.21****	**64.60****	37.46**	40.79**	35.40**

Table 5.2: Model Evaluation for the Named Entity Recognition Task (Tokenizer Analysis) - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different entities for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Wilcoxon test. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

Observations on the NRA show a larger disparity, now confirmed with statistical significance. Moreover, the frequency in which TransBERT significantly outperforms cTransBERT is approximately ten times more than the previous and futur comparisons. Specifically, TransBERT achieved a statistically significant higher F_1 in 40 cases, which represents about 13% of the tested entities/folds. It is worth noting that more than half of these results were obtained in the same dataset, as shown in Table H.11. This table also highlights the significant performance disparity among entities with low support.

There is indeed an increase in performance in NER this time validated by a statistical test. As the only difference in the methodology is linked to the choice of tokenizer, we can conclude that for NER choosing an adhoc tokenizer to pre-train a model, at least for the life science domain, has an significant impact in the downstream task performance.

Now that the performance improvement in NER has been confirmed by a statistical test. Given that the only variation in methodology is the selection of tokenizer, it can be concluded that choosing a specialized tokenizer to pre-train a model, particularly in the life sciences field, significantly influences the downstream task performance. As our hypothesis statement is only about performance enhancement, we could move on to the next task, however, it might seem superficial to overlook the crucial step of tokenizing the named entity in this downstream task.

In summary, NER leverages token vector representations to determine entity detection, considering the context of surrounding words. The tokenizer is the only difference between TransBERT and cTransBERT, making it worthwhile to examine tokenization differences in real examples. Table 5.3 aims to provide insight into the entity length before tokenization, measured in words, and after tokenization, measured in tokens, for both TransTokenizer and CamemBERT's tokenizers. For

each cell, the first number indicates the average number of words per entity, which points with an arrow to the number of tokens each tokenizer produces per entity. The transformation rate is displayed above the arrow.

	TransTokenizer			**CamemBERT**		
	set(T)	**set(T)∩set(cT)**	**set(cT)**	**set(T)**	**set(T)∩set(cT)**	**set(cT)**
FN	$1.57 \xrightarrow{\times 1.8} \underline{2.85}$	$1.83 \xrightarrow{\times 1.7} \underline{3.07}$	$1.57 \xrightarrow{\times 1.6} \underline{2.58}$	$1.57 \xrightarrow{\times \mathbf{2.2}} \mathbf{3.50}$	$1.83 \xrightarrow{\times \mathbf{2.0}} \mathbf{3.63}$	$1.57 \xrightarrow{\times \mathbf{2.0}} \mathbf{3.17}$
FP	$1.57 \xrightarrow{\times 1.8} \underline{2.88}$	$1.55 \xrightarrow{\times 1.6} \underline{2.47}$	$1.52 \xrightarrow{\times 1.8} \underline{2.66}$	$1.57 \xrightarrow{\times \mathbf{2.1}} \mathbf{3.33}$	$1.55 \xrightarrow{\times \mathbf{2.0}} \mathbf{3.13}$	$1.52 \xrightarrow{\times \mathbf{2.1}} \mathbf{3.18}$
TP	$1.57 \xrightarrow{\times 1.6} \underline{2.58}$	$1.27 \xrightarrow{\times 1.5} \underline{1.91}$	$1.57 \xrightarrow{\times 1.8} \underline{2.85}$	$1.57 \xrightarrow{\times \mathbf{2.0}} \mathbf{3.17}$	$1.27 \xrightarrow{\times \mathbf{1.8}} \mathbf{2.30}$	$1.57 \xrightarrow{\times \mathbf{2.2}} \mathbf{3.50}$

Table 5.3: Tokenization Difference Statistics - The table shows, along the column axis, the sets of TransBERT, cTransBERT, and their intersections for FN, FP, and TP. The initial number denotes the quantity of words per entity in a set, followed by an arrow pointing to the number of tokens per entity generated by each tokenizer. The figure above the arrow indicates the multiplication rate involved in the tokenization process.

For each model, predictions are compared to the true entities to form three sets: FN, FP, and TP. These sets are then compared across models to identify each possible set. Specifically, FN within set(T) represents the FN uniquely identified by TransBERT, while FN within the intersection set(T)∩set(cT) represents FN entities for both models. This division aims to highlight differences in tokenization rates between the sets as the prevailing hypothesis is that fewer tokens needed to represent an entity lead to more accurate results. Nevertheless, for any specific tokenizer, Table 5.3 does not reveal a significant difference across the sets. The only minor variance is in the overlapping TPs for both models, where they exhibit the lowest transformation rate among all, supporting the hypothesis that smaller entities tend to be classified more accurately.

The main takeaway from this table is that CamemBERT generally requires more tokens to represent the same entities, which may lead to incorrect entity classifications. An overall analysis indicates that, although CamemBERT requires approximately 21% more tokens, its ratio is 37% more variable than TransTokenizer, suggesting higher unpredictability when encountering unfamiliar words. Further examination of cTransBERT's misclassifications shows that the CamemBERT tokenizer may need up to 32 additional tokens compared to TransTokenizer for identical chemical compounds. As illustrated in Figure 5.1, this specific scenario is demonstrated, but there are additional instances involving similar chemical structures.

Given the extensive list of tokenizations that could be reviewed, Appendix I illustrates some of them by highlighting the key differences in tokenization length between TransTokenizer and CamemBERT. For this study, all cTransBERT FN and FP instances were analyzed. The computed ratio was obtained using both tokenizer ratios. Examples are presented in descending order of this ratio to emphasize the most notable cases. To keep the list concise, only 10 examples per ratio were randomly chosen.

Entity: ['3-[[6-[4-[(1,2,3,4,5,6-hexahydro-2-pyrimidinyl)iminocarbonyl]-1-pipéridinyl]-5-méthyl-4-pyrimidinyl]amino]-N-[(phénylméthoxy)carbonyl]alanine'] (1 word)
TransBERT: ['__3', '-[', '[', '6', '-[', '4', '-[(', '1,2,3', ',', '4,5', ',', '6-', 'hexa', 'hydro', '-2-', 'pyrimidin', 'yl', ')', 'imino', 'carbonyl', ']', '-1-', 'pi', 'péri', 'd', 'inyl', ']', '-5-', 'méthyl', '-4-', 'pyrimidin', 'yl', ']', 'amino', ']-', 'N', '-[(', 'phényl', 'méthoxy', ')', 'carbonyl', ']', 'alanine'] (43 tokens)
CamemBERT: ['__3', '-', '[', '[', '6', '-', '[', '4', '-', '[', '(', '1,', '2,', '3,4', ',', '5,6', '-', 'h', 'ex', 'a', 'hydro', '-2', '-', 'py', 'ri', 'midi', 'ny', 'l', ')', 'im', 'ino', 'car', 'bon', 'yl', ']', '-1', '-', 'pi', 'péri', 'din', 'yl', ']', '-5', '-', 'm', 'éthyl', '-4', '-', 'py', 'ri', 'midi', 'ny', 'l', ']', 'am', 'ino', ']', '-', 'N', '-', '[', '(', 'phé', 'ny', 'l', 'méth', 'oxy', ')', 'car', 'bon', 'yl', ']', 'a', 'lan', 'ine'] (Δ+32)

Figure 5.1: CamemBERT Vs TransTokenizer for Chemical Compounds - An example of tokenization shows that TransTokenizer requires 43 tokens to represent this entity, whereas CamemBERT needs 75 tokens.

5.3.3 Part-of-Speech Tagging Task Analysis

Table 5.4 demonstrates that TransBERT achieves higher scores across all metrics and aggregation methods. Predictably, the results are very close with no significant differences. TransBERT achieves a marginally higher number of wins across all metrics but falls short by one statistically significant positive result. Consequently, even with a higher NRA, the difference between the two models in this task is minimal.

	TransBERT			**cTransBERT**		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	**98.33**	**98.30**	**98.31**	98.31	98.29	98.29
Macro Avg	**92.51**	**90.14**	**90.86**	92.29	89.43	90.23
🥇 / 🥈	**6**/**200**	6/**209**	**6**/**191**	**6**/185	**7**/205	**6**/174
🥉 / 🏅	**5**/84	**5**/75	**5**/93	**5**/**99**	4/**79**	**5**/**110**
NRA	**52.54**	**50.51**	**52.88**	47.46	49.49	47.12

Table 5.4: Model Evaluation for the Part-of-Speech Tagging Task (Tokenizer Analysis) - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different tags for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Wilcoxon test. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

5.3.4 Semantic Textual Similarity Task Analysis

Results in Table 5.4 indicate that cTransBERT achieves a higher R^2 averaged across folds by a narrow margin. It also secures relatively more wins, attaining

statistical significance twice, compared to once for TransBERT. Nevertheless, due to the limited number of experiments conducted in this task, this performance does not establish statistical significance.

	TransBERT	**cTransBERT**
	R^2	R^2
Weighted Avg	83.04	**84.36**
Macro Avg	83.04	**84.36**
🥇 / 🥈	1/2	**2/5**
🥉 / 🏅	**2/5**	1/2
NRA	30.00	**70.00**

Table 5.5: Model Evaluation for the Semantic Textual Similarity Task (Tokenizer Analysis) - This table presents the weighted and macro aggregations for R^2 for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Wilcoxon test. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

5.4 Conclusion & Future Works

Initially, the plan for this chapter was to compare both models on a dataset-by-dataset basis. However, after performing several comparisons, it became evident that there were overlaps with Chapter 4, which had already examined each dataset individually. As a result, the focus shifted to conducting only aggregated analysis. With a focus on a tokenizer analysis when significance was detected.

As outlined in the description of the experimental setup, this chapter employs the same methodology as the previous one, which implies it shares the same limitations. Although there is no distinct difference between TransBERT and cTransBERT in classification tasks, the observed disparity in NER is remarkable in both metrics and statistical significance. As a reminder, the only difference which stand between both models was the use of a tokenizer, which was trained on a mainstream French corpus. From a broader perspective, there is approximately a 10-point difference in the $F_{1_{macro}}$, which aligns with the previously observed performance of CamemBERT and DrBERT. For the remaining two tasks, there are no noteworthy significant differences to report.

Even in a brief chapter, it's essential to recall the research hypothesis formulated earlier:

> **Domain-Specific (DS) tokenization enhances the performance of Pre-trained Language Models (PLMs) on specialized downstream tasks.**

Even though enhancements were noted in just one of the four tasks reviewed, there's a notable enhancement in NER performance, with no reduction in other areas, supporting the hypothesis of our chapter. While a segmentation analysis has been performed, a comprehensive investigation into the reasons behind such

an enhancement in a token representation-based task could still be carried out by examining the signal at the token level.

It is essential to highlight the particular configuration of our experiment. Although integrating a DS tokenizer prior to pre-training with a DS corpus shows improvements, it does not ensure the same enhancement when training on a non-DS corpus, despite evidence suggesting this potential. Indeed, although no experiment directly supports this hypothesis, it can be deduced from the fact that TransBERT significantly outperforms CamemBERT (Chapter 4) in the NER task while cTransBERT performs only on the same level as CamemBERT (Table J.2). In other words, it implies that tokenization has a significant impact in NER task as TransBERT significantly outperforms cTransBERT. Therefore, examining a model pre-trained on the CamemBERT corpus with the TransTokenizer would likely yield better results in at least DS NER datasets. The question remains if it would be competitive with a model pre-trained on a DS corpus as it could be a compound effect. Although this might seem a bit trivial, the data and computational power required to train a tokenizer are very low, and these findings could allow better pre-training of DS LM by only pre-training a LM on a generic corpus using the DS tokenizer.

Chapter 6

Discussion & Conclusion

While Chapter 3 was mainly devoted to the development of modules that, despite requiring a significant investment of time and delivering few immediate metrics, played a pivotal role in underpinning our research, Chapter 4 and 5 ultimately provided crucial insights fundamental to this thesis. This chapter synthesizes the primary findings of the research, examines their limitations, and proposes directions for future study, concluding with a recap of this thesis's contributions.

6.1 TransBERT: A Synthetically Translated Language Model

Although having multiple chapters that address parallel research questions is often standard in the field, it is essential to acknowledge that Chapter 3 serves as a foundation for upcoming research issues where substantial work was reviewed, although it did not produce directly exploitable results, such as sentence-by-sentence translation evaluations through abstracts. This groundwork significantly contributes to pre-training the first Language Model (LM) utilizing exclusively artificially translated data. In effect, Chapter 4 represents the culmination of a sequence of modules, integrating all previous efforts, detailed choices, and comparative analyses.

In order to validate the hypothesis that the current state of Machine Translation (MT) supports the development of a LM trained on a machine-translated corpus, we created a rigorous experimental framework. This setup provided a uniform treatment for all the models that were compared, with the Pre-trained Language Model (PLM) being the sole variable change across the datasets and folds tested. We also improved the benchmark by adding 5-folds cross-validation, Hyperparameter Optimization (HPO), and other adjustments to increase its robustness. Furthermore, we set up an in-depth statistical testing procedure, which was meticulously applied at multiple levels. This approach allowed us not only to discern minor differences between models, but also to pinpoint statistically significant variations from smaller ones in an ad hoc reporting system.

A detailed benchmark analysis with its English counterpart shows that Dr-Benchmark is quite representative of the two main life science benchmarks, covering a wide array of tasks and including a significant number of datasets. The findings strongly support our hypothesis, especially in classification and Named Entity Recognition (NER) tasks, which are crucial in the life science sector. More specifi-

cally, TransBERT exhibited statistically significant superiority over DrBERT in all tasks. Furthermore, it surpassed CamemBERT with statistical significance in classification and NER tasks, while achieving similar performance in Part-Of-Speech (POS) tagging and Semantic Textual Similarity (STS) tasks. Remarkably, TransBERT's performance consistently remained strong, never being statistically outperformed by any other model in any task. These results highlight the robustness and adaptability of TransBERT in managing diverse linguistic challenges within the life science field.

The key takeaway message from Chapter 4 is that a BERT-like LM can indeed be pre-trained using entirely synthetically translated data, and when matched in size and translation quality, the resulting PLM can outperform the performance of a general-domain native PLM. As mentioned in Chapter 4, while it would be insightful to specify the exact amount of data or quality of translation necessary, the cost of exploring this research question is beyond the scope of this thesis.

6.2 The Impact of Domain-Specific Tokenization on Pre-trained Language Models Performance

In order to confirm the hypothesis that Domain-Specific (DS) tokenization improves the effectiveness of PLMs on specialized downstream tasks, we used a rigorous experimental approach to compare two models: TransBERT and cTransBERT. This comparative study employed the framework laid out in Chapter 4, with a significant adjustment in the statistical testing procedure to allow for pairwise comparisons. The experimental setup relied on the principle that the only difference between TransBERT and cTransBERT lies in the use of a domain-specific tokenizer.

By controlling this variable, it becomes possible to isolate and evaluate how tokenization uniquely impacts model performance across various life science applications. Upon reviewing the results for each task, it was discovered that the DS tokenizer resulted in significant improvements in the NER task, reaching statistical significance. Although other tasks exhibited slight performance differences, these were not statistically significant. This comprehensive analysis suggests that the impact of DS tokenization differs according to the task.

A deeper analysis of the tokenizers' outputs showed that CamemBERT regularly needed a significantly greater number of tokens compared to TransTokenizer to encode similar sequences. This finding affects computational efficiency and may influence the model's capacity to grasp DS subtleties in the text. This outcome is consistent with the common beliefs that the tokenizer DS is crucial to improving the efficiency of PLMs, particularly in niche fields such as life sciences.

6.3 Limitations & Discussion

This section covers the limitations concerning the interpretation of the findings, whereas limitations associated with the experimental design are addressed in the conclusion of each chapter.

6.3.1 In-Domain/Language Generalization

While our benchmark study presents strong evidence for the effectiveness of our proposed model across various datasets, it is important to note the limitations in generalizing these findings. Although our benchmark was meticulously designed to cover a wide array of tasks within the life sciences domain, it cannot comprehensively represent every possible scenario or use case.

One major limitation lies in the wide variety of Natural Language Processing (NLP) tasks and the continually evolving nature of scientific language. Even though our benchmark includes a broad range of datasets and tasks, it is impossible to cover every potential application or future development in the field. The performance of our model, while impressive within the scope of our study, may not necessarily be consistent across all possible tasks or datasets in the life sciences domain.

Additionally, the idea of a universally 'best' model is inherently flawed in the realm of NLP. Different models might excel in particular contexts or specific types of tasks, and their performance can be affected by factors such as domain specificity, data distribution, and the nuances of individual use cases. What works optimally in one scenario may not be the best choice in another, emphasizing the need for context-specific model evaluation and selection.

It is also important to recognize that the fast-paced advancements in NLP research could lead to new architectures, pre-training techniques, or fine-tuning strategies that may surpass our current model in certain aspects. The dynamic nature of the field requires ongoing evaluation and comparison against new innovations.

Moreover, although we aim for representativeness in our benchmark, it may unintentionally include biases or limitations in dataset selection or task formulation. These potential biases could affect the generalizability of our findings to real-world applications or to datasets significantly different from those in our benchmark.

In light of these factors, a nuanced interpretation of our results is essential. Although our model shows considerable promise and outperforms existing State-of-the-Art (SOTA) models in several datasets or tasks, it should be seen as a competitive option rather than an absolute universal solution. We recommend further testing in various real-world scenarios and continuous evaluation against emerging models and methodologies.

Ultimately, the choice of an appropriate model should be driven by the specific requirements of the task at hand, the nature of the available data, and a thorough understanding of the strengths and limitations of the model. Our study offers valuable information to aid in this decision-making process, but should be considered along with other relevant factors and ongoing research in the field.

6.3.2 Other Domains Generalization

Although our model, which was trained on translated synthetic data within the life sciences corpus, shows encouraging generalization towards other domains, it is important to recognize the constraints when extrapolating these results to other areas. The success of our method in addressing the lack of native language data in life sciences should not be automatically expected to apply to other specialized sectors such as finance, law, or engineering. Each field presents its own unique linguistic hurdles, specialized terminologies, and DS conceptual frameworks that

general-purpose machine translation systems might not handle effectively. The quality and relevance of translated synthetic data can differ greatly between domains, possibly affecting the model's performance and dependability. Moreover, the subtleties of DS language use, such as idiomatic phrases, technical lingo, and context-dependent meanings, may not be accurately preserved in translated data, which could lead to misunderstandings or errors in other fields. Additionally, the success of our approach may depend on the degree to which translatable concepts are within a given domain, which can vary greatly. For example, concepts that are highly specific to a culture or legally bound in sectors like law or social sciences might pose particular difficulties for this approach. Hence, while our results suggest a promising avenue for mitigating language resource shortages in specialized fields, further research is essential to confirm the broad applicability of this method across various domains, each with its own distinct linguistic and conceptual challenges.

6.3.3 Other Languages Generalization

Although our study highlights the effectiveness of employing synthetic translated data for training LMs in the field of life sciences in French, caution is warranted when applying these findings to other languages, especially those with limited resources. We believe that the success of our method is highly dependent on the quality and availability of machine translation systems for the target language, which can differ greatly among various language pairs. Even if M2M-100 has a great potential to secure relatively great results in low-resource languages compared to other models, some language pairs often lack strong machine translation models, which can undermine the quality of the translated synthetic data. Additionally, the linguistic gap between the source language and the target language can greatly affect the effectiveness of the approach. Languages with different syntactic frameworks, morphological structures, or writing systems might pose additional difficulties in maintaining semantic subtleties and DS language during translation. Furthermore, the cultural and scientific context embedded in the original material might not always have direct counterparts in the target language or culture, which could result in meaning loss or the introduction of biases. The degree of standardization in scientific terminology across languages may also play a role in the consistency and accuracy of the translated corpus. Moreover, the success of this technique could be influenced by the target language's scientific literature environment and how well it integrates with global scientific discourse. Therefore, although our findings indicate a potential solution for addressing the deficit of scientific corpora in some languages, the method's suitability across different linguistic contexts, especially for low-resource languages, requires thorough evaluation and additional investigation.

6.3.4 Generative Language Models

Although generative Large Language Models (LLMs) exhibit exceptional abilities in various NLP tasks, it is essential to account for the considerable costs associated with their creation and use when assessing their practical value. These models achieve outstanding performance, often outperforming task-specific models in fields such as NER and classification. However, the economic implications of employing generative LLMs for such tasks are significant and complex. The fixed expenses for

training these models are enormous, frequently amounting to millions of dollars due to the vast computational resources required. Moreover, the variable costs of inference, such as energy consumption and cloud computing fees, can be excessively high for many organizations, particularly for large-scale or real-time applications. The significant disparity in resource requirements between generative and Natural Language Understanding (NLU) models raises crucial questions about sustainability and accessibility.

Although generative models provide exceptional versatility, the economic and environmental costs of using them for relatively simple tasks like binary classification might be disproportionate to the slight improvements in performance they deliver. This imbalance in cost versus benefit highlights the need for a more refined approach to model selection that considers not only raw performance metrics but also economic efficiency, environmental impact, and long-term sustainability. As the field advances, it is increasingly vital to create strategies that capitalize on the strengths of generative models while reducing their resource-intensive nature, possibly through methods such as model distillation or task-specific fine-tuning of smaller and more efficient models.

6.4 Future Works

Although our work has provided important information on the use of translated synthetic data for training LMs within the field of life sciences, it has generated more research questions than definitive answers. This result emphasizes the intricate and dynamic nature of NLP in specialized areas. The issues prompted by our study span several aspects of machine translation, domain adaptation, and the interaction between artificial and natural language data. These emerging research paths underline the necessity for ongoing exploration into the subtleties of cross-lingual and cross-domain knowledge transfer in language models.

One encouraging direction for future research is to expand our approach to encompass a wider array of languages, especially those that are underrepresented in the life sciences field. Applying our methodology across various linguistic settings will help us better understand its generalizability and any possible constraints. Additionally, creating multilingual models capable of managing several languages within the life sciences sector poses an intriguing challenge. These models might exploit cross-lingual knowledge transfer, allowing for a more efficient use of scarce data resources and promoting a more inclusive global scientific community.

Another path for future research is an extensive comparison between our method and the latest generative LLMs on identical datasets. Such a comparison would yield valuable understanding of the trade-offs between specialized, domain-focused models and more general, resource-heavy models LLMs. Assessing performance, efficiency, and cost-effectiveness across different life science tasks would help researchers and practitioners in making informed decisions. Furthermore, this analysis could highlight the possibility of integrating the strengths of both approaches.

A promising direction for upcoming research involves exploring the use of generative LLMs to create synthetic data for training DS models, as an alternative to our translation-based method. This approach could yield more varied and nuanced datasets, encapsulating intricate domain-specific knowledge and linguistic patterns. Assessing the quality, reliability, and possible biases of LMs-generated

synthetic data in comparison to translated data could offer valuable insights into data augmentation strategies for low-resource domains and languages. Furthermore, this method could be expanded to explore the potential of LMs in generating multilingual synthetic data, potentially addressing some of the challenges related to cross-lingual knowledge transfer.

6.5 Thesis Contribution

This thesis presents several groundbreaking contributions to the field of NLP in the life sciences, with a specific emphasis on the French language. At the core of this research lies TransCorpus, an innovative 30GB corpus composed of 22M abstracts automatically translated from English to French. It stands as a leading large-scale DS corpus of its kind, created entirely through machine translation. On top of it stands TransBERT and its corresponding TransTokenizer, a LM/tokenizer duo solely trained on TransCorpus. To our knowledge, this is the first occurrence of a LM being trained entirely on automatically translated data in the life sciences domain of this scale. The creation of TransBERT highlights the potential of machine translation to address the shortage of resources in languages with limited DS corpus. Furthermore, a forthcoming paper will provide detailed information on the performance and implications of TransBERT, offering valuable information to the scientific community. These resources, TransCorpus, TransBERT and TransTokenizer, will be made available to the public, providing researchers and practitioners with powerful tools to advance NLP in the field of life sciences in French. This work addresses a significant gap in language resources and facilitates further exploration in cross-lingual knowledge transfer and DS language modeling.

Bibliography

Almeida, T., Jonker, R. A. A., Poudel, R., Silva, J. M., and Matos, S. (2023). BIT.UA at BioASQ 11B: Two-Stage IR with Synthetic Training and Zero-Shot Answer Generation. In Conference and Labs of the Evaluation Forum.

Artetxe, M. and Schwenk, H. (2019). Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. Transactions of the Association for Computational Linguistics, 7:597–610.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. arXiv preprint arXiv:1409.0473.

Bahl, L. R., Jelinek, F., and Mercer, R. L. (1983). A Maximum Likelihood Approach to Continuous Speech Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5(2):179–190.

Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). A Neural Probabilistic Language Model. J. Mach. Learn. Res., 3(null):1137–1155.

Bergman, E., Dürlich, L., Arthurson, V., Sundström, A., Larsson, M., Bhuiyan, S., et al. (2023). BERT Based Natural Language Processing for Triage of Adverse Drug Reaction Reports Shows Close to Human-Level Performance. PLOS Digital Health, 2(12):e0000409.

Bird, S., Klein, E., and Loper, E. (2009). Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit. O'Reilly Media, Inc.

Britz, D., Goldie, A., Luong, M.-T., and Le, Q. (2017). Massive Exploration of Neural Machine Translation Architectures.

Brown, P., Cocke, J., Della Pietra, S., Della Pietra, V., Jelinek, F., Mercer, R., and Roossin, P. (1988). A Statistical Approach to Language Translation. In Coling Budapest 1988 Volume 1: International Conference on Computational Linguistics.

Brown, P. F., Cocke, J., Della Pietra, S. A., Della Pietra, V. J., Jelinek, F., Lafferty, J. D., Mercer, R. L., and Roossin, P. S. (1990). A Statistical Approach to Machine Translation. Computational Linguistics, 16(2):79–85.

Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., and Mercer, R. L. (1993). The Mathematics of Statistical Machine Translation: Parameter Estimation. Computational Linguistics, 19(2):263–311.

Brown, P. F., Della Pietra, V. J., deSouza, P. V., Lai, J. C., and Mercer, R. L. (1992). Class-Based n-gram Models of Natural Language. Computational Linguistics, 18(4):467–480.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language Models Are Few-Shot Learners.

Cardon, R., Grabar, N., Grouin, C., and Hamon, T. (2020). Présentation De La Campagne d'évaluation DEFT 2020 : similarité Textuelle En Domaine Ouvert Et Extraction d'information précise Dans Des Cas Cliniques (Presentation of the DEFT 2020 Challenge : Open Domain Textual Similarity and Precise Information Extraction From Clinical Cases). In Cardon, R., Grabar, N., Grouin, C., and Hamon, T., editors, Actes de la 6e conférence conjointe Journées d'Études sur la Parole (JEP, 33e édition), Traitement Automatique des Langues Naturelles (TALN, 27e édition), Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL, 22e édition). Atelier DÉfi Fouille de Textes, pages 1–13, Nancy, France. ATALA et AFCP.

Carrino, C. P., Armengol-Estapé, J., Gutiérrez-Fandiño, A., Llop-Palao, J., Pàmies, M., Gonzalez-Agirre, A., and Villegas, M. (2021). Biomedical and Clinical Language Models for Spanish: On the Benefits of Domain-Specific Pretraining in a Mid-Resource Scenario.

Caswell, I., Chelba, C., and Grangier, D. (2019). Tagged Back-Translation. In Bojar, O., Chatterjee, R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Martins, A., Monz, C., Negri, M., Névéol, A., Neves, M., Post, M., Turchi, M., and Verspoor, K., editors, Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers), pages 53–63, Florence, Italy. Association for Computational Linguistics.

Cañete, J., Chaperon, G., Fuentes, R., Ho, J.-H., Kang, H., and Pérez, J. (2023). Spanish Pre-trained BERT Model and Evaluation Data.

Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., and Androutsopoulos, I. (2020). LEGAL-BERT: The Muppets Straight Out of Law School. In Cohn, T., He, Y., and Liu, Y., editors, Findings of the Association for Computational Linguistics: EMNLP 2020, pages 2898–2904, Online. Association for Computational Linguistics.

Chen, Q., Allot, A., and Lu, Z. (2020). LitCovid: an Open Database of COVID-19 Literature. Nucleic Acids Research, 49(D1):D1534–D1540.

Cheng, J., Dong, L., and Lapata, M. (2016). Long Short-Term Memory-Networks for Machine Reading. CoRR, abs/1601.06733.

Chitnis, R. and DeNero, J. (2015). Variable-Length Word Encodings for Neural Translation Models. In Proceedings of the 2015 Conference on Empirical

Methods in Natural Language Processing, pages 2088–2093, Lisbon, Portugal. Association for Computational Linguistics.

Cho, K., van Merrienboer, B., Bahdanau, D., and Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder-Decoder Approaches.

Collobert, R. and Weston, J. (2008). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In Proceedings of the 25th International Conference on Machine Learning, ICML '08, page 160–167, New York, NY, USA. Association for Computing Machinery.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2019). Unsupervised Cross-lingual Representation Learning at Scale. CoRR, abs/1911.02116.

Consortium, T. U. (2022). UniProt: the Universal Protein Knowledgebase in 2023. Nucleic Acids Research, 51(D1):D523–D531.

Copara, J., Knafou, J., Naderi, N., Moro, C., Ruch, P., and Teodoro, D. (2020a). Contextualized French Language Models for Biomedical Named Entity Recognition. In Actes de la 6e conférence conjointe Journées d'Études sur la Parole (JEP, 33e édition), Traitement Automatique des Langues Naturelles (TALN, 27e édition), Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL, 22e édition). Atelier DÉfi Fouille de Textes, pages 36–48, Nancy, France. ATALA et AFCP.

Copara, J., Naderi, N., Knafou, J., Ruch, P., and Teodoro, D. (2020b). Named Entity Recognition in Chemical Patents Using Ensemble of Contextual Language Models.

Cui, Y., Che, W., Liu, T., Qin, B., and Yang, Z. (2021). Pre-Training With Whole Word Masking for Chinese BERT. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29:3504–3514.

Dalloux, C., Claveau, V., Grabar, N., Oliveira, L. E. S., Moro, C. M. C., Gumiel, Y. B., and Carvalho, D. R. (2021). Supervised Learning for the Detection of Negation and of Its Scope in French and Brazilian Portuguese Biomedical Corpora. Natural Language Engineering, 27(2):181–201.

Demner-Fushman, D., Chapman, W. W., and McDonald, C. J. (2009). What Can Natural Language Processing Do for Clinical Decision Support? Journal of biomedical informatics, 42(5):760–772.

Demšar, J. (2006). Statistical Comparisons of Classifiers Over Multiple Data Sets. J. Mach. Learn. Res., 7:1–30.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805.

Dietterich, T. G. (1998). Approximate Statistical Test For Comparing Supervised Classification Learning Algorithms. Neural Computation, 10(7):1895–1923.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2020). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. CoRR, abs/2010.11929.

Dunn, O. J. (1961). Multiple Comparisons Among Means. Journal of the American Statistical Association, 56(293):52–64.

El-Kishky, A., Chaudhary, V., Guzmán, F., and Koehn, P. (2020). CCAligned: A Massive Collection of Cross-Lingual Web-Document Pairs. In Webber, B., Cohn, T., He, Y., and Liu, Y., editors, Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 5960–5969, Online. Association for Computational Linguistics.

Fan, A., Bhosale, S., Schwenk, H., Ma, Z., El-Kishky, A., Goyal, S., Baines, M., Celebi, O., Wenzek, G., Chaudhary, V., Goyal, N., Birch, T., Liptchinsky, V., Edunov, S., Grave, E., Auli, M., and Joulin, A. (2020). Beyond English-Centric Multilingual Machine Translation. CoRR, abs/2010.11125.

Friedman, M. (1937). The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. Journal of the American Statistical Association, 32(200):675–701.

Gage, P. (1994). A New Algorithm for Data Compression. C Users J., 12(2):23–38.

Grabar, N., Claveau, V., and Dalloux, C. (2018). CAS: French Corpus with Clinical Cases. In Lavelli, A., Minard, A.-L., and Rinaldi, F., editors, Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis, pages 122–128, Brussels, Belgium. Association for Computational Linguistics.

Gu, J., Wang, Y., Cho, K., and Li, V. O. (2019). Improved Zero-shot Neural Machine Translation via Ignoring Spurious Correlations. In Korhonen, A., Traum, D., and Màrquez, L., editors, Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1258–1268, Florence, Italy. Association for Computational Linguistics.

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. (2020). Domain-specific Language Model Pretraining for Biomedical Natural Language Processing. arXiv preprint arXiv:2007.15779.

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. (2021). Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. ACM Transactions on Computing for Healthcare, 3(1):1–23.

He, P., Liu, X., Gao, J., and Chen, W. (2021). DeBERTa: Decoding-enhanced BERT with Disentangled Attention.

Hiebel, N., Ferret, O., Fort, K., and Névéol, A. (2022). CLISTER : A Corpus for Semantic Textual Similarity in French Clinical Narratives. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Odijk, J., and Piperidis, S., editors,

Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 4306–4315, Marseille, France. European Language Resources Association.

Isbister, T., Carlsson, F., and Sahlgren, M. (2021). Should We Stop Training More Monolingual Models, and Simply Use Machine Translation Instead?

Joshi, R. (2022). L3Cube-HindBERT and DevBERT: Pre-Trained BERT Transformer Models for Devanagari Based Hindi and Marathi Languages. arXiv preprint arXiv:2211.11418.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent Continuous Translation Models. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 1700–1709, Seattle, Washington, USA. Association for Computational Linguistics.

Khan, S. A., Tyrchan, C., and Moreau, Y. (2022). AI-based Language Models Powering Drug Discovery and Development. Drug discovery today, 27(5):1274–1284.

Kingma, D. P. and Ba, J. (2017). Adam: A Method for Stochastic Optimization.

Knafou, J., Haas, Q., Borissov, N., Counotte, M., Low, N., Imeri, H., Ipekci, A. M., Buitrago-Garcia, D., Heron, L., Amini, P., et al. (2023). Ensemble of Deep Learning Language Models to Support the Creation of Living Systematic Reviews for the COVID-19 Literature. Systematic Reviews, 12(1):94.

Knafou, J., Naderi, N., Copara, J., Teodoro, D., and Ruch, P. (2020). BiTeM at WNUT 2020 Shared Task-1: Named Entity Recognition Over Wet Lab Protocols Using an Ensemble of Contextual Language Models. In Xu, W., Ritter, A., Baldwin, T., and Rahimi, A., editors, Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020), pages 305–313, Online. Association for Computational Linguistics.

Kocabiyikoglu, A. C., Portet, F., Gibert, P., Blanchon, H., Babouchkine, J.-M., and Gavazzi, G. (2022). A Spoken Drug Prescription Dataset in French for Spoken Language Understanding.

Kors, J. A., Clematide, S., Akhondi, S. A., van Mulligen, E. M., and Rebholz-Schuhmann, D. (2015). A Multilingual Gold-Standard Corpus for Biomedical Concept Recognition: the Mantra GSC. Journal of the American Medical Informatics Association, 22(5):948–956.

Kudo, T. (2018). Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

Kudo, T. and Richardson, J. (2018). SentencePiece: A Simple and Language Independent Subword Tokenizer and Detokenizer for Neural Text Processing. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

la Rosa, J. D., Ponferrada, E. G., Romero, M., Villegas, P., de Prado Salas, P. G., and Grandury, M. (2022). BERTIN: Efficient Pre-Training of a Spanish Language Model Using Perplexity Sampling. Procesamiento del Lenguaje Natural, 68(0):13–23.

Labrak, Y., Bazoge, A., Daille, B., Dufour, R., Morin, E., and Rouvier, M. (2023a). Tâches Et systèmes De détection Automatique Des réponses Correctes Dans Des QCMs liés Au Domaine médical : présentation De La Campagne DEFT 2023. In JEPTALNRECITAL.

Labrak, Y., Bazoge, A., Dufour, R., Daille, B., Gourraud, P.-A., Morin, E., and Rouvier, M. (2022). FrenchMedMCQA: A French Multiple-Choice Question Answering Dataset for Medical Domain. In Lavelli, A., Holderness, E., Jimeno Yepes, A., Minard, A.-L., Pustejovsky, J., and Rinaldi, F., editors, Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI), pages 41–46, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Labrak, Y., Bazoge, A., Dufour, R., Rouvier, M., Morin, E., Daille, B., and Gourraud, P.-A. (2023b). DrBERT: A Robust Pre-trained Model in French for Biomedical and Clinical Domains. In Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics (ACL'23), Long Paper, Toronto, Canada. Association for Computational Linguistics.

Labrak, Y., Bazoge, A., El Khettari, O., Rouvier, M., Constant Dit Beaufils, P., Grabar, N., Daille, B., Quiniou, S., Morin, E., Gourraud, P.-a., and Dufour, R. (2024). DrBenchmark: A Large Language Understanding Evaluation Benchmark for French Biomedical Domain. In Fourteenth Language Resources and Evaluation Conference (LREC-COLING 2024), Torino, Italy. Nicoletta Calzolari and Min-Yen Kan.

Labrak, Y., Rouvier, M., and Dufour, R. (2023c). MORFITT : A Multi-Label Corpus of French Scientific Articles in the Biomedical Domain. In 30e Conférence sur le Traitement Automatique des Langues Naturelles (TALN) Atelier sur l'Analyse et la Recherche de Textes Scientifiques, Paris, France. Florian Boudin.

Lample, G. and Conneau, A. (2019). Cross-lingual Language Model Pretraining. arXiv preprint arXiv:1901.07291.

Lauby-Secretan, B., Dossus, L., Marant-Micallef, C., and His, M. (2019). Obésité Et Cancer. Bulletin du Cancer, 106(7):635–646.

Le, H., Vial, L., Frej, J., Segonne, V., Coavoux, M., Lecouteux, B., Allauzen, A., Crabbé, B., Besacier, L., and Schwab, D. (2020). FlauBERT: Unsupervised Language Model Pre-training for French. In Proceedings of the 12th Language Resources and Evaluation Conference, pages 2479–2490, Marseille, France. European Language Resources Association.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2019). BioBERT: a Pre-Trained Biomedical Language Representation Model for Biomedical Text Mining. Bioinformatics.

Lin, Z., Feng, M., dos Santos, C. N., Yu, M., Xiang, B., Zhou, B., and Bengio, Y. (2017). A Structured Self-attentive Sentence Embedding.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. CoRR, abs/1907.11692.

Lothritz, C., Lebichot, B., Allix, K., Veiber, L., Bissyande, T., Klein, J., Boytsov, A., Lefebvre, C., and Goujon, A. (2022). LuxemBERT: Simple and Practical Data Augmentation in Language Model Pre-Training for Luxembourgish. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Odijk, J., and Piperidis, S., editors, Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 5080–5089, Marseille, France. European Language Resources Association.

Luong, T., Sutskever, I., Le, Q., Vinyals, O., and Zaremba, W. (2015). Addressing the Rare Word Problem in Neural Machine Translation. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 11–19, Beijing, China. Association for Computational Linguistics.

Magnini, B., Altuna, B., Lavelli, A., Speranza, M., and Zanoli, R. (2020). The E3C Project: Collection and Annotation of a Multilingual Corpus of Clinical Cases, pages 258–264.

Martin, L., Muller, B., Ortiz Suárez, P. J., Dupont, Y., Romary, L., de la Clergerie, É., Seddah, D., and Sagot, B. (2020). CamemBERT: a Tasty French Language Model. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7203–7219, Online. Association for Computational Linguistics.

Mays, E., Damerau, F. J., and Mercer, R. L. (1991). Context Based Spelling Correction. Information Processing & Management, 27(5):517–522.

McPhie, P. (1975). The Origin of the Alkaline Inactivation of Pepsinogen. Biochemistry, 14(24):5253—5256.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013b). Distributed Representations of Words and Phrases and Their Compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13, page 3111–3119, Red Hook, NY, USA. Curran Associates Inc.

Mnih, A. and Hinton, G. E. (2008). A Scalable Hierarchical Distributed Language Model. Advances in neural information processing systems, 21:1081–1088.

Morin, F. and Bengio, Y. (2005). Hierarchical Probabilistic Neural Network Language Model. In Cowell, R. G. and Ghahramani, Z., editors, Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics, volume R5 of Proceedings of Machine Learning Research, pages 246–252. PMLR. Reissued by PMLR on 30 March 2021.

Mottin, L., Goldman, J.-P., Jäggli, C., Achermann, R., Gobeill, J., Knafou, J., Ehrsam, J., Wicky, A., Gérard, C. L., Schwenk, T., Charrier, M., Tsantoulis, P., Lovis, C., Leichtle, A., Kiessling, M. K., Michielin, O., Pradervand, S., Foufi, V., and Ruch, P. (2023). Multilingual RECIST Classification of Radiology Reports Using Supervised Learning. Frontiers in Digital Health, 5.

Naderi, N., Knafou, J., Copara, J., Ruch, P., and Teodoro, D. (2021). Ensemble of Deep Masked Language Models for Effective Named Entity Recognition in Health and Life Science Corpora. Frontiers in research metrics and analytics, 6:689803.

Nentidis, A., Katsimpras, G., Krithara, A., López, S. L., Farré-Maduell, E., Gasco, L., Krallinger, M., and Paliouras, G. (2023). Overview of BioASQ 2023: The Eleventh BioASQ Challenge on Large-Scale Biomedical Semantic Indexing and Question Answering.

Ngo, C., Trinh, T. H., Phan, L., Tran, H., Dang, T., Nguyen, H., Nguyen, M., and Luong, M.-T. (2022). MTet: Multi-domain Translation for English and Vietnamese.

Nicholson, D. N. and Greene, C. S. (2020). Constructing Knowledge Graphs and Their Biomedical Applications. Computational and structural biotechnology journal, 18:1414–1428.

Névéol, A., Grouin, C., Leixa, J., Rosset, S., and Zweigenbaum, P. (2014). The QUAERO French Medical Corpus: A Ressource for Medical Entity Recognition and Normalization. In Proc of BioTextMining Work, pages 24–30.

OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., Berdine, J., Bernadett-Shapiro, G., Berner, C., Bogdonoff, L., Boiko, O., Boyd, M., Brakman, A.-L., Brockman, G., Brooks, T., Brundage, M., Button, K., Cai, T., Campbell, R., Cann, A., Carey, B., Carlson, C., Carmichael, R., Chan, B., Chang, C., Chantzis, F., Chen, D., Chen, S., Chen, R., Chen, J., Chen, M., Chess, B., Cho, C., Chu, C., Chung, H. W., Cummings, D., Currier, J., Dai, Y., Decareaux, C., Degry, T., Deutsch, N., Deville, D., Dhar, A., Dohan, D., Dowling, S., Dunning, S., Ecoffet, A., Eleti, A., Eloundou, T., Farhi, D., Fedus, L., Felix, N., Fishman, S. P., Forte, J., Fulford, I., Gao, L., Georges, E., Gibson, C., Goel, V., Gogineni, T., Goh, G., Gontijo-Lopes, R., Gordon, J., Grafstein, M., Gray, S., Greene, R., Gross, J., Gu, S. S., Guo, Y., Hallacy, C., Han, J., Harris, J., He, Y., Heaton, M., Heidecke, J., Hesse, C., Hickey, A., Hickey, W., Hoeschele, P., Houghton, B., Hsu, K., Hu, S., Hu, X., Huizinga, J., Jain, S., Jain, S., Jang, J., Jiang, A., Jiang, R., Jin, H., Jin, D., Jomoto, S., Jonn, B., Jun, H., Kaftan, T., Łukasz Kaiser, Kamali, A., Kanitscheider, I., Keskar, N. S., Khan, T.,

Kilpatrick, L., Kim, J. W., Kim, C., Kim, Y., Kirchner, J. H., Kiros, J., Knight, M., Kokotajlo, D., Łukasz Kondraciuk, Kondrich, A., Konstantinidis, A., Kosic, K., Krueger, G., Kuo, V., Lampe, M., Lan, I., Lee, T., Leike, J., Leung, J., Levy, D., Li, C. M., Lim, R., Lin, M., Lin, S., Litwin, M., Lopez, T., Lowe, R., Lue, P., Makanju, A., Malfacini, K., Manning, S., Markov, T., Markovski, Y., Martin, B., Mayer, K., Mayne, A., McGrew, B., McKinney, S. M., McLeavey, C., McMillan, P., McNeil, J., Medina, D., Mehta, A., Menick, J., Metz, L., Mishchenko, A., Mishkin, P., Monaco, V., Morikawa, E., Mossing, D., Mu, T., Murati, M., Murk, O., Mély, D., Nair, A., Nakano, R., Nayak, R., Neelakantan, A., Ngo, R., Noh, H., Ouyang, L., O'Keefe, C., Pachocki, J., Paino, A., Palermo, J., Pantuliano, A., Parascandolo, G., Parish, J., Parparita, E., Passos, A., Pavlov, M., Peng, A., Perelman, A., de Avila Belbute Peres, F., Petrov, M., de Oliveira Pinto, H. P., Michael, Pokorny, Pokrass, M., Pong, V. H., Powell, T., Power, A., Power, B., Proehl, E., Puri, R., Radford, A., Rae, J., Ramesh, A., Raymond, C., Real, F., Rimbach, K., Ross, C., Rotsted, B., Roussez, H., Ryder, N., Saltarelli, M., Sanders, T., Santurkar, S., Sastry, G., Schmidt, H., Schnurr, D., Schulman, J., Selsam, D., Sheppard, K., Sherbakov, T., Shieh, J., Shoker, S., Shyam, P., Sidor, S., Sigler, E., Simens, M., Sitkin, J., Slama, K., Sohl, I., Sokolowsky, B., Song, Y., Staudacher, N., Such, F. P., Summers, N., Sutskever, I., Tang, J., Tezak, N., Thompson, M. B., Tillet, P., Tootoonchian, A., Tseng, E., Tuggle, P., Turley, N., Tworek, J., Uribe, J. F. C., Vallone, A., Vijayvergiya, A., Voss, C., Wainwright, C., Wang, J. J., Wang, A., Wang, B., Ward, J., Wei, J., Weinmann, C., Welihinda, A., Welinder, P., Weng, J., Weng, L., Wiethoff, M., Willner, D., Winter, C., Wolrich, S., Wong, H., Workman, L., Wu, S., Wu, J., Wu, M., Xiao, K., Xu, T., Yoo, S., Yu, K., Yuan, Q., Zaremba, W., Zellers, R., Zhang, C., Zhang, M., Zhao, S., Zheng, T., Zhuang, J., Zhuk, W., and Zoph, B. (2024). GPT-4 Technical Report.

Organization, W. H. (2015). International Statistical Classification of Diseases and Related Health Problems. World Health Organization, 10th revision, fifth edition, 2016 edition.

Ortiz Suárez, P. J., Sagot, B., and Romary, L. (2019). Asynchronous Pipelines for Processing Huge Corpora on Medium to Low Resource Infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Parikh, A. P., Täckström, O., Das, D., and Uszkoreit, J. (2016). A Decomposable Attention Model for Natural Language Inference. CoRR, abs/1606.01933.

Paulus, R., Xiong, C., and Socher, R. (2017). A Deep Reinforced Model for Abstractive Summarization.

Peng, Y., Yan, S., and Lu, Z. (2019). Transfer Learning in Biomedical Natural Language Processing: An Evaluation of BERT and ELMo on Ten Benchmarking

Datasets. In Demner-Fushman, D., Cohen, K. B., Ananiadou, S., and Tsujii, J., editors, Proceedings of the 18th BioNLP Workshop and Shared Task, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global Vectors for Word Representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 1532–1543.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). 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), pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Phan, L., Dang, T., Tran, H., Trinh, T. H., Phan, V., Chau, L. D., and Luong, M.-T. (2023). Enriching Biomedical Knowledge for Low-resource Language Through Large-scale Translation. In Vlachos, A. and Augenstein, I., editors, Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 3131–3142, Dubrovnik, Croatia. Association for Computational Linguistics.

Phan, L., Tran, H., Nguyen, H., and Trinh, T. H. (2022). ViT5: Pretrained Text-to-Text Transformer for Vietnamese Language Generation.

Project, C.-. O. A. (2020). Living Evidence on COVID-19.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. (2022). Robust Speech Recognition via Large-Scale Weak Supervision.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2023). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text.

Roberts, K., Alam, T., Bedrick, S., Demner-Fushman, D., Lo, K., Soboroff, I., Voorhees, E., Wang, L. L., and Hersh, W. R. (2021). Searching for Scientific Evidence in a Pandemic: An Overview of TREC-COVID.

Romanov, A. and Shivade, C. (2018). Lessons From Natural Language Inference in the Clinical Domain.

Salazar, J., Liang, D., Nguyen, T. Q., and Kirchhoff, K. (2020). Masked Language Model Scoring. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2020). DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter.

Schneider, E. T. R., de Souza, J. V. A., Knafou, J., Oliveira, L. E. S. e., Copara, J., Gumiel, Y. B., Oliveira, L. F. A. d., Paraiso, E. C., Teodoro, D., and Barra, C. M. C. M. (2020). BioBERTpt - A Portuguese Neural Language Model for Clinical Named Entity Recognition. In Rumshisky, A., Roberts, K., Bethard, S., and Naumann, T., editors, Proceedings of the 3rd Clinical Natural Language Processing Workshop, pages 65–72, Online. Association for Computational Linguistics.

Schuster, M. and Nakajima, K. (2012). Japanese and Korean Voice Search. In 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5149–5152.

Schwenk, H., Dechelotte, D., and Gauvain, J.-L. (2006). Continuous Space Language Models for Statistical Machine Translation. In Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pages 723–730, Sydney, Australia. Association for Computational Linguistics.

Schwenk, H., Wenzek, G., Edunov, S., Grave, E., and Joulin, A. (2020). CCMatrix: Mining Billions of High-Quality Parallel Sentences on the WEB.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Shrestha, M. (2021). Development of a Language Model for Medical Domain. masterthesis, Hochschule Rhein-Waal.

Souza, F., Nogueira, R., and Lotufo, R. (2020). BERTimbau: Pretrained BERT Models for Brazilian Portuguese. In Cerri, R. and Prati, R. C., editors, Intelligent Systems, pages 403–417, Cham. Springer International Publishing.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q., editors, Advances in Neural Information Processing Systems, volume 27. Curran Associates, Inc.

Teodoro, D., Ferdowsi, S., Borissov, N., Kashani, E., Vicente Alvarez, D., Copara, J., Gouareb, R., Naderi, N., and Amini, P. (2021). Information Retrieval in an Infodemic: The Case of COVID-19 Publications. J Med Internet Res, 23(9):e30161.

Teodoro, D., Knafou, J., Naderi, N., Pasche, E., Gobeill, J., Arighi, C. N., and Ruch, P. (2020). UPCLASS: a Deep Learning-Based Classifier for UniProtKB Entry Publications. Database, 2020. baaa026.

Touchent, R., Romary, L., and de la Clergerie, E. (2023). CamemBERT-bio: a Tasty French Language Model Better for Your Health.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. (2023). LLaMA: Open and Efficient Foundation Language Models.

Turian, J., Ratinov, L.-A., and Bengio, Y. (2010). Word Representations: A Simple and General Method for Semi-Supervised Learning. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 384–394, Uppsala, Sweden. Association for Computational Linguistics.

Türkmen, H., Dikenelli, O., Eraslan, C., Çallı, M. C., and Özbek, S. S. (2023). BioBERTurk: Exploring Turkish Biomedical Language Model Development Strategies in Low-Resource Setting. 7(4):433–446.

Urbizu, G., San Vicente, I., Saralegi, X., Agerri, R., and Soroa, A. (2022). BasqueGLUE: A Natural Language Understanding Benchmark for Basque. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Odijk, J., and Piperidis, S., editors, Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 1603–1612, Marseille, France. European Language Resources Association.

Urbizu, G., San Vicente, I., Saralegi, X., and Corral, A. (2023). Not Enough Data to Pre-train Your Language Model? MT to the Rescue! In Rogers, A., Boyd-Graber, J., and Okazaki, N., editors, Findings of the Association for Computational Linguistics: ACL 2023, pages 3826–3836, Toronto, Canada. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is All You Need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc.

Voorhees, E., Alam, T., Bedrick, S., Demner-Fushman, D., Hersh, W. R., Lo, K., Roberts, K., Soboroff, I., and Wang, L. L. (2020). TREC-COVID: Constructing a Pandemic Information Retrieval Test Collection.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. (2018). GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Wang, L. L., Lo, K., Chandrasekhar, Y., Reas, R., Yang, J., Burdick, D., Eide, D., Funk, K., Katsis, Y., Kinney, R., Li, Y., Liu, Z., Merrill, W., Mooney, P., Murdick, D., Rishi, D., Sheehan, J., Shen, Z., Stilson, B., Wade, A., Wang, K., Wang, N. X. R., Wilhelm, C., Xie, B., Raymond, D., Weld, D. S., Etzioni, O., and Kohlmeier, S. (2020). CORD-19: The COVID-19 Open Research Dataset.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Łukasz Kaiser, Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., and Dean, J. (2016). Google's Neural Machine Translation System: Bridging the Gap Between Human and Machine Translation. CoRR, abs/1609.08144.

Yang, P., Fang, H., and Lin, J. (2017). Anserini: Enabling the Use of Lucene for Information Retrieval Research. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '17, page 1253–1256, New York, NY, USA. Association for Computing Machinery.

Yang, Y., UY, M. C. S., and Huang, A. (2020). FinBERT: A Pretrained Language Model for Financial Communications.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). XLNet: Generalized Autoregressive Pretraining for Language Understanding. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.

Zellers, R., Bisk, Y., Schwartz, R., and Choi, Y. (2018). SWAG: A Large-Scale Adversarial Dataset for Grounded Commonsense Inference.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books.

Appendix A

Example of a Raw JSON File


```
{
    "title": "The origin of the alkaline inactivation of pepsinogen.",
    "abstract": "Above pH 8.5, pepsinogen is converted into a form which
    ↪  cannot be activated to pepsin on exposure to low pH. Intermediate
    ↪  exposure to neutral pH, however, returns the protein to a form
    ↪  which can be activated. Evidence is presented for a reversible,
    ↪  small conformational change in the molecule, distinct from the
    ↪  unfolding of the protein. At the same time, the molecule is
    ↪  converted to a form of limited solubility, which is precipitated
    ↪  at low pH, where activation is normally seen. The results are
    ↪  interpreted in terms of the peculiar structure of the pepsinogen
    ↪  molecule. Titration of the basic NH2-terminal region produced an
    ↪  open form, which can return to the native form at neutral pH, but
    ↪  which is maintained at low pH by neutralization of carboxylate
    ↪  groups in the pepsin portion.",
    "journal": "Biochemistry",

    "authors": ["McPhie P"],
    "affiliations": [],
    "pubyear": "1975",
    "entrez_date": "1975-12-02",
    "pmid": "44",
    "_id": "44",
    "mesh_terms": ["D006454:Hemoglobins", "D006863:Hydrogen-Ion
    ↪  Concentration", "D007700:Kinetics", "D010434:Pepsin A",
    ↪  "D010435:Pepsinogens", "D011487:Protein Conformation",
    ↪  "D013056:Spectrophotometry, Ultraviolet"],
    "sup_mesh_terms": [],
    "chemicals": ["RN 0, D006454:Hemoglobins", "RN 0,
    ↪  D010435:Pepsinogens", "RN EC 3.4.23.1, D010434:Pepsin A"],
    "publication_types": ["Journal Article"],
    "keywords": [],
    "comments": {
        "comments_in": [],
        "comments_on": []
    },
    "pmcid": "",
    "doi": "10.1021/bi00695a003",
    "medline_ta": "Biochemistry"
}
```


Figure A.1: RAW Abstract from MBR Dataset - Example of a citation in JSON directly drawn from the MBR database.

Appendix B

Examples of Translation with repetition

Model Size: 418M

Translation Approach: By sentence

Abstract: Changements structurels et fonctionnels dans les ovaires des souris adultes traités avec diétylstilboestrol au cours de la période néonatale. Les ovaires des souris NMRI féminins âgés de 8 semaines à différentes étapes du cycle estroïde, ou des femelles néonatales traitées avec l'estrogène synthétique diethylstilboestrol (DES; 5-10(-6) microgrammes par jour pendant 5 jours), ont été étudiés histologiquement et pour la capacité de synthétiser les stéroïdes de [3H]pregnenolone in vitro. Les doses quotidiennes de 10(-4) microgrammes DES ou plus ont entraîné l'absence de corpora lutea. Dans les ovaries qui manquaient de corpora lutea, le tissu interstitial était dominé et les cellules dans ce compartiment étaient grandes avec un cytoplasme clair. Les stéroïdes synthétisés dans les homogènes ovariens ont été séparés par la chromatographie de la couche mince. L'homogénéité des stéroïdes a été vérifiée dans les expériences de recrystalisation. Les doses quotidiennes de 5-10(-4) microgrammes DES au cours de la période néonatale ont entraîné des déviations prononcées dans le modèle des stéroïdes ovariens synthétisés par rapport aux ovaries de contrôle. Dans les ovaries exposées à DES, la synthèse d'androstenedione et, surtout, de la progestérone était élevée tandis que la synthèse de 17 alpha-hydroxyprogesterone et de testostérone a été réduite par rapport aux contrôles. Ces résultats pourraient argumenter une différence dans l'activité de 17 alpha-hydroxylase et 17 beta-ol-dehydrogénase dans les ovaries des femmes traitées par DES par rapport aux contrôles. Après la transplantation des ovaires exposés à DES à des femelles ovarectomées de contrôle, le modèle stéroïde a changé à celui typique pour les ovaires de contrôle. Les ovaries de contrôle transplantées aux femmes traitées par DES avaient un modèle stéroïde similaire à celui des ovaries exposées par DES.

Figure B.1: Example of a Translation: 418M, Sentence-by-Sentence

Model Size: 418M

Translation Approach: By abstract

Abstract: Des modifications structurelles et fonctionnelles dans les ovaries de l'ovaire de contrôle des ovaries des ovaries de contrôle des ovaries.

Figure B.2: Example of a Translation: 418M, By Abstract (With Repetition)

Model Size: 1.2B

Translation Approach: By sentence

Abstract: Changements structurels et fonctionnels des ovaires chez les souris adultes traitées avec du diéthylstilboestrol pendant la période néonatale. Les ovaires de souris NMRI femelles âgées de 8 semaines à différents stades du cycle estropique, ou de femelles néonatales traitées avec l'œstrogène synthétique diéthylstilboestrol (DES; 5-10(-6) microgrammes par jour pendant 5 jours), ont été étudiés histologiquement et pour la capacité de synthétiser des stéroïdes de [3H]pregnenolone in vitro. Des doses quotidiennes de 10(-4) microgrammes de DES ou plus ont entraîné l'absence de corpora lutea. Dans les ovaires manquant de corpora lutea, le tissu interstitiel a dominé et les cellules dans ce compartiment étaient grandes avec un cytoplasme clair. Les stéroïdes synthétisés dans les homogénates ovariens ont été séparés par la chromatographie à couche mince. L'homogénéité des stéroïdes a été vérifiée dans les expériences de recrystallisation. Des doses quotidiennes de 5-10(-4) microgrammes de DES dans la période néonatale ont entraîné des écarts prononcés dans le schéma des stéroïdes ovariens synthétisés par rapport aux ovaires de contrôle. Dans les ovaires exposés au DES, la synthèse d'androstenedione et, surtout, de progestérone était élevée tandis que la synthèse de 17 alpha-hydroxyprogesterone et de testostérone était réduite par rapport aux contrôles. Ces résultats pourraient soutenir une différence d'activité de 17 alpha-hydroxylase et 17 bêta-ol-déhydrogénase dans les ovaires des femmes traitées par DES par rapport aux contrôles. Après la transplantation des ovaires exposés au DES aux femelles ovariectomées de contrôle, le modèle de stéroïde a changé à celui typique pour les ovaires de contrôle. Les ovaires de contrôle transplantés chez les femelles traitées par le DES avaient un schéma stéroïdien similaire à celui des ovaires exposés par le DES.

Figure B.3: Example of a Translation: 1.2B, Sentence-by-Sentence

Model Size: 1.2B

Translation Approach: By abstract

Abstract: Ces ovocytes ont contrôlé les modifications stéroïdiennes et fonctionnelles ovocytes contrôlés stéroïdes ovocytes de souris adultes traitées avec diethylstilboestrol dans la période néonatale. Les ovocytes de souris NMRI féminines âgées de 8 semaines ont été étudiés histologiquement et pour la capacité de synthétiser des stéroïdes de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse de la synthèse, ou de la synthèse de la synthéraxtrale. Les stéroïdes synthéraux synthéraux synthéraux synthéraux synthéraux de la synthéraux synthéraux de synthéraux ont été séparés par jour par thermés par thermologiques synthéraux synthéraïdiques synthéraïques synthéraïques synthéraïdes.

Figure B.4: Example of a Translation: 1.2B, by Abstract (With Repetition)

Appendix C

Example of Sentence & Word Tokenization

PMID: 44

Sentence 1: The origin of the alkaline inactivation of pepsinogen.
['_The', '_origin', '_of', '_the', '_alkal', 'ine', '_in', 'activ', 'ation', '_of', '_pep', 'sin', 'ogen', '.']

Sentence 2: Above pH 8.5, pepsinogen is converted into a form which cannot be activated to pepsin on exposure to low pH.
['_Ab', 'ove', '_pH', '_8.', '5,', '_pep', 'sin', 'ogen', '_is', '_convert', 'ed', '_into', '_a', '_form', '_which', '_cannot', '_be', '_activ', 'ated', '_to', '_pep', 'sin', '_on', '_expos', 'ure', '_to', '_low', '_pH', '.']

Sentence 3: Intermediate exposure to neutral pH, however, returns the protein to a form which can be activated.
['_Inter', 'medi', 'ate', '_expos', 'ure', '_to', '_neutral', '_pH', ',', '_however', ',', '_retur', 'ns', '_the', '_protein', '_to', '_a', '_form', '_which', '_can', '_be', '_activ', 'ated', '.']

Sentence 4: Evidence is presented for a reversible, small conformational change in the molecule, distinct from the unfolding of the protein.
['_Ev', 'idence', '_is', '_present', 'ed', '_for', '_a', '_re', 'vers', 'ible', ',', '_small', '_conform', 'ational', '_change', '_in', '_the', '_mol', 'ec', 'ule', ',', '_distin', 'ct', '_from', '_the', '_un', 'fold', 'ing', '_of', '_the', '_protein', '.']

Sentence 5: At the same time, the molecule is converted to a form of limited solubility, which is precipitated at low pH, where activation is normally seen.
['_At', '_the', '_same', '_time', ',', '_the', '_mol', 'ec', 'ule', '_is', '_convert', 'ed', '_to', '_a', '_form', '_of', '_limited', '_sol', 'ub', 'ility', ',', '_which', '_is', '_precip', 'itat', 'ed', '_at', '_low', '_pH', ',', '_where', '_activ', 'ation', '_is', '_norm', 'ally', '_seen', '.']

Sentence 6: The results are interpreted in terms of the peculiar structure of the pepsinogen molecule.
['_The', '_results', '_are', '_interpret', 'ed', '_in', '_terms', '_of', '_the', '_pec', 'uliar', '_structure', '_of', '_the', '_pep', 'sin', 'ogen', '_mol', 'ec', 'ule', '.']

Sentence 7: Titration of the basic NH2-terminal region produced an open form, which can return to the native form at neutral pH, but which is maintained at low pH by neutralization of carboxylate groups in the pepsin portion.
['_T', 'itr', 'ation', '_of', '_the', '_basic', '_NH', '2-', 'termin', 'al', '_region', '_produc', 'ed', '_an', '_open', '_form', ',', '_which', '_can', '_return', '_to', '_the', '_n', 'ative', '_form', '_at', '_neutral', '_pH', ',', '_but', '_which', '_is', '_mainta', 'ined', '_at', '_low', '_pH', '_by', '_neutr', 'aliz', 'ation', '_of', '_car', 'box', 'yl', 'ate', '_groups', '_in', '_the', '_pep', 'sin', '_por', 'tion', '.']

Figure C.1: Example of Sentence & Word Tokenization

Appendix D

Translation Examples

Original (PMID:33739270)
Le foie assure une grande partie du métabolisme des xénobiotiques. Ses particularités en font pourtant une cible privilégiée pour des composés toxiques. Les hépatotoxicités des xénobiotiques, ces molécules étrangères à notre organisme, constituent un vrai défi pour les cliniciens, l'industrie pharmaceutique, et les agences de santé. à la différence des hépatotoxicités intrinsèques, prévisibles et reproductibles, les hépatotoxicités idiosyncrasiques surviennent de manière non prévisible. La physiopathologie des hépatotoxicités idiosyncrasiques à médiation immune reste la moins bien connue. Le développement d'outils qui permettent désormais d'améliorer la prédiction et la compréhension de ces atteintes hépatiques paraît être une approche prometteuse pour identifier des facteurs de risque, et de nouveaux mécanismes de toxicité.

Translated (PMID:33739270)
Le foie assure une grande partie du métabolisme des xénobiotiques grâce à son équipement enzymatique considérable, à sa localisation anatomique et à sa vascularisation abondante. Cependant, ces différentes caractéristiques en font également une cible privilégiée pour les composés toxiques, en particulier dans le cas d'un métabolisme toxique. L'hépatotoxicité induite par les xénobiotiques est une cause majeure de lésions hépatiques et un véritable défi pour les cliniciens, l'industrie pharmaceutique et les agences de santé. Les hépatotoxicités intrinsèques, c'est-à-dire les hépatotoxicités prévisibles et reproductibles qui se produisent à des doses limites, sont distinguées des hépatotoxicités idiosyncratiques, qui se produisent de manière imprévisible chez les personnes présentant des sensibilités individuelles. Parmi eux, la pathophysiologie de l'hépatotoxicité immunomédiée idiosyncratique n'est toujours pas claire. Cependant, le développement d'outils visant à améliorer la prévision et la compréhension de ces troubles peut ouvrir des voies pour l'identification de facteurs de risque et de nouveaux mécanismes de toxicité.

Original (PMID:32334967)
La tuberculose est due au complexe M. tuberculosis, dont la croissance lente entraîne un long délai de rendu des tests phénotypiques utilisés pour le diagnostic bactériologique. La biologie moléculaire a réduit considérablement ce délai, notamment grâce au déploiement de la méthode Xpert® MTB/RIF (Cepheid) qui permet de détecter le complexe M. tuberculosis et la résistance à la rifampicine en 2 heures. D'autres tests détectant en plus la résistance à l'isoniazide et aux antituberculeux de seconde ligne ont été développés. Cependant, les performances de ces tests sont nettement moins bonnes si l'examen microscopique est négatif. Il est donc crucial de restreindre leur indication aux fortes suspicions cliniques. Les tests de détection de la résistance n'explorent que certaines positions caractérisées ; or, toutes les mutations responsables de l'acquisition de résistance ne sont pas connues. De plus, les performances sont variables pour les différents antituberculeux. L'avènement du séquençage génomique est une perspective prometteuse. La faisabilité en routine doit encore être évaluée et l'analyse des données reste à standardiser. L'essor des techniques de biologie moléculaire a révolutionné le diagnostic de la tuberculose et de la résistance. Cependant, elles restent des tests de dépistage dont les résultats doivent être confrontés aux méthodes phénotypiques de référence.

Translated (PMID:32334967)
La tuberculose est causée par le complexe M. tuberculosis. Sa croissance lente retarde le diagnostic bactériologique basé sur des tests phénotypiques. La biologie moléculaire a considérablement réduit ce retard, notamment grâce au déploiement du test Xpert® MTB/RIF (Cepheid), qui détecte le complexe de M. tuberculose et la résistance à la rifampicine en 2 heures. D'autres tests détectant la résistance à l'isoniazide et aux médicaments antituberculeux de deuxième ligne ont été développés. Cependant, les performances des tests moléculaires sont considérablement réduites si le dépistage de la microscopie de bacille acide rapide est négatif. Il est donc crucial de limiter leur indication à de fortes suspicions cliniques. Les tests de détection de la résistance n'explorent que certaines positions caractérisées; cependant, toutes les mutations de résistance aux médicaments ne sont pas connues. En outre, les performances varient pour différents médicaments antituberculeux. L'avènement de la séquençage génomique est prometteur. Son intégration dans le flux de travail de routine doit encore être évaluée et l'analyse des données doit encore être normalisée. La montée des techniques de biologie moléculaire a révolutionné le diagnostic de la tuberculose et de la résistance aux médicaments. Cependant, ils restent des tests de dépistage; les résultats doivent encore être confirmés par des méthodes de référence phénotypiques.

Original (PMID: 33742585)
Dans un souci d'amélioration de la qualité de vie des personnes atteintes de maladie chronique, les pratiques de soins se sont enrichies de l'éducation thérapeutique du patient (ETP). Celle-ci vise l'acquisition de savoirs et de compétences plurielles par les malades pour favoriser une gestion optimale de la pathologie au quotidien et des changements qui en découlent, en limitant les répercussions négatives sur leur autonomie et leur bien-être. Le sujet est placé au cœur de son dispositif, en position de décision et de responsabilité, et collabore activement avec les différents acteurs de soins. L'ETP implique donc la prise en compte de la dimension psychique du patient, en s'appuyant sur la psychologie et des concepts fondamentaux pour sa mise en œuvre.

Translated (PMID: 33742585)
Dans un effort pour améliorer la qualité de vie des personnes atteintes de maladies chroniques, les pratiques de soins ont été enrichis par l'éducation thérapeutique des patients (TPE). Cela vise à l'acquisition de connaissances et de compétences plurielles par les patients, ce qui favorise une gestion optimale de la maladie sur une base quotidienne et des changements qui en découlent, en limitant leurs répercussions négatives sur leur autonomie et leur bien-être. Le sujet est placé au cœur du système, dans une position de décision et de responsabilité, et collabore activement avec les différents acteurs de la santé. Le TPE implique donc la prise en compte de la dimension psychologique du patient, en utilisant la psychologie et les concepts fondamentaux pour sa mise en œuvre.

Appendix E

Hyperparameter Optimization Range

```
per_device_train_batch_size: tune.qrandint(8, 16, 8) # batch size from 8
↪  to 16 with stride of 8
metrics: "f1" # Main metric used for optimization
direction: ["max","maximize"] # Direction of the optimization (minimize
↪  or maximize)
learning_rate: tune.loguniform(1e-6, 1e-4) # Learning rate
num_train_epochs: tune.qrandint(10, 20, 10) # Number of epochs from 10 to
↪  20 with stride of 10
weight_decay: tune.uniform(0, .5) # Weight decay (Adam Optimizer)
warmup_ratio: tune.uniform(0, .5) # Warmup ratio (Adam Optimizer)
dropout: tune.uniform(0, .5) # Dropout
reduction_factor: 2 # Reduction factor, ratio of the trials to terminate
↪  early
grace_period: 2 # Grace period, number of epoch at least a trial is
↪  trained
max_t: 100 # Trials will be stopped after max_t training iterations
n_trials: 10 # Number of trials conducted during the hyperparameter
↪  tuning process
```

Figure E.1: Hyperparameter Optimization Range: CAS

```
per_device_train_batch_size: 16
metrics: "rmse"
direction: ["min","minimize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 50, 10)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 4
max_t: 100
n_trials: 20
```

Figure E.2: Hyperparameter Optimization Range: CLISTER

```
per_device_train_batch_size: 16
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 50, 10)
gradient_accumulation_steps: tune.randint(1, 5) # Gradient accumulation
↪  which multiply the batch size from 1 to 5
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 4
max_t: 100
n_trials: 20
```

Figure E.3: Hyperparameter Optimization Range: DiaMed

```
per_device_train_batch_size: tune.qrandint(8, 16, 8)
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 50, 5)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 2
max_t: 100
n_trials: 10
```

Figure E.4: Hyperparameter Optimization Range: E3C

```
per_device_train_batch_size: 16
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 30, 10)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 2
max_t: 100
n_trials: 10
```

Figure E.5: Hyperparameter Optimization Range: ESSAI

```
per_device_train_batch_size: 16
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 40, 10)
gradient_accumulation_steps: tune.randint(1, 5)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 3
max_t: 100
n_trials: 10
```

Figure E.6: Hyperparameter Optimization Range: FrenchMedMCQA

```
per_device_train_batch_size: tune.qrandint(8, 16, 4)
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 50, 10)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 4
max_t: 100
n_trials: 40
```

Figure E.7: Hyperparameter Optimization Range: Mantra-GSC

```
per_device_train_batch_size: 16
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 40, 10)
gradient_accumulation_steps: tune.randint(1, 5)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 3
max_t: 100
n_trials: 10
```

Figure E.8: Hyperparameter Optimization Range: MorFITT

```
per_device_train_batch_size: tune.qrandint(8, 16, 8)
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 30, 10)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 4
max_t: 100
n_trials: 20
```

Figure E.9: Hyperparameter Optimization Range: PxCorpus

```
per_device_train_batch_size: 16
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 30, 10)
gradient_accumulation_steps: tune.randint(1, 5)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 4
max_t: 100
n_trials: 20
```

Figure E.10: Hyperparameter Optimization Range: PxCorpus

```
per_device_train_batch_size: tune.qrandint(8, 16, 8)
metrics: "f1"
direction: ["max","maximize"]
learning_rate: tune.loguniform(1e-6, 1e-4)
num_train_epochs: tune.qrandint(10, 40, 10)
weight_decay: tune.uniform(0, .5)
warmup_ratio: tune.uniform(0, .5)
dropout: tune.uniform(0, .5)
reduction_factor: 2
grace_period: 4
max_t: 100
n_trials: 20
```

Figure E.11: Hyperparameter Optimization Range: QUAERO

Appendix F

Fine-Tuning: Dataset Statistics

POS Tags	Count	Percentage (%)	Cumulative Percentage (%)
Noun	20,052	23.1	23.1
Personal Pronoun	11,049	12.73	35.83
Adjective	9,179	10.57	46.4
Article	9,085	10.47	56.87
Punctuation	7,500	8.64	65.51
Number	4,298	4.95	70.46
Sentence	3,883	4.47	74.93
Past Participle Verb	3,114	3.59	78.52
Conjunction	2,655	3.06	81.58
Present Tense Verb	2,485	2.86	84.44
Adverb	2,468	2.84	87.29
Possessive Determiner	2,233	2.57	89.86
Imperfect Tense Verb	2,117	2.44	92.3
Personal Pronoun	1,583	1.82	94.12
Proper Noun	1,446	1.67	95.79
Infinitive Verb	567	0.65	96.44
Present Participle Verb	512	0.59	97.03
Abbreviation	471	0.54	97.57
Possessive Determiner	428	0.49	98.06
Demonstrative Pronoun	397	0.46	98.52
Relative Pronoun	320	0.37	98.89
Indefinite Pronoun	263	0.3	99.19
Quotation Punctuation	232	0.27	99.46
Symbol	210	0.24	99.7
Simple Past Verb	130	0.15	99.85
Future Tense Verb	46	0.05	99.91
Conditional Verb	26	0.03	99.94
Subjunctive Present Verb	22	0.03	99.96
Interjection	18	0.02	99.98
Subjunctive Imperfect Verb	16	0.02	100.0
Total	86,805	100	100

Table F.1: CAS POS Tags Distribution - CAS is a POS Tagging task containing a total of 86,805 instances distributed across 30 POS tags. For easy navigation, dataset is presented in Section 3.4.3.3.1 and Table 4.14 shows the tasks results.

Classes	**Count**	(%)	**Cum. (%)**
Neoplasms	242	33.33	33.33
Certain infectious and parasitic diseases	89	12.26	45.59
Injury, poisoning and certain other consequences of external causes	74	10.19	55.79
Congenital malformations, deformations and chromosomal abnormalities	55	7.58	63.36
Diseases of the musculoskeletal system and connective tissue	52	7.16	70.52
Diseases of the circulatory system	43	5.92	76.45
Diseases of the digestive system	34	4.68	81.13
Endocrine, nutritional and metabolic diseases	24	3.31	84.44
Pregnancy, childbirth and the puerperium	23	3.17	87.6
Diseases of the eye and adnexa	21	2.89	90.5
Diseases of the genitourinary system	20	2.75	93.25
Diseases of the skin and subcutaneous tissue	19	2.62	95.87
Diseases of the nervous system	13	1.79	97.66
Diseases of the respiratory system	10	1.38	99.04
Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism	7	0.96	100.0
Total	726	100	100

Table F.2: DiaMed Classes Distribution - DiaMed is a multi-class classification task containing a total of 726 instances distributed across 15 classes. For easy navigation, dataset is presented in Section 3.4.3.1.2 and Table 4.4 shows the tasks results.

Named Entity	Count	Percentage (%)	Cumulative Percentage (%)
Clinical Entity	3,270	100.0	100.0
Total	3,270	100	100

Table F.3: E3C/Clinical Named Entity Distribution - E3C/Clinical is a NER task containing a total of 3,270 instances with a single named entity. For easy navigation, dataset is presented in Section 3.4.3.2.1 and Table 4.8 shows the tasks results.

Named Entities	Count	Percentage (%)	Cumulative Percentage (%)
Event	3,836	66.64	66.64
Body Part	654	11.36	78.01
Lab Result	507	8.81	86.81
Actor	426	7.4	94.21
Time Expression	333	5.79	100.0
Total	5,756	100	100

Table F.4: E3C/Temporal Named Entities Distribution - E3C/Temporal is a NER task containing a total of 5,756 instances distributed across 5 named entities. For easy navigation, dataset is presented in Section 3.4.3.2.1 and Table 4.9 shows the tasks results.

POS Tags	Count	Percentage (%)	Cumulative Percentage (%)
Noun	39,279	26.14	26.14
Personal Pronoun	22,261	14.81	40.95
Article	18,404	12.25	53.2
Adjective	11,056	7.36	60.56
Punctuation	9,272	6.17	66.73
Sentence	6,016	4.0	70.73
Conjunction	5,653	3.76	74.49
Number	5,530	3.68	78.17
Possessive Determiner	5,480	3.65	81.82
Past Participle Verb	4,821	3.21	85.03
Present Tense Verb	3,556	2.37	87.4
Adverb	3,490	2.32	89.72
Proper Noun	2,622	1.74	91.46
Future Tense Verb	2,562	1.7	93.17
Infinitive Verb	2,442	1.63	94.79
Demonstrative Pronoun	1,796	1.2	95.99
Present Participle Verb	1,661	1.11	97.09
Indefinite Pronoun	1,210	0.81	97.9
Personal Pronoun	1,089	0.72	98.62
Relative Pronoun	672	0.45	99.07
Abbreviation	325	0.22	99.29
Possessive Determiner	312	0.21	99.49
Quotation Punctuation	212	0.14	99.64
Singular or Mass Noun	161	0.11	99.74
Symbol	156	0.1	99.85
Conditional Verb	90	0.06	99.91
Subjunctive Present Verb	53	0.04	99.94
Simple Past Verb	46	0.03	99.97
Imperfect Tense Verb	42	0.03	100.0
Total	150,269	100	100

Table F.5: ESSAI POS Tags Distribution - ESSAI is a POS Tagging task containing a total of 150,269 instances distributed across 29 POS tags. For easy navigation, dataset is presented in Section 3.4.3.3.2 and Table 4.15 shows the tasks results.

Named Entities	Count	Percentage (%)	Cumulative Percentage (%)
Disorders	288	32.76	32.76
Chemical/Drugs	236	26.85	59.61
Procedures	129	14.68	74.29
Living Beings	91	10.35	84.64
Anatomy	66	7.51	92.15
Physiology	44	5.01	97.16
Objects	25	2.84	100.0
Total	879	100	100

Table F.6: MantraGSC/Merged Named Entities Distribution - MantraGSC/Merged is a NER task containing a total of 879 instances distributed across 7 named entities. For easy navigation, dataset is presented in Section 3.4.3.2.2 and Table 4.10 shows the tasks results.

Labels	**Count**	**Percentage (%)**	**Cumulative Percentage (%)**
Veterinary	824	16.11	16.11
Etiology	741	14.49	30.6
Psychology	608	11.89	42.48
Surgery	549	10.73	53.22
Genetics	505	9.87	63.09
Physiology	490	9.58	72.67
Pharmacology	299	5.85	78.51
Microbiology	273	5.34	83.85
Immunology	262	5.12	88.97
Chemistry	212	4.14	93.12
Virology	200	3.91	97.03
Parasitology	152	2.97	100.0
Total	5,115	100	100

Table F.7: MorFITT Labels Distribution - MorFITT is a multi-label classification task containing a total of 5,115 instances distributed across 12 labels. For easy navigation, dataset is presented in Section 3.4.3.1.4 and Table 4.6 shows the tasks results.

NE ID	Deduced Entity	Count	Percentage (%)	Cumulative Percentage (%)
dos_val	Dosage Value	1,600	13.96	13.96
dos_uf	Dosage Unit Factor	1,513	13.2	27.15
rhythm_tdte	Rhythm To Date	1,320	11.51	38.67
dur_val	Duration Value	1,208	10.54	49.2
dur_ut	Duration Unit	1,205	10.51	59.71
drug	Drug Name	935	8.16	67.87
d_dos_val	Daily Dosage Value	849	7.41	75.27
d_dos_up	Daily Dosage Upper Limit	822	7.17	82.44
inn	INN	380	3.31	85.76
cma_event	CMA Event	313	2.73	88.49
d_dos_form	Daily Dosage Form	280	2.44	90.93
rhythm_perday	Rhythm Per Day	241	2.1	93.03
dos_cond	Dosage Condition	134	1.17	94.2
rhythm_hour	Rhythm Hour	112	0.98	95.18
freq_ut	Frequency Unit	109	0.95	96.13
d_dos_form_ext	Daily Dosage Form Extended	66	0.58	96.7
A	A	52	0.45	97.16
roa	ROA	46	0.4	97.56
freq_int_v1	Frequency Interval V1	31	0.27	97.83
qsp_val	Quantity Sufficient Value	29	0.25	98.08
rhythm_rec_ut	Rhythm Record Unit	29	0.25	98.33
max_unit_val	Maximum Unit Value	28	0.24	98.58
qsp_ut	Quantity Sufficient Unit	28	0.24	98.82
freq_int_v1_ut	Frequency Interval V1 Unit	26	0.23	99.05
rhythm_rec_val	Rhythm Record Value	24	0.21	99.26
freq_int_v2	Frequency Interval V2	20	0.17	99.43
freq_val	Frequency Value	19	0.17	99.6
fasting	Fasting	18	0.16	99.76
max_unit_uf	Maximum Unit Factor	18	0.16	99.91
freq_int_v2_ut	Frequency Interval V2 Unit	10	0.09	100.0
Total		11,465	100	100

Table F.8: PxCorpus/Task 1 Named Entities Distribution - PxCorpus/Task 1 is a NER task containing a total of 11,465 instances distributed across 30 named entities. For easy navigation, dataset is presented in Section 3.4.3.2.3 and Table 4.11 shows the tasks results.

Classes	Count	Percentage (%)	Cumulative Percentage (%)
Medical prescription	1,574	91.14	91.14
None	115	6.66	97.8
Negate	21	1.22	99.02
Replace	17	0.98	100.0
Total	1,727	100	100

Table F.9: PxCorpus/Task 2 Classes Distribution - PxCorpus/Task 2 is a multi-class classification task containing a total of 1,727 instances distributed across 4 classes. For easy navigation, dataset is presented in Section 3.4.3.1.5 and Table 4.7 shows the tasks results.

Named Entities	Count	Percentage (%)	Cumulative Percentage (%)
Chemical/Drugs	2,167	36.11	36.11
Disorders	1,286	21.43	57.54
Procedures	835	13.91	71.45
Living Beings	722	12.03	83.49
Physiology	300	5.0	88.49
Anatomy	265	4.42	92.9
Objects	162	2.7	95.6
Devices	144	2.4	98.0
Geo. Areas	64	1.07	99.07
Phenomena	56	0.93	100.0
Total	6,001	100	100

Table F.10: QUAERO/EMEA Named Entities Distribution - QUAERO/EMEA is a NER task containing a total of 6,001 instances distributed across 10 named entities. For easy navigation, dataset is presented in Section 3.4.3.2.4 and Table 4.12 shows the tasks results.

Named Entities	Count	Percentage (%)	Cumulative Percentage (%)
Disorders	2,115	31.26	31.26
Procedures	1,528	22.59	53.85
Chemical/Drugs	819	12.11	65.96
Living Beings	777	11.49	77.44
Anatomy	744	11.0	88.44
Physiology	353	5.22	93.66
Geo. Areas	126	1.86	95.52
Phenomena	123	1.82	97.34
Devices	97	1.43	98.77
Objects	83	1.23	100.0
Total	6,765	100	100

Table F.11: QUAERO/Medline Named Entities Distribution - QUAERO (*Medline*) is a NER task containing a total of 6,765 instances distributed across 10 named entities. For easy navigation, dataset is presented in Section 3.4.3.2.4 and Table 4.13 shows the tasks results.

Appendix G

Task Data Samples

List of words: ['la', 'diarrhée', 'était', 'associée', 'à', 'des', 'douleurs', 'hypogastriques', '.']

POS Tags: ['B-DET:ART', 'B-NOM', 'B-VER:impf', 'B-VER:pper', 'B-PRP', 'B-PRP:det', 'B-NOM', 'B-ADJ', 'B-SENT']

Figure G.1: Data Sample - CAS - Example of a sequence of words with POS Tags. The CAS task focuses on detecting grammatical features in clinical cases. For easy navigation, dataset is presented in Section 3.4.3.3.1 and Table F.1 shows the dataset statistics.

Similarity Score (average): 3

Sentence 1: L'UIV a objectivé un retard de sécrétion avec importante dilatation pyélo-calicielle et de l'uretère lombaire en amont d'un énorme calcul de l'uretère iliaque et pelvien droit (Figure 2).

Sentence 2: L'UIV a montré une importante dilatation urétéro-pyélo-calicielle en amont d'un énorme calcul de l'uretère gauche, le coté droit était sans anomalies (Figure 6).

Figure G.2: Data Sample - CLISTER - Example of a pair of sentences a global similarity score computed using the mean of the multiple curators scores. For easy navigation, dataset is presented in Section 3.4.3.4.1.

Scores: [4.0, 4.5, 2.0, 4.0, 4.0]

Similarity Score (average): 3.7

Sentence 1: - En l'absence d'amélioration comme en cas de persistance des symptômes, prendre un avis médical.

Sentence 2: En l'absence d'amélioration comme en cas de persistance des symptômes au-delà de 7 jours de traitement, prenez un avis médical.

Figure G.3: Data Sample - DEFT-2020/Task 1 - Example of a pair of sentences with score per curator and a global similarity score computed using the mean of the scores. For easy navigation, dataset is presented in Section 3.4.3.4.2.

Source: une amnésie antérograde ainsi que des altérations des fonctions psychomotrices sont susceptibles d' apparaître dans les heures qui suivent la prise

Target 1: des troubles de mémoire ainsi que des altérations des fonctions psychomotrices sont susceptibles d' apparaître dans les heures qui suivent la prise du médicament

Target 2: ce médicament se présente sous forme de comprimé

Target 3: celui-ci se caractérise par l' apparition , en quelques heures ou en quelques jours , de signes tels que anxiété importante , insomnie , douleurs musculaires , mais on peut observer également une agitation , une irritabilité , des maux de tête , un engourdissement ou des picotements des extrémités , une sensibilité anormale au bruit , à la lumière ou aux contacts physiques , etc. les modalités de l' arrêt du traitement doivent être définies avec votre médecin

Correct target: Target 1

Figure G.4: Data Sample - DEFT-2020/Task 2 - Example of a source sentence related to three target sentences. For easy navigation, dataset is presented in Section 3.4.3.1.1.

Source: 7 et 8 juillet : sommet du G20 à Hambourg (Allemagne).

Target 1: Abduction

Target 2: Les sept péchés capitaux.

Target 3: sommet du G20 à Hambourg, en Allemagne.

Correct target: Target 3

Figure G.5: DEFT-2020/Task 2: Illustration of Misclassification in Non-Life Science Instance

Title: Des Furoncles résistants aux antibiotiques: penser à la myiase!!

Clinical Case: Il s'agit d'un militaire tunisien, âgé de 37 ans, de sexe masculin, en détachement onusien au contingent tunisien en République Démocratique du Congo durant l'année 2008. Dans ses antécédents, il y a un diabète non insulinodépendant bien équilibré sous antidiabétiques oraux depuis cinq ans et une obésité (Indice de Masse Corporelle =37 kg/m2) associée à une dyslipidémie. En octobre 2008, ce militaire était déployé pendant 10 semaines dans une zone périphérique de Kinshasa. Un mois après, il s´est présenté à la consultation, souffrant de deux papules prurigineuses isolées apparues depuis 6 jours au niveau du thorax et de l'épaule. L'examen clinique a trouvé deux lésions furonculoïdes, de 5 mm de diamètre chacune, avec un petit orifice ne laissant pas sourdre du pus, entouré d'un liseré érythémateux légèrement tuméfié, sans chaleur locale (Figure 1). Le reste de l'examen était sans particularité notamment pas de fièvre et le bilan biologique standard n´apportait pas d´élément d´orientation. Devant le terrain de débilité (diabète type II), un traitement à base d'oxacycline à la dose de 2 g/jour pendant 7 jours per os avec des soins locaux a été instauré mais s'est avéré inefficace. Le retard de guérison fut d'abord rattaché à son diabète et les lésions ont été traitées par une pommade antibiotique à base de cyclines (pommade hydrophobe). Après deux jours de traitement par la pommade grasse, deux larves blanchâtres de dimensions 10x5mm ont sailli de chaque lésion à la suite d'une compression bidigitale. Ces larves ont été retirées facilement à l'aide d'une pince, laissant derrière elles un orifice propre sans pus. L'aspect furonculoïde de la lésion cutanée, la chronologie des événements, la notion de séjour en république Démocratique du Congo, confrontés à la morphologie de la larve ont permis de conclure au diagnostic de myiase furonculoïde et les larves étaient identifiées comme appartenant à Cordylobia anthropophaga.

ICD-10: L00-L99 Diseases of the skin and subcutaneous tissue

Figure G.6: Data Sample - Diamed - In this clinical case classification example, two insights are noted: (1) the title is highly informative and succinct, (2) the clinical case description is extensive, containing 303 words, which suggests that the length of the sequence could approach the model's input limit when tokenized into subwords. For easy navigation, dataset is presented in Section 3.4.3.1.2 and Table F.2 shows the dataset statistics.

List of words: ['L', "'examen", 'retrouvait', 'par', 'ailleurs', 'une', 'tuméfaction', 'médio', 'claviculaire', 'gauche', 'suggestive', 'de', 'cal', 'osseux', 'sur', 'une', 'fracture', 'antérieure', ';', 'des', 'bourrelets', 'épiphysaires', 'aux', 'poignets', ',', 'un', 'chapelet', 'costal', '.']

NER Tags: ['O', 'O', 'O', 'O', 'O', 'O', 'B-CLINENTITY', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'B-CLINENTITY', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'B-CLINENTITY', 'I-CLINENTITY', 'O']

Figure G.7: Data Sample - E3C/Clinical Illustration of NER tagging, where words in color denote CLINENTITY for improved clarity. For easy navigation, dataset is presented in Section 3.4.3.2.1 and Table F.3 shows the dataset statistics.

List of words: ['La', 'patiente', 'a', 'eu', 'sept', 'accouchements', 'par', 'voie', 'basse', 'sans', 'complications', 'notables', 'et', 'une', 'ligature', 'des', 'trompes', 'il', 'ya', '35', 'ans', '.']

NER Tags: ['B-ACTOR', 'I-ACTOR', 'O', 'O', 'B-TIMEX3', 'B-EVENT', 'O', 'O', 'O', 'O', 'B-EVENT', 'O', 'O', 'O', 'B-EVENT', 'B-BODYPART', 'I-BODYPART', 'O', 'O', 'B-TIMEX3', 'I-TIMEX3', 'O']

Figure G.8: Data Sample - E3C/Temporal Illustration of NER tagging, where words in color denote a Named Entity for improved clarity. For easy navigation, dataset is presented in Section 3.4.3.2.1 and Table F.4 shows the dataset statistics.

List of words: ['Cancer', 'du', 'rein', 'métastatique', ',', 'à', 'cellules', 'claires', '.']

POS Tags: ['NOM', 'PRP:det', 'NOM', 'ADJ', 'PUN', 'PRP', 'NOM', 'ADJ', 'SENT']

Figure G.9: Data Sample - ESSAI Illustration of POS tagging in a sequence extracted from a clinical trial. For easy navigation, dataset is presented in Section 3.4.3.3.2 and Table F.5 shows the dataset statistics.

Question: Parmi les affirmations suivantes, une seule est fausse, indiquer laquelle: les particules alpha

Answer a: Sont formées de noyaux d'hélium

Answer b: Sont peu pénétrantes

Answer c: Sont arrêtées par une feuille de papier

Answer d: Sont arrêtées par une feuille de papier

Answer e: Sont peu ionisantes

Correct answers: [e]

Number of correct answers: 1

Figure G.10: Data Sample - FrenchMedMCQA - Illustration of a question with five answer options, where only one is correct in this instance. For easy navigation, dataset is presented in Section 3.4.3.1.3 and Figure 3.8 shows the dataset statistics.

List of words (EMEA): ['En', 'cas', 'de', 'surdosage', 'possible,', 'contactez', 'immédiatement', 'votre', 'médecin.']

NER Tags (EMEA): ['O', 'O', 'O', 'B-DISO', 'O', 'O', 'O', 'O', 'B-LIVB']

Figure G.11: Data Sample - MantraGSC/EMEA Illustration of NER tagging, one example per sub-dataset, where words in color denote a Named Entity for improved clarity. For easy navigation, dataset is presented in Section 3.4.3.2.2 and Table F.6 shows the dataset statistics.

List of words (Medline): ["L'obésité.", 'Quelques', 'remarques', 'sur', "l'approche", 'psychosomatique.']

NER Tags (Medline): ['B-DISO', 'O', 'O', 'O', 'O', 'O']

Figure G.12: Data Sample - MantraGSC/Medline Illustration of NER tagging, one example per sub-dataset, where words in color denote a Named Entity for improved clarity. For easy navigation, dataset is presented in Section 3.4.3.2.2 and Table F.6 shows the dataset statistics.

List of words (Patents): [”L’utilisation”, ’selon’, ’la’, ’revendication’, ’3’, ’,’, ’où’, ’le’, ’laxatif’, ’osmotique’, ’est’, ’du’, ’glycol’, ’de’, ’polyéthylène’, ’3350’, ’.’]

NER Tags (Patents): [’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’B-CHEM’, ’I-CHEM’, ’O’, ’O’, ’B-CHEM’, ’I-CHEM’, ’I-CHEM’, ’I-CHEM’, ’O’]

Figure G.13: Data Sample - MantraGSC/Patents Illustration of NER tagging, one example per sub-dataset, where words in color denote a Named Entity for improved clarity. For easy navigation, dataset is presented in Section 3.4.3.2.2 and Table F.6 shows the dataset statistics.

Abstract: Prévalence et nouveaux génotypes d'Enterocytozoon bieneusi chez les chiens et chats de refuges dans la province du Sichuan, dans le sud-ouest de la Chine. Enterocytozoon bieneusi est un parasite intracellulaire commun qui infecte un large éventail d'hôtes, y compris les humains et les animaux de compagnie, ce qui soulève des problèmes de transmission zoonotique. Cependant, il existe peu d'informations épidémiologiques sur la prévalence et les génotypes d'E. bieneusi chez les chiens et les chats des refuges dans la province du Sichuan, au sud-ouest de la Chine. Au total, 880 échantillons de matières fécales ont été prélevés dans des refuges dans différentes villes de la province du Sichuan, dont 724 échantillons de chiens et 156 échantillons de chats. Enterocytozoon bieneusi a été déterminé par analyse de séquence de l'espaceur transcrit interne ribosomique (ITS). Dans l'ensemble, la prévalence d'E. bieneusi était de 18 % (158/880) et le parasite a été détecté chez 18,8 % (136/724) et 14,1 % (22/156) des chiens et des chats examinés, respectivement. L'analyse des séquences a révélé la présence de cinq génotypes chez le chien, dont trois génotypes connus CD9 (n = 92), PtEb IX (n = 41) et type IV (n = 1), et deux nouveaux génotypes SCD-1 (n = 1) et SCD-2 (n = 1). De même, quatre génotypes ont été identifiés chez les chats, dont CD9 (n = 11), Type IV (n = 6), D (n = 4) et PtEb IX (n = 1). Les génotypes D et de type IV ont été précédemment identifiés chez l'homme et sont rapportés chez des chiens et des chats des refuges dans la présente étude, ce qui indique que ces animaux pourraient être des sources potentielles d'infections par microsporidiose chez les humains.

Speciality: [Genetics, Veterinary]

Figure G.14: Data Sample - MorFITT - Example of an abstract annotated with two labels. For easy navigation, dataset is presented in Section 3.4.3.1.4 and Table F.7 shows the dataset statistics.

List of words: ['antacapone', '200', 'milligrammes', '2', 'comprimés', 'le', 'matin', '1', 'à', 'midi', 'et', '2', 'le', 'soir', 'traitement', 'pour', '4', 'semaines']

NER Tags: ['B-drug', 'B-d_dos_val', 'B-d_dos_up', 'B-dos_val', 'B-dos_uf', 'O', 'B-rhythm_tdte', 'B-dos_val', 'O', 'B-rhythm_hour', 'O', 'B-dos_val', 'O', 'B-rhythm_tdte', 'O', 'O', 'B-dur_val', 'B-dur_ut']

Labels: Medical Prescription

Figure G.15: Data Sample - PxCorpus/Task 1 & 2: Named Entity Recognition & Classification Illustration of NER tagging, where words in color denote named entities for improved clarity. For easy navigation, dataset is presented in Section 3.4.3.2.3, Table F.8 and Table F.9 shows the datasets statistics.

List of words: ['Prialt', 'ne', 'doit', 'pas', 'être', 'utilisé', 'chez', 'l', "'", 'enfant', '.']

NER Tags: ['B-CHEM', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'B-LIVB', 'O']

Figure G.16: Data Sample - QUAERO/EMEA) Illustration of NER tagging, where words in color denote names entities for improved clarity. For easy navigation, dataset is presented in Section 3.4.3.2.4 and Table F.10 shows the dataset statistics.

List of words: ['Traitement', 'des', 'métastases', 'hépatiques', 'des', 'cancers', 'colorectaux', ':', "jusqu'", 'où', 'aller', '?']

NER Tags: ['B-PROC', 'O', 'B-DISO', 'I-DISO', 'O', 'B-DISO', 'I-DISO', 'O', 'O', 'O', 'O', 'O']

Figure G.17: Data Sample - QUAERO/Medline Illustration of NER tagging, where words in color denote names entities for improved clarity. For easy navigation, dataset is presented in Section 3.4.3.2.4 and Table F.11 shows the dataset statistics.

Appendix H

TransBERT Vs cTransBERT: All results by datasets

Labels	TransBERT			cTransBERT			Support
	P	***R***	***F_1***	***P***	***R***	***F_1***	
Veterinary	81.02	**90.54**	**85.49**	**82.45**	88.25	85.24	824
Etiology	**69.38**	68.75	**68.90**	67.84	**69.12**	68.24	741
Psychology	**85.60**	**87.67**	**86.58**	85.28	87.51	86.33	608
Surgery	**81.58**	86.84	84.04	80.74	**87.81**	**84.10**	549
Genetics	**75.41**	**78.91**	**77.04**	75.11	77.61	76.22	505
Physiology	**68.57**	**54.10**	**60.36**	67.11	53.68	59.56	490
Pharmacology	70.18	**69.41**	**69.45**	**71.33**	68.78	69.41	299
Microbiology	71.36	**76.24**	73.53	**74.14**	75.27	**74.41**	273
Immunology	**68.09**	**67.43**	**67.21**	67.72	67.32	67.16	262
Chemistry	**69.80**	**54.88**	**60.57**	68.48	49.11	55.73	212
Virology	69.95	**73.59**	**71.25**	**72.16**	69.84	70.40	200
Parasitology	69.21	**75.71**	**72.27**	**69.34**	67.06	67.95	152
Weighted avg	**75.32**	**76.12**	**75.36**	75.31	75.03	74.76	5,115
Macro avg	73.35	**73.67**	**73.06**	**73.48**	71.78	72.06	5,115
Micro avg	**75.38**	**76.12**	**75.74**	75.34	75.03	75.15	5,115

Table H.1: Detailed Model Evaluation for MorFITT (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 12 labels. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Classes	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
1 Correct Answer	93.19	**90.46**	91.79	**96.32**	88.68	**92.33**	1,079
2 Correct Answers	**36.38**	**37.43**	**35.90**	33.54	20.80	23.90	670
3 Correct Answers	**47.65**	63.19	53.74	44.51	**77.32**	**56.17**	929
4 Correct Answers	**21.49**	**9.25**	**12.64**	9.65	5.29	6.75	381
5 Correct Answers	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	43
Weighted avg	**57.24**	**59.38**	**57.25**	55.26	59.12	54.86	3,102
Macro avg	**39.74**	**40.07**	**38.82**	36.81	38.42	35.83	3,102
Micro avg (Accuracy)	←	**59.38**	→	←	59.12	→	3,102

Table H.2: Detailed Model Evaluation for FrenchMedMCQA (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 5 classes. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Classes	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Micro avg (Accuracy)	←	98.82	→	←	**99.27**	→	1,100

Table H.3: Detailed Model Evaluation for DEFT-2020/Task 2 (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 3 classes. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Classes	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Medical Prescription	97.02	**99.24**	**98.11**	**97.31**	98.79	98.04	1,574
None	**80.77**	68.59	**73.84**	75.94	**72.10**	73.63	115
Negate	**66.67**	**41.67**	**50.48**	39.76	**41.67**	38.91	21
Replace	**36.67**	**18.67**	**24.43**	0.00	0.00	0.00	17
Weighted avg	**95.28***	**95.77**	**95.34***	94.34*	95.37	94.79*	1,727
Macro avg	**70.28***	**57.04**	**61.71***	53.25*	53.14	52.64*	1,727
Micro avg (Accuracy)	←	**95.77**	→	←	95.37	→	1,727

Table H.4: Detailed Model Evaluation for PxCorpus/Task 2 (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 4 classes. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Classes	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Neoplasms	89.34	**91.09**	**90.10**	**91.20**	87.88	89.42	242
Infectious	78.34	**85.44**	81.34	**81.90**	84.67	**82.94**	89
Injury	78.40	**83.86**	**80.32**	**78.81**	81.03	79.39	74
Cong. Malform.	**70.92**	61.89	**63.57**	64.60	**63.17**	63.29	55
Musculoskeletal	**67.74**	68.68	**65.86**	62.39	**70.50**	64.77	52
Circulatory	**73.81**	**58.27**	**62.16**	72.19	54.69	57.76	43
Digestive	58.11	61.94	59.17	**61.56**	**66.94**	**63.72**	34
Endocrine	65.95	**72.67**	**66.39**	**68.94**	72.00	61.90	24
Pregnancy	**78.67**	**81.00**	**77.11**	69.33	60.33	60.69	23
Eye	**58.33**	**59.67**	**57.67**	54.00	53.67	53.43	21
Genitourinary	**79.33**	70.57	**70.78**	58.17	**76.29**	63.87	20
Skin	**66.00**	51.33	**55.97**	53.00	**60.33**	51.69	19
Nervous	**10.00**	**20.00**	**13.33**	6.67	13.33	8.89	13
Respiratory	**20.00**	6.67	10.00	10.67	**30.00**	**15.71**	10
Blood Disorders	16.67	**30.00**	21.33	**30.00**	20.00	**23.33**	7
Weighted avg	**77.39**	**75.88**	**75.31**	75.99	74.23	73.91	726
Macro avg	**60.77**	**60.21**	**58.34**	57.56	59.66	56.05	726
Micro avg (Accuracy)	←	**75.88**	→	←	74.23	→	726

Table H.5: Detailed Model Evaluation for DiaMed (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 15 classes. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Named Entities	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Clinical Entity	**76.80**	76.89	**76.83**	74.50	**76.94**	75.70	3,270

Table H.6: Detailed Model Evaluation for E3C/Clinical (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the only named entity. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Named Entities	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Event	**86.83**	**89.04**	**87.91**	86.23	88.86	87.52	3,836
Body Part	**75.87**	**76.66**	**76.14**	71.11	73.08	72.01	654
Lab Result	**79.03**	82.76	**80.83**	78.27	**83.19**	80.62	507
Actor	**88.94**	**91.45**	**90.13**	88.00	90.96	89.44	426
Time Expression	79.00	**84.07**	81.43	**80.08**	83.49	**81.65**	333
Weighted avg	**84.63**	**86.94**	**85.74***	83.60	86.39	84.95*	5,756
Macro avg	**81.93**	**84.80**	**83.29***	80.74	83.92	82.25*	5,756
Micro avg	**84.57**	**86.94**	**85.73**	83.55	86.39	84.95	5,756

Table H.7: Detailed Model Evaluation for E3C/Temporal (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 5 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Named Entities	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Disorders	**66.74**	**65.01**	**65.73**	65.28	60.46	62.68	288
Chemical/Drugs	64.36	69.78	66.52	**68.12**	**71.55**	**69.76**	243
Procedures	**54.84**	53.29	53.44	54.55	**60.92**	**56.75**	129
Living Beings	**70.93**	71.27	**70.71**	66.80	**72.06**	68.42	91
Anatomy	**69.57**	**61.90**	**65.18**	49.12	53.52	50.80	66
Physiology	**29.27**	24.44	**26.22**	23.49	**30.44**	25.92	44
Objects	**42.00**	**58.15**	**38.42**	36.33	35.33	30.10	26
Weighted avg	**63.87**	**62.76**	**62.56**	61.41	61.71	60.93	887
Macro avg	**56.82**	**57.69**	**55.17**	51.96	54.90	52.06	887
Micro avg	**63.82**	**62.76**	**63.24**	60.45	61.71	61.03	887

Table H.8: Detailed Model Evaluation for MantraGSC/Merged (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 7 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Named Entities	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Chemical/Drugs	91.58	92.61	92.08	**91.84**	**93.08**	**92.44**	2,167
Disorders	81.02	82.88	81.93	**83.33**	**83.49**	**83.41**	1,286
Procedures	**84.64**	82.65	**83.61**	83.84	**83.39**	83.57	835
Living Beings	91.92	**93.51**	92.70	**92.82**	92.67	**92.74**	722
Physiology	**67.19**	67.79	**67.41**	66.23	**68.44**	67.09	300
Anatomy	**76.36**	72.18	73.79	76.02	**73.91**	**74.88**	265
Objects	**69.94**	70.19	69.83	67.93	**75.11**	**71.07**	162
Devices	**86.99**	**83.04**	**84.91**	85.59	79.81	82.48	144
Geo. Areas	88.52	**87.67**	**87.95**	**89.70**	86.45	87.91	64
Phenomena	**70.68**	54.63	**61.34**	68.99	**56.42**	59.55	56
Weighted avg	85.61	85.85	85.67	**86.03**	**86.27**	**86.08**	6,001
Macro avg	**80.88**	78.72	**79.55**	80.63	**79.28**	79.51	6,001
Micro avg	85.59	85.85	85.72	**85.97**	**86.27**	**86.12**	6,001

Table H.9: Detailed Model Evaluation for QUAERO/EMEA (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 10 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Named Entities	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Disorders	**67.31**	**64.85**	**66.04**	64.64	62.46	63.52	2,115
Procedures	65.12	**67.57**	**66.28**	**65.16**	65.27	65.18	1,528
Chemical/Drugs	**72.48**	72.17	**72.27**	71.39	**72.62**	71.94	819
Living Beings	74.42	**73.87**	**74.11**	**74.73**	72.58	73.60	777
Anatomy	58.84	53.55	55.97	**59.89**	**54.08**	**56.63**	744
Physiology	41.11	**39.45**	40.17	**44.56**	38.67	**41.29**	353
Geo. Areas	**77.63**	78.88	**77.97**	75.65	**79.79**	77.51	126
Phenomena	**33.08**	**23.01**	**26.56**	31.14	21.55	25.03	123
Devices	**45.00**	38.95	41.07	43.86	**39.28**	**41.27**	97
Objects	36.80	32.46	33.10	**43.09**	**39.77**	**39.19**	83
Weighted avg	**64.87**	**63.50**	**64.05**	64.31	62.26	63.13	6,765
Macro avg	57.18	54.47	55.35	**57.41**	**54.61**	**55.52**	6,765
Micro avg	**65.10**	**63.50**	**64.29**	64.58	62.26	63.39	6,765

Table H.10: Detailed Model Evaluation for QUAERO/Medline (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 10 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Named Entities	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
dos_val	**96.08**	**97.07**	**96.56**	93.07	95.50	94.26	1,600
dos_uf	**96.24**	**96.24**	**96.24**	94.32	95.14	94.73	1,513
rhythm_tdte	**99.22**	**99.93**	**99.57**	98.43	99.79	99.10	1,320
dur_val	**98.13**	99.58	**98.85**	96.21	**99.67**	97.89	1,208
dur_ut	**98.06**	99.67	**98.85**	95.61	**99.83**	97.66	1,205
drug	**90.62**	88.81	**89.67**	85.23	**89.25**	86.77	935
d_dos_val	**96.05**	**97.03**	**96.53**	95.26	**97.03**	96.13	849
d_dos_up	96.96	**98.78**	**97.86**	**97.30**	98.18	97.73	822
inn	**79.03**	**85.84**	**82.07**	60.26	63.82	61.90	380
cma_event	**78.42**	**82.44**	**80.33**	77.35	75.90	76.60	313
d_dos_form	**90.21**	**93.90**	**92.00**	80.02	92.32	85.27	280
rhythm_perday	**95.02**	**97.15**	**95.91**	86.78	96.45	90.82	241
dos_cond	**82.92**	**86.79**	**84.51**	68.71	65.37	66.60	134
rhythm_hour	**95.20**	**98.00**	**96.46**	91.33	79.82	77.33	112
freq_ut	**94.53**	**98.22**	**96.30**	75.11	78.18	76.48	109
d_dos_form_ext	**92.60**	**81.60**	**85.69**	52.51	53.46	52.91	66
A	**85.18**	**80.99**	**79.81**	55.14	50.00	50.74	52
roa	**82.78**	**91.57**	**85.28**	56.25	54.57	55.18	46
freq_int_v1	**87.78**	**88.33**	**87.42**	55.00	50.56	52.50	31
qsp_val	**100.00**	**100.00**	**100.00**	57.78	57.78	57.78	29
rhythm_rec_ut	**90.00**	**89.44**	**89.00**	46.98	55.56	50.68	29
max_unit_val	**80.00**	**62.67**	**69.29**	40.00	32.00	35.00	28
qsp_ut	**96.00**	**100.00**	**97.78**	60.00	57.78	58.82	28
freq_int_v1_ut	**83.43**	**84.44**	**80.41**	55.00	53.33	53.14	26
rhythm_rec_val	**87.67**	**96.00**	**90.88**	48.50	53.14	49.85	24
freq_int_v2	**100.00**	**90.00**	**94.18**	60.00	53.33	56.00	20
freq_val	**93.33**	**76.67**	**82.67**	53.33	41.90	45.33	19
fasting	**100.00**	**80.67**	**84.85**	60.00	56.67	58.18	18
max_unit_uf	**66.00**	**56.67**	**59.11**	13.33	13.33	13.33	18
freq_int_v2_ut	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	10
weighted avg	**94.82***	**95.72***	**95.17***	90.22*	92.51*	91.15*	11,465
macro avg	**87.72***	**86.62***	**86.27***	66.96*	66.99*	66.29*	11,465
micro avg	**94.82**	**95.72**	**95.26**	92.33	92.51	92.42	11,465

Table H.11: Detailed Model Evaluation for PxCorpus/Task 1 (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 30 named entities. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

POS Tags	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Noun	97.30	96.88	97.09	**97.30**	**96.94**	**97.12**	20,052
Pers. Pr.	99.61	99.66	99.64	**99.64**	**99.69**	**99.66**	11,049
Adjective	95.35	**95.23**	95.29	**95.75**	95.15	**95.45**	9,179
Article	**99.72**	99.90	**99.81**	99.71	**99.90**	99.80	9,085
Punctuation	**99.94**	99.75	99.84	99.93	**99.86**	**99.90**	7,500
Number	**98.43**	**99.02**	**98.72**	98.33	98.81	98.57	4,298
Sentence	99.97	99.87	99.92	**100.00**	**99.95**	**99.97**	3,883
PastP Verb	95.14	96.56	95.84	**95.25**	**96.91**	**96.06**	3,114
Conjunction	**98.04**	98.40	98.21	97.90	**98.77**	**98.33**	2,655
Present Verb	96.60	**97.45**	97.02	**97.00**	97.25	**97.12**	2,485
Adverb	97.64	**97.66**	97.65	**97.91**	97.58	**97.74**	2,468
Poss. Pr.	**99.78**	**99.91**	**99.84**	99.73	**99.91**	99.82	2,233
Imperfect Verb	**99.53**	**99.71**	**99.62**	99.24	99.71	99.47	2,117
Pers. Pr.	**99.32**	98.30	98.80	99.12	**98.61**	**98.86**	1,583
Proper Noun	82.42	**85.72**	84.02	**83.41**	84.72	**84.03**	1,446
Inf. Verb	97.93	**98.06**	**97.97**	**98.14**	97.43	97.75	567
PresP Verb	**95.33**	94.79	**95.00**	94.43	**94.87**	94.60	512
Abbreviation	**82.30**	73.02	77.09	78.15	**76.98**	**77.51**	471
Poss. Det.	**99.36**	**99.57**	**99.46**	99.32	**99.57**	99.43	428
Demon. Pr.	**99.48**	99.75	**99.61**	99.02	**100.00**	99.50	397
Relative Pr.	98.08	95.64	96.81	**98.75**	**96.17**	**97.41**	320
Indef. Pr.	**98.35**	98.46	**98.40**	97.68	**98.52**	98.08	263
Quot. Punct.	**100.00**	98.13	99.05	99.69	**99.58**	**99.63**	232
Symbol	95.09	99.26	97.02	**98.72**	**99.61**	**99.15**	210
Past Verb	**80.04**	**70.11**	**74.53**	79.29	68.48	73.29	130
Future Verb	**79.68**	51.65	61.11	75.40	**57.87**	**65.05**	46
Cond. Verb	81.07	**83.33**	**81.44**	**91.00**	73.33	79.88	26
SubjP Verb	**81.67**	60.95	61.72	**81.67**	**64.29**	**65.05**	22
Interjection	**100.00**	**70.33**	**81.48**	85.00	53.67	63.81	18
SubjI Verb	**13.33**	**8.00**	**10.00**	10.00	6.67	8.00	16
Weighted avg	97.78	97.73	97.74	**97.81**	**97.76**	**97.78**	86,805
Macro avg	**92.02**	**88.84**	**89.73**	91.55	88.36	89.34	86,805
Micro avg	97.79	97.73	97.76	**97.83**	**97.76**	**97.79**	86,805

Table H.12: Detailed Model Evaluation for CAS (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 30 POS tags. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

POS Tags	TransBERT			cTransBERT			Support
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**	
Noun	**98.56**	**98.42**	**98.49**	98.48	98.39	98.44	39,279
Pers. Pr.	**99.66**	99.86	**99.76**	99.62	**99.87**	99.74	22,261
Article	**99.81**	**99.88**	**99.84**	99.79	99.86	99.83	18,404
Adjective	**96.69**	**95.75**	**96.21**	96.64	95.62	96.12	11,056
Punctuation	**100.00**	99.92	99.96	99.96	**100.00**	**99.98**	9,272
Sentence	**99.98**	**99.98**	**99.98**	99.98	99.95	99.97	6,016
Conjunction	**98.78**	98.82	**98.80**	98.66	**98.89**	98.77	5,653
Number	**99.06**	99.33	**99.20**	98.94	**99.37**	99.15	5,530
Poss. Pr.	99.93	**99.91**	**99.92**	**99.93**	99.89	99.91	5,480
PastP Verb	96.86	97.67	97.26	**96.96**	**97.74**	**97.34**	4,821
Present Verb	98.72	97.58	98.15	**98.81**	**97.86**	**98.33**	3,556
Adverb	98.06	**98.19**	**98.12**	**98.22**	98.01	98.11	3,490
Proper Noun	**88.02**	**91.99**	**89.93**	86.99	91.58	89.19	2,622
Future Verb	**99.53**	**99.53**	**99.53**	99.42	99.34	99.38	2,562
Inf. Verb	**99.10**	**99.51**	**99.30**	99.05	99.35	99.20	2,442
Demon. Pr.	**99.83**	**100.00**	**99.92**	**99.83**	**100.00**	**99.92**	1,796
PresP Verb	**98.32**	**98.62**	**98.47**	98.25	98.62	98.43	1,661
Indef. Pr.	**99.07**	99.34	99.20	99.01	**99.44**	**99.22**	1,210
Pers. Pr.	98.32	96.84	97.57	**98.35**	**96.85**	**97.59**	1,089
Relative Pr.	**99.27**	98.18	98.71	98.85	**98.78**	**98.81**	672
Abbreviation	64.72	**61.39**	**62.72**	**65.55**	55.15	59.21	325
Poss. Det.	**100.00**	**99.35**	**99.67**	99.38	**99.35**	99.37	312
Quot. Punct.	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	212
Noun Sing./Mass	**95.66**	**98.75**	**97.14**	95.18	98.24	96.64	161
Symbol	**100.00**	**98.75**	**99.35**	99.41	**98.75**	99.06	156
Cond. Verb	**96.72**	**90.52**	**93.39**	87.47	86.80	86.76	90
SubjP Verb	76.67	**52.05**	**60.33**	**86.00**	41.56	52.24	53
Past Verb	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	46
Imperfect Verb	96.36	**83.21**	**87.84**	**100.00**	76.51	82.76	42
Weighted avg	**98.65**	**98.63**	**98.64**	98.59	98.59	98.58	150,269
Macro avg	93.02	**91.49**	**92.03**	**93.06**	90.54	91.15	150,269
Micro avg	**98.66**	**98.63**	**98.65**	98.61	98.59	98.60	150,269

Table H.13: Detailed Model Evaluation for ESSAI (Tokenizer Analysis) - The table shows model metrics averaged over all five folds for the 29 POS tags. Bold and underline formatting are employed to emphasize the best and second-best outcomes, respectively. Medals colors indicate the rank of each metric at each fold, with gold denoting the top model. Red medals denote that every model received a Null metric. Although pastel medal colors illustrate an absolute ranking, vibrant colors indicate statistical significance using $\alpha = 0.05$. For each fold, micro average and individual class level statistical evaluations were carried out using the McNemar test. Macro and weighted averages significance were evaluated using the Wilcoxon test on labels metrics. For easy navigation, tokenizers main analysis is in Section 5.3.

Appendix I

TransBERT Vs cTransBERT: Tokenization Examples

Ratio: 7.0 (1 entity)

Entity: ['lévofloxacine'] (1 word)
TransBERT: ['▁lévofloxacine'] (1 token)
CamemBERT: ['▁l', 'é', 'vo', 'flo', 'x', 'a', 'cine'] (Δ+6)
DrBERT: ['▁lévofloxacine'] (Δ0)

Ratio: 6.0 (1 entity)

Entity: ['dexaméthasone'] (1 word)
TransBERT: ['▁dexaméthasone'] (1 token)
CamemBERT: ['▁de', 'x', 'a', 'méth', 'as', 'one'] (Δ+5)
DrBERT: ['▁dexaméthasone'] (Δ0)

Ratio: 5.0 (46 entities)

Entity: ['thyroïdite'] (1 word)
TransBERT: ['▁thyroïdite'] (1 token)
CamemBERT: ['▁', 'thy', 'r', 'oïd', 'ite'] (Δ+4)
DrBERT: ['▁thyroï', 'd', 'ite'] (Δ+2)

Entity: ['lymphoblastique'] (1 word)
TransBERT: ['▁lymphoblastique'] (1 token)
CamemBERT: ['▁l', 'ymph', 'o', 'blast', 'ique'] (Δ+4)
DrBERT: ['▁lymph', 'oblastique'] (Δ+1)

Entity: ['Massachusetts'] (1 word)
TransBERT: ['__Massachusetts'] (1 token)
CamemBERT: ['__Massa', 'chu', 's', 'ett', 's'] (Δ+4)
DrBERT: ['__Mass', 'ach', 'us', 'et', 'ts'] (Δ+4)

Entity: ['méthotrexate'] (1 word)
TransBERT: ['__méthotrexate'] (1 token)
CamemBERT: ['__mé', 'tho', 'tre', 'x', 'ate'] (Δ+4)
DrBERT: ['__méthotrexate'] (Δ0)

Entity: ['antirétroviral'] (1 word)
TransBERT: ['__antirétroviral'] (1 token)
CamemBERT: ['__anti', 'ré', 'tro', 'vir', 'al'] (Δ+4)
DrBERT: ['__antirétroviral'] (Δ0)

Entity: ['Absorption'] (1 word)
TransBERT: ['__Absorption'] (1 token)
CamemBERT: ['__Ab', 's', 'or', 'p', 'tion'] (Δ+4)
DrBERT: ['__Absorption'] (Δ0)

Entity: ['céphalosporine'] (1 word)
TransBERT: ['__céphalosporine'] (1 token)
CamemBERT: ['__c', 'épha', 'los', 'por', 'ine'] (Δ+4)
DrBERT: ['__céphal', 'osporine'] (Δ+1)

Entity: ['antécédents'] (1 word)
TransBERT: ['__antécédents'] (1 token)
CamemBERT: ['__an', 'té', 'cé', 'dent', 's'] (Δ+4)
DrBERT: ['__antécédents'] (Δ0)

Entity: ['antirétroviraux'] (1 word)
TransBERT: ['__antirétroviraux'] (1 token)
CamemBERT: ['__anti', 'ré', 'tro', 'vir', 'aux'] (Δ+4)
DrBERT: ['__antirétroviraux'] (Δ0)

Entity: ['phosphatases'] (1 word)
TransBERT: ['__phosphatases'] (1 token)
CamemBERT: ['__pho', 's', 'pha', 'tas', 'es'] (Δ+4)
DrBERT: ['__phosphatases'] (Δ0)

Ratio: 4.5 (1 entity)

Entity: ['adénocarcinome', 'kystique'] (2 words)
TransBERT: ['__adénocarcinome', '__kystique'] (2 tokens)
CamemBERT: ['__ad', 'éno', 'car', 'ci', 'nome', '__', 'ky', 's', 'tique'] (Δ+7)
DrBERT: ['__adénocarcinome', '__kys', 'tique'] (Δ+1)

Ratio: 4.0 (128 entities)

Entity: ['convulsions'] (1 word)
TransBERT: ['__convulsions'] (1 token)
CamemBERT: ['__con', 'vul', 's', 'ions'] (Δ+3)
DrBERT: ['__convulsions'] (Δ0)

Entity: ['myocardite', 'fulminante'] (2 words)
TransBERT: ['__myocardite', '__fulminante'] (2 tokens)
CamemBERT: ['__m', 'yo', 'car', 'dite', '__', 'ful', 'min', 'ante'] (Δ+6)
DrBERT: ['__myoc', 'ardite', '__fulmin', 'ante'] (Δ+2)

Entity: ['lévodopa'] (1 word)
TransBERT: ['__lévodopa'] (1 token)
CamemBERT: ['__l', 'év', 'odo', 'pa'] (Δ+3)
DrBERT: ['__lévodopa'] (Δ0)

Entity: ['médullaire'] (1 word)
TransBERT: ['__médullaire'] (1 token)
CamemBERT: ['__m', 'éd', 'ul', 'laire'] (Δ+3)
DrBERT: ['__médullaire'] (Δ0)

Entity: ['myocardique'] (1 word)
TransBERT: ['__myocardique'] (1 token)
CamemBERT: ['__m', 'yo', 'card', 'ique'] (Δ+3)
DrBERT: ['__myocardique'] (Δ0)

Entity: ['vésicules'] (1 word)
TransBERT: ['__vésicules'] (1 token)
CamemBERT: ['__vé', 's', 'icule', 's'] (Δ+3)
DrBERT: ['__vésicules'] (Δ0)

Entity: ['cholestéatome'] (1 word)
TransBERT: ['__cholestéatome'] (1 token)
CamemBERT: ['__cho', 'les', 'té', 'atome'] (Δ+3)
DrBERT: ['__ch', 'oles', 'té', 'at', 'ome'] (Δ+4)

Entity: ['prolapsus'] (1 word)
TransBERT: ['__prolapsus'] (1 token)
CamemBERT: ['__pro', 'la', 'ps', 'us'] (Δ+3)
DrBERT: ['__prolapsus'] (Δ0)

Entity: ['métastase'] (1 word)
TransBERT: ['__métastase'] (1 token)
CamemBERT: ['__méta', 'sta', 's', 'e'] (Δ+3)
DrBERT: ['__métastase'] (Δ0)

Entity: ['microsomes'] (1 word)
TransBERT: ['__microsomes'] (1 token)
CamemBERT: ['__micro', 's', 'ome', 's'] (Δ+3)
DrBERT: ['__micros', 'omes'] (Δ+1)

Ratio: 3.5 (25 entities)

Entity: ['fistule', 'cholécysto'] (2 words)
TransBERT: ['__fistule', '__cholécysto'] (2 tokens)
CamemBERT: ['__fist', 'ule', '__cho', 'lé', 'cy', 's', 'to'] (Δ+5)
DrBERT: ['__fistule', '__cholécys', 'to'] (Δ+1)

Entity: ['cardiomyopathie', 'dilatée'] (2 words)
TransBERT: ['__cardiomyopathie', '__dilatée'] (2 tokens)
CamemBERT: ['__cardio', 'my', 'opathie', '__d', 'il', 'a', 'tée'] (Δ+5)
DrBERT: ['__cardiomyopathie', '__dila', 'tée'] (Δ+1)

Entity: ['épaississement', 'péritonéal'] (2 words)
TransBERT: ['__épaississement', '__péritonéal'] (2 tokens)
CamemBERT: ['__épais', 's', 'issement', '__péri', 'ton', 'é', 'al'] (Δ+5)
DrBERT: ['__épaississement', '__péritoné', 'al'] (Δ+1)

Entity: ['adénomes', 'parathyroïdiens', 'ectopiques'] (3 words)
TransBERT: ['__adénomes', '__parathyroïdien', 's', '__ectopiques'] (4 tokens)
CamemBERT: ['__a', 'dé', 'nome', 's', '__par', 'ath', 'y', 'r', 'oïd', 'iens', '__', 'ect', 'op', 'iques'] (Δ+10)
DrBERT: ['__adénomes', '__parathyroï', 'diens', '__ect', 'opiques'] (Δ+1)

Entity: ['Anatomie', 'stéréotaxique'] (2 words)
TransBERT: ['__Anatomie', '__stéréotaxique'] (2 tokens)
CamemBERT: ['__Ana', 'tom', 'ie', '__stéréo', 'ta', 'x', 'ique'] (Δ+5)
DrBERT: ['__Anatomie', '__stéréotax', 'ique'] (Δ+1)

Entity: ['paucisymptomatique'] (1 word)
TransBERT: ['__pauci', 'symptomatique'] (2 tokens)
CamemBERT: ['__pa', 'uci', 's', 'y', 'mp', 'to', 'matique'] (Δ+5)
DrBERT: ['__p', 'auc', 'is', 'ymptom', 'atique'] (Δ+3)

Entity: ['acétylsalicylique'] (1 word)
TransBERT: ['__a', 'cétylsalicylique'] (2 tokens)
CamemBERT: ['__ac', 'ét', 'yl', 's', 'ali', 'cy', 'lique'] (Δ+5)
DrBERT: ['__acétyl', 'salicylique'] (Δ0)

Entity: ['protéinurie', 'néphrotique'] (2 words)
TransBERT: ['__protéinurie', '__néphrotique'] (2 tokens)
CamemBERT: ['__', 'proté', 'in', 'urie', '__né', 'phro', 'tique'] (Δ+5)
DrBERT: ['__protéinurie', '__néphrotique'] (Δ0)

Entity: ['convulsions', 'fébriles'] (2 words)
TransBERT: ['__convulsions', '__fébriles'] (2 tokens)
CamemBERT: ['__con', 'vul', 's', 'ions', '__fé', 'bri', 'les'] (Δ+5)
DrBERT: ['__convulsions', '__fébriles'] (Δ0)

Entity: ['Plasmodium', 'vivax'] (2 words)
TransBERT: ['__Plasmodium', '__vivax'] (2 tokens)
CamemBERT: ['__Pla', 's', 'mo', 'dium', '__', 'viv', 'ax'] (Δ+5)
DrBERT: ['__Plasmodium', '__viv', 'ax'] (Δ+1)

Ratio: 3.3 (5 entities)

Entity: ['cardiomyopathie', 'dilatée', 'hypertensive'] (3 words)
TransBERT: ['__cardiomyopathie', '__dilatée', '__hypertensive'] (3 tokens)
CamemBERT: ['__cardio', 'my', 'opathie', '__d', 'il', 'a', 'tée', '__hyper', 'tens', 'ive'] (Δ+7)
DrBERT: ['__cardiomyopathie', '__dila', 'tée', '__hypertensive'] (Δ+1)

Entity: ['spasme', 'coronarien', 'occlusif'] (3 words)
TransBERT: ['__spasme', '__coronarien', '__occlusif'] (3 tokens)
CamemBERT: ['__spa', 's', 'me', '__cor', 'on', 'arien', '__', 'oc', 'clus', 'if'] (Δ+7)
DrBERT: ['__spasme', '__coronarien', '__occl', 'usif'] (Δ+1)

Entity: ['adénomes', 'parathyroïdiens'] (2 words)
TransBERT: ['__adénomes', '__parathyroïdien', 's'] (3 tokens)
CamemBERT: ['__a', 'dé', 'nome', 's', '__par', 'ath', 'y', 'r', 'oïd', 'iens'] (Δ+7)
DrBERT: ['__adénomes', '__parathyroï', 'diens'] (Δ0)

Entity: ['fistule', 'cholécysto', 'duodénale'] (3 words)
TransBERT: ['__fistule', '__cholécysto', '__duodénale'] (3 tokens)
CamemBERT: ['__fist', 'ule', '__cho', 'lé', 'cy', 's', 'to', '__duo', 'dé', 'nale'] (Δ+7)
DrBERT: ['__fistule', '__cholécys', 'to', '__duodén', 'ale'] (Δ+2)

Entity: ['lympho-histiocytose'] (1 word)
TransBERT: ['__lympho', '-', 'histiocytose'] (3 tokens)
CamemBERT: ['__l', 'ymph', 'o', '-', 'hi', 's', 'tio', 'cy', 'tos', 'e'] (Δ+7)
DrBERT: ['__lymph', 'o', '-', 'h', 'isti', 'ocytose'] (Δ+3)

Ratio: 3.0 (330 entities)

Entity: ['infectieux'] (1 word)
TransBERT: ['▁infectieux'] (1 token)
CamemBERT: ['▁', 'infect', 'ieux'] (Δ+2)
DrBERT: ['▁infectieux'] (Δ0)

Entity: ['fongicides', 'topiques'] (2 words)
TransBERT: ['▁fongicides', '▁topiques'] (2 tokens)
CamemBERT: ['▁f', 'ong', 'icide', 's', '▁top', 'iques'] (Δ+4)
DrBERT: ['▁fong', 'icides', '▁topiques'] (Δ+1)

Entity: ['des', 'ostéoblastes'] (2 words)
TransBERT: ['▁des', '▁ostéoblastes'] (2 tokens)
CamemBERT: ['▁des', '▁', 'ost', 'éo', 'blast', 'es'] (Δ+4)
DrBERT: ['▁des', '▁ostéoblastes'] (Δ0)

Entity: ['positivité'] (1 word)
TransBERT: ['▁positivité'] (1 token)
CamemBERT: ['▁', 'posit', 'ivité'] (Δ+2)
DrBERT: ['▁positivité'] (Δ0)

Entity: ['perforation', 'intestinale'] (2 words)
TransBERT: ['▁perforation', '▁intestinale'] (2 tokens)
CamemBERT: ['▁perf', 'or', 'ation', '▁', 'intestin', 'ale'] (Δ+4)
DrBERT: ['▁perforation', '▁intestinale'] (Δ0)

Entity: ['puberté'] (1 word)
TransBERT: ['▁puberté'] (1 token)
CamemBERT: ['▁pu', 'bert', 'é'] (Δ+2)
DrBERT: ['▁puberté'] (Δ0)

Entity: ['stéroïdes', 'génitaux'] (2 words)
TransBERT: ['▁stéroïdes', '▁génitaux'] (2 tokens)
CamemBERT: ['▁st', 'ér', 'oïdes', '▁gén', 'it', 'aux'] (Δ+4)
DrBERT: ['▁stéroïdes', '▁génitaux'] (Δ0)

Entity: ['antibiotiques'] (1 word)
TransBERT: ['__antibiotiques'] (1 token)
CamemBERT: ['__anti', 'biotique', 's'] (Δ+2)
DrBERT: ['__antibiotiques'] (Δ0)

Entity: ['aortique'] (1 word)
TransBERT: ['__aortique'] (1 token)
CamemBERT: ['__a', 'or', 'tique'] (Δ+2)
DrBERT: ['__aortique'] (Δ0)

Entity: ['convexe'] (1 word)
TransBERT: ['__convexe'] (1 token)
CamemBERT: ['__con', 'vex', 'e'] (Δ+2)
DrBERT: ['__con', 'vexe'] (Δ+1)

Ratio: 2.8 (4 entities)

Entity: ['méningo', 'encéphalite', 'tuberculeuse'] (3 words)
TransBERT: ['__méningo', '__', 'encéphalite', '__tuberculeuse'] (4 tokens)
CamemBERT: ['__mé', 'ning', 'o', '__en', 'c', 'épha', 'lite', '__tube', 'r', 'cule', 'use'] (Δ+7)
DrBERT: ['__méning', 'o', '__encéphal', 'ite', '__tubercule', 'use'] (Δ+2)

Entity: ['lympho-histiocytose', 'familiale'] (2 words)
TransBERT: ['__lympho', '-', 'histiocytose', '__familiale'] (4 tokens)
CamemBERT: ['__l', 'ymph', 'o', '-', 'hi', 's', 'tio', 'cy', 'tos', 'e', '__familiale'] (Δ+7)
DrBERT: ['__lymph', 'o', '-', 'h', 'isti', 'ocytose', '__familiale'] (Δ+3)

Entity: ['polyvinylpyrrolidone', 'iodée'] (2 words)
TransBERT: ['__poly', 'vinylpyrrolidone', '__iodé', 'e'] (4 tokens)
CamemBERT: ['__poly', 'vin', 'yl', 'py', 'rro', 'li', 'don', 'e', '__', 'io', 'dée'] (Δ+7)
DrBERT: ['__poly', 'vin', 'yl', 'py', 'r', 'rol', 'idone', '__i', 'odée'] (Δ+5)

Entity: ['méningite', 'a', 'Listeria', 'monocytogenes'] (4 words)
TransBERT: ['__méningite', '__a', '__Listeria', '__monocytogenes'] (4 tokens)
CamemBERT: ['__mé', 'ning', 'ite', '__a', '__Liste', 'ria', '__mono', 'cy', 'to', 'gene', 's'] (Δ+7)
DrBERT: ['__méningite', '__a', '__Listeria', '__monocytogenes'] (Δ0)

Ratio: 2.7 (11 entities)

Entity: ['glomérulonéphrites'] (1 word)
TransBERT: ['__g', 'lomérulonéphrite', 's'] (3 tokens)
CamemBERT: ['__g', 'lom', 'ér', 'ulo', 'né', 'ph', 'rite', 's'] (Δ+5)
DrBERT: ['__glomérul', 'onéph', 'rites'] (Δ0)

Entity: ['infarctus', 'de', 'myocarde'] (3 words)
TransBERT: ['__infarctus', '__de', '__myocarde'] (3 tokens)
CamemBERT: ['__inf', 'arc', 'tu', 's', '__de', '__my', 'oc', 'arde'] (Δ+5)
DrBERT: ['__infarctus', '__de', '__myocarde'] (Δ0)

Entity: ['néphropathie', 'interstitielle', 'chronique'] (3 words)
TransBERT: ['__néphropathie', '__interstitielle', '__chronique'] (3 tokens)
CamemBERT: ['__né', 'phro', 'pathie', '__inter', 's', 'titi', 'elle', '__chronique'] (Δ+5)
DrBERT: ['__néphropathie', '__interstitielle', '__chronique'] (Δ0)

Entity: ['acide', 'acétylsalicylique'] (2 words)
TransBERT: ['__acide', '__a', 'cétylsalicylique'] (3 tokens)
CamemBERT: ['__acide', '__ac', 'ét', 'yl', 's', 'ali', 'cy', 'lique'] (Δ+5)
DrBERT: ['__acide', '__acétyl', 'salicylique'] (Δ0)

Entity: ['masse', 'ovarienne', 'kystique'] (3 words)
TransBERT: ['__masse', '__ovarienne', '__kystique'] (3 tokens)
CamemBERT: ['__masse', '__', 'ova', 'rienne', '__', 'ky', 's', 'tique'] (Δ+5)
DrBERT: ['__masse', '__ovarienne', '__kys', 'tique'] (Δ+1)

Entity: ['thrombose', 'veineuse', 'iliaque'] (3 words)
TransBERT: ['__thrombose', '__veineuse', '__iliaque'] (3 tokens)
CamemBERT: ['__thrombo', 's', 'e', '__veine', 'use', '__il', 'ia', 'que'] (Δ+5)
DrBERT: ['__thrombose', '__veineuse', '__iliaque'] (Δ0)

Entity: ['alkylsulfonyle'] (1 word)
TransBERT: ['__alkyl', 'sulfonyl', 'e'] (3 tokens)
CamemBERT: ['__a', 'lk', 'yl', 's', 'ulf', 'on', 'y', 'le'] (Δ+5)
DrBERT: ['__alkyl', 'sulf', 'on', 'yle'] (Δ+1)

Entity: ['chondrolyse', 'dégénérative'] (2 words)
TransBERT: ['__chondro', 'lyse', '__dégénérative'] (3 tokens)
CamemBERT: ['__', 'chon', 'dro', 'lyse', '__dé', 'géné', 'r', 'ative'] (Δ+5)
DrBERT: ['__chondro', 'lyse', '__dé', 'générative'] (Δ+1)

Entity: ['néphropathies', 'interstitielles'] (2 words)
TransBERT: ['__néphropathie', 's', '__interstitielles'] (3 tokens)
CamemBERT: ['__né', 'phro', 'pathie', 's', '__inter', 's', 'titi', 'elles'] (Δ+5)
DrBERT: ['__néph', 'ropathies', '__interstiti', 'elles'] (Δ+1)

Entity: ['affections', 'tumorales', 'malignes'] (3 words)
TransBERT: ['__affections', '__tumorales', '__malignes'] (3 tokens)
CamemBERT: ['__affection', 's', '__tu', 'mor', 'ales', '__ma', 'ligne', 's'] (Δ+5)
DrBERT: ['__affections', '__tumorales', '__malignes'] (Δ0)

Ratio: 2.5 (135 entities)

Entity: ['méthode', 'chromogénique'] (2 words)
TransBERT: ['__méthode', '__chromogénique'] (2 tokens)
CamemBERT: ['__méthode', '__ch', 'r', 'omo', 'génique'] (Δ+3)
DrBERT: ['__méthode', '__chrom', 'ogénique'] (Δ+1)

Entity: ['polyester', 'sulfurique'] (2 words)
TransBERT: ['__polyester', '__sulfurique'] (2 tokens)
CamemBERT: ['__polyester', '__s', 'ulf', 'ur', 'ique'] (Δ+3)
DrBERT: ['__poly', 'ester', '__sulf', 'urique'] (Δ+2)

Entity: ['manifestations', 'hypoglycémiques'] (2 words)
TransBERT: ['__manifestations', '__hypoglycémiques'] (2 tokens)
CamemBERT: ['__manifestations', '__hypo', 'glyc', 'émique', 's'] (Δ+3)
DrBERT: ['__manifestations', '__hypoglyc', 'émiques'] (Δ+1)

Entity: ['saignements', 'intracrâniens'] (2 words)
TransBERT: ['__saignements', '__intracrâniens'] (2 tokens)
CamemBERT: ['__saignement', 's', '__intra', 'crânien', 's'] (Δ+3)
DrBERT: ['__saignements', '__intracrân', 'iens'] (Δ+1)

Entity: ['capsulotomie'] (1 word)
TransBERT: ['__capsul', 'otomie'] (2 tokens)
CamemBERT: ['__cap', 's', 'ulo', 'tom', 'ie'] (Δ+3)
DrBERT: ['__caps', 'ul', 'otomie'] (Δ+1)

Entity: ['niveau', 'abdominal'] (2 words)
TransBERT: ['__niveau', '__abdominal'] (2 tokens)
CamemBERT: ['__niveau', '__ab', 'dom', 'in', 'al'] (Δ+3)
DrBERT: ['__niveau', '__abdominal'] (Δ0)

Entity: ['du', 'parenchyme'] (2 words)
TransBERT: ['__du', '__parenchyme'] (2 tokens)
CamemBERT: ['__du', '__par', 'en', 'chy', 'me'] (Δ+3)
DrBERT: ['__du', '__parenchyme'] (Δ0)

Entity: ['claudication', 'intermittente'] (2 words)
TransBERT: ['__claudication', '__intermittente'] (2 tokens)
CamemBERT: ['__cl', 'au', 'dication', '__intermittent', 'e'] (Δ+3)
DrBERT: ['__cl', 'audication', '__intermittente'] (Δ+1)

Entity: ['lymphocytaires'] (1 word)
TransBERT: ['__lymphocytaire', 's'] (2 tokens)
CamemBERT: ['__l', 'ymph', 'oc', 'y', 'taires'] (Δ+3)
DrBERT: ['__lymph', 'ocytaires'] (Δ0)

Entity: ['coma', 'hypercapnique'] (2 words)
TransBERT: ['__coma', '__hypercapnique'] (2 tokens)
CamemBERT: ['__com', 'a', '__hyper', 'cap', 'nique'] (Δ+3)
DrBERT: ['__coma', '__hyperc', 'ap', 'n', 'ique'] (Δ+3)

Ratio: 2.4 (2 entities)

Entity: ['kystique', 'congénitale', 'des', 'voies', 'biliaires'] (5 words)
TransBERT: ['__kystique', '__congénitale', '__des', '__voies', '__biliaires'] (5 tokens)
CamemBERT: ['__', 'ky', 's', 'tique', '__con', 'génital', 'e', '__des', '__voies', '__', 'bili', 'aires'] (Δ+7)
DrBERT: ['__kys', 'tique', '__congénitale', '__des', '__voies', '__biliaires'] (Δ+1)

Entity: ['Déficit', 'en', 'glutathion-peroxydase'] (3 words)
TransBERT: ['__Déficit', '__en', '__glutathion', '-', 'peroxydase'] (5 tokens)
CamemBERT: ['__Défi', 'cit', '__en', '__glu', 't', 'ath', 'ion', '-', 'per', 'oxy', 'das', 'e'] (Δ+7)
DrBERT: ['__Déficit', '__en', '__glutathion', '-', 'per', 'oxydase'] (Δ+1)

Ratio: 2.3 (61 entities)

Entity: ['hématome', 'sous', 'capsulaire'] (3 words)
TransBERT: ['__hématome', '__sous', '__capsulaire'] (3 tokens)
CamemBERT: ['__', 'hémat', 'ome', '__sous', '__cap', 's', 'ulaire'] (Δ+4)
DrBERT: ['__hématome', '__sous', '__caps', 'ulaire'] (Δ+1)

Entity: ['lymphoplasmocytes'] (1 word)
TransBERT: ['__lympho', 'plasm', 'ocytes'] (3 tokens)
CamemBERT: ['__l', 'ymph', 'op', 'las', 'mo', 'cy', 'tes'] (Δ+4)
DrBERT: ['__lymph', 'oplasm', 'ocytes'] (Δ0)

Entity: ['cardioversion', 'ou', 'défibrillation'] (3 words)
TransBERT: ['__cardioversion', '__ou', '__défibrillation'] (3 tokens)
CamemBERT: ['__cardio', 'version', '__ou', '__défi', 'br', 'il', 'lation'] (Δ+4)
DrBERT: ['__cardio', 'version', '__ou', '__défib', 'rillation'] (Δ+2)

Entity: ['antithrombotique', 'injectable'] (2 words)
TransBERT: ['__', 'antithrombotique', '__injectable'] (3 tokens)
CamemBERT: ['__anti', 'thro', 'mbo', 'tique', '__in', 'ject', 'able'] (Δ+4)
DrBERT: ['__anti', 'thrombo', 'tique', '__injectable'] (Δ+1)

Entity: ['arylsulfonamides'] (1 word)
TransBERT: ['__aryl', 'sulfonamide', 's'] (3 tokens)
CamemBERT: ['__ar', 'yl', 's', 'ulf', 'on', 'ami', 'des'] (Δ+4)
DrBERT: ['__ar', 'yl', 'sulf', 'on', 'amides'] (Δ+2)

Entity: ['dilatation', 'des', 'bronches'] (3 words)
TransBERT: ['__dilatation', '__des', '__bronches'] (3 tokens)
CamemBERT: ['__d', 'il', 'a', 'tation', '__des', '__bron', 'ches'] (Δ+4)
DrBERT: ['__dilatation', '__des', '__bronches'] (Δ0)

Entity: ['spondylodiscite'] (1 word)
TransBERT: ['__spondylo', 'disc', 'ite'] (3 tokens)
CamemBERT: ['__s', 'pond', 'y', 'lo', 'dis', 'ci', 'te'] (Δ+4)
DrBERT: ['__spondyl', 'odis', 'c', 'ite'] (Δ+1)

Entity: ['Complications', 'laryngées'] (2 words)
TransBERT: ['__Complications', '__laryngée', 's'] (3 tokens)
CamemBERT: ['__Com', 'plication', 's', '__la', 'ry', 'ng', 'ées'] (Δ+4)
DrBERT: ['__Complications', '__laryng', 'ées'] (Δ0)

Entity: ['ostéocalcine', 'sérique'] (2 words)
TransBERT: ['__', 'ostéocalcine', '__sérique'] (3 tokens)
CamemBERT: ['__', 'ost', 'éo', 'cal', 'cine', '__s', 'érique'] (Δ+4)
DrBERT: ['__osté', 'oc', 'alc', 'ine', '__sérique'] (Δ+2)

Entity: ['bilirubine', 'conjuguée', 'augmentée'] (3 words)
TransBERT: ['__bilirubine', '__conjuguée', '__augmentée'] (3 tokens)
CamemBERT: ['__', 'bili', 'rub', 'ine', '__conjugué', 'e', '__augmentée'] (Δ+4)
DrBERT: ['__bilirubine', '__conjug', 'uée', '__augmentée'] (Δ+1)

Ratio: 2.2 (26 entities)

Entity: ['Méthylhydroxypropylcellulose'] (1 word)
TransBERT: ['__M', 'éthyl', 'hydroxypropyl', 'cellulose'] (4 tokens)
CamemBERT: ['__M', 'éthyl', 'hydr', 'oxy', 'prop', 'yl', 'cell', 'ul', 'ose'] (Δ+5)
DrBERT: ['__Méth', 'yl', 'hydrox', 'ypropylcellulose'] (Δ0)

Entity: ['atrophie', 'cérébrale', 'et', 'médullaire'] (4 words)
TransBERT: ['__atrophie', '__cérébrale', '__et', '__médullaire'] (4 tokens)
CamemBERT: ['__a', 'trophi', 'e', '__cérébrale', '__et', '__m', 'éd', 'ul', 'laire'] (Δ+5)
DrBERT: ['__atrophie', '__cérébrale', '__et', '__médullaire'] (Δ0)

Entity: ['convulsions', 'hypocalcémiques'] (2 words)
TransBERT: ['__convulsions', '__hypo', 'calc', 'émiques'] (4 tokens)
CamemBERT: ['__con', 'vul', 's', 'ions', '__hypo', 'cal', 'c', 'émique', 's'] (Δ+5)
DrBERT: ['__convulsions', '__hyp', 'oc', 'alc', 'émiques'] (Δ+1)

Entity: ['Actinomycose', 'cervico-faciale'] (2 words)
TransBERT: ['__Actin', 'omycose', '__cervico', '-', 'faciale'] (5 tokens)
CamemBERT: ['__Act', 'ino', 'my', 'cos', 'e', '__ce', 'rv', 'ico', '-', 'facial', 'e'] (Δ+6)
DrBERT: ['__Actin', 'omyc', 'ose', '__cervico', '-', 'faciale'] (Δ+1)

Entity: ['carcinomes', 'épidermoïdes', 'cutanés'] (3 words)
TransBERT: ['__carcinomes', '__épidermoïde', 's', '__cutanés'] (4 tokens)
CamemBERT: ['__car', 'ci', 'nome', 's', '__épi', 'derm', 'oïdes', '__cutané', 's'] (Δ+5)
DrBERT: ['__carcinomes', '__épiderm', 'oïdes', '__cutanés'] (Δ0)

Entity: ['glutamine', 'gamma-glutamyl', 'transferase'] (3 words)
TransBERT: ['__glutamine', '__gamma', '-', 'glutamyl', '__', 'transferase'] (6 tokens)
CamemBERT: ['__glu', 't', 'amine', '__g', 'amma', '-', 'g', 'lut', 'am', 'yl', '__trans', 'fer', 'ase'] (Δ+7)
DrBERT: ['__glutamine', '__gamma', '-', 'glut', 'amyl', '__transfer', 'ase'] (Δ+1)

Entity: ['hernie', 'diaphragmatiques', 'congénitales'] (3 words)
TransBERT: ['__hernie', '__diaphragmatique', 's', '__congénitales'] (4 tokens)
CamemBERT: ['__her', 'nie', '__dia', 'ph', 'rag', 'matiques', '__con', 'génital', 'es'] (Δ+5)
DrBERT: ['__hernie', '__diaphrag', 'matiques', '__congénitales'] (Δ0)

Entity: ['goitre', 'multihétéronodulaire'] (2 words)
TransBERT: ['__goitre', '__multi', 'hétéro', 'nodulaire'] (4 tokens)
CamemBERT: ['__go', 'it', 're', '__multi', 'h', 'été', 'ron', 'od', 'ulaire'] (Δ+5)
DrBERT: ['__g', 'oit', 're', '__multi', 'hé', 'téron', 'od', 'ulaire'] (Δ+4)

Entity: ['rétinites', 'à', 'cytomégalovirus'] (3 words)
TransBERT: ['__rétinite', 's', '__à', '__cytomégalovirus'] (4 tokens)
CamemBERT: ['__ré', 'tin', 'ites', '__à', '__cyto', 'm', 'égal', 'o', 'virus'] (Δ+5)
DrBERT: ['__rétin', 'ites', '__à', '__cyt', 'omégalovirus'] (Δ+1)

Entity: ['obstructions', 'biliaires', 'tumorales'] (3 words)
TransBERT: ['__obstruction', 's', '__biliaires', '__tumorales'] (4 tokens)
CamemBERT: ['__', 'obstruction', 's', '__', 'bili', 'aires', '__tu', 'mor', 'ales'] (Δ+5)
DrBERT: ['__obs', 'tructions', '__biliaires', '__tumorales'] (Δ0)

Ratio: 2.1 (1 entity)

Entity: ['analogues', 'nucléosidiques', 'inhibiteurs', 'de', 'la', 'transcriptase', 'inverse'] (7 words)
TransBERT: ['__analogues', '__nucléosidique', 's', '__inhibiteurs', '__de', '__la', '__transcriptase', '__inverse'] (8 tokens)
CamemBERT: ['__analogue', 's', '__', 'nuclé', 'os', 'idique', 's', '__in', 'hibi', 'teurs', '__de', '__la', '__', 'tran', 'script', 'ase', '__inverse'] (Δ+9)
DrBERT: ['__analogues', '__nucléosi', 'diques', '__inhibiteurs', '__de', '__la', '__transcriptase', '__inverse'] (Δ0)

Ratio: 2.0 (740 entities)

Entity: ['nasal'] (1 word)
TransBERT: ['__nasal'] (1 token)
CamemBERT: ['__na', 'sal'] (Δ+1)
DrBERT: ['__nasal'] (Δ0)

Entity: ['allergiques'] (1 word)
TransBERT: ['▁allergiques'] (1 token)
CamemBERT: ['▁allergique', 's'] (Δ+1)
DrBERT: ['▁allergiques'] (Δ0)

Entity: ['HDL'] (1 word)
TransBERT: ['▁HDL'] (1 token)
CamemBERT: ['▁HD', 'L'] (Δ+1)
DrBERT: ['▁HDL'] (Δ0)

Entity: ['particule'] (1 word)
TransBERT: ['▁particule'] (1 token)
CamemBERT: ['▁part', 'icule'] (Δ+1)
DrBERT: ['▁particule'] (Δ0)

Entity: ['Hémophiles'] (1 word)
TransBERT: ['▁Hémo', 'philes'] (2 tokens)
CamemBERT: ['▁H', 'émo', 'phile', 's'] (Δ+2)
DrBERT: ['▁Hém', 'ophiles'] (Δ0)

Entity: ['fluctuations'] (1 word)
TransBERT: ['▁fluctuations'] (1 token)
CamemBERT: ['▁fluctuation', 's'] (Δ+1)
DrBERT: ['▁fluctuations'] (Δ0)

Entity: ['étoposide'] (1 word)
TransBERT: ['▁', 'étoposide'] (2 tokens)
CamemBERT: ['▁é', 'top', 'o', 'side'] (Δ+2)
DrBERT: ['▁ét', 'oposide'] (Δ0)

Entity: ['coagulation'] (1 word)
TransBERT: ['▁coagulation'] (1 token)
CamemBERT: ['▁coagul', 'ation'] (Δ+1)
DrBERT: ['▁coagulation'] (Δ0)

Entity: ['hamster', 'doré'] (2 words)
TransBERT: ['__hamster', '__doré'] (2 tokens)
CamemBERT: ['__', 'ham', 'ster', '__doré'] (Δ+2)
DrBERT: ['__hams', 'ter', '__d', 'oré'] (Δ+2)

Entity: ['translocation'] (1 word)
TransBERT: ['__translocation'] (1 token)
CamemBERT: ['__trans', 'location'] (Δ+1)
DrBERT: ['__translocation'] (Δ0)

Ratio: 1.9 (4 entities)

Entity: ['drépanocytose', 'hétérozygote', 'composite', 'SC'] (4 words)
TransBERT: ['__d', 'ré', 'pan', 'ocytose', '__hétérozygote', '__composite', '__SC'] (7 tokens)
CamemBERT: ['__d', 'ré', 'pan', 'oc', 'y', 'tos', 'e', '__hétéro', 'zy', 'got', 'e', '__composite', '__SC'] (Δ+6)
DrBERT: ['__drépanocytose', '__hétérozygote', '__composite', '__SC'] (Δ-3)

Entity: ['infarctus', 'de', 'myocarde', 'antéro', 'septal'] (5 words)
TransBERT: ['__infarctus', '__de', '__myocarde', '__', 'ant', 'éro', '__septal'] (7 tokens)
CamemBERT: ['__inf', 'arc', 'tu', 's', '__de', '__my', 'oc', 'arde', '__', 'ant', 'éro', '__sept', 'al'] (Δ+6)
DrBERT: ['__infarctus', '__de', '__myocarde', '__antéro', '__sept', 'al'] (Δ-1)

Entity: ['processus', 'tumoral', 'pariéto', 'occipital', 'droit'] (5 words)
TransBERT: ['__processus', '__tumoral', '__par', 'i', 'éto', '__occipital', '__droit'] (7 tokens)
CamemBERT: ['__processus', '__tu', 'm', 'oral', '__par', 'ié', 'to', '__', 'oc', 'cip', 'it', 'al', '__droit'] (Δ+6)
DrBERT: ['__processus', '__tumoral', '__pariét', 'o', '__occip', 'ital', '__droit'] (Δ0)

Entity: ['Staphylococcus', 'aureus', 'résistant', 'à', 'la', 'méticilline'] (6 words)
TransBERT: ['__Staphylococcus', '__aureus', '__résistant', '__à', '__la', '__mét', 'i', 'cilline'] (8 tokens)
CamemBERT: ['__Sta', 'phyl', 'oco', 'c', 'cus', '__au', 're', 'us', '__résistant', '__à', '__la', '__mé', 'tic', 'il', 'line'] (Δ+7)
DrBERT: ['__Staphylococcus', '__aureus', '__résistant', '__à', '__la', '__méticilline'] (Δ-2)

Ratio: 1.8 (88 entities)

Entity: ['Test', 'de', 'freinage', 'à', 'la', 'dexaméthasone'] (6 words)
TransBERT: ['__Test', '__de', '__freinage', '__à', '__la', '__dexaméthasone'] (6 tokens)
CamemBERT: ['__Test', '__de', '__freinage', '__à', '__la', '__de', 'x', 'a', 'méth', 'as', 'one'] (Δ+5)
DrBERT: ['__Test', '__de', '__frein', 'age', '__à', '__la', '__dexaméthasone'] (Δ+1)

Entity: ['drépanocytose'] (1 word)
TransBERT: ['__d', 'ré', 'pan', 'ocytose'] (4 tokens)
CamemBERT: ['__d', 'ré', 'pan', 'oc', 'y', 'tos', 'e'] (Δ+3)
DrBERT: ['__drépanocytose'] (Δ-3)

Entity: ['Enquête', 'séro-immunologique'] (2 words)
TransBERT: ['__Enquête', '__séro', '-', 'immunologique'] (4 tokens)
CamemBERT: ['__Enquête', '__s', 'éro', '-', 'imm', 'un', 'ologique'] (Δ+3)
DrBERT: ['__Enquête', '__séro', '-', 'immun', 'ologique'] (Δ+1)

Entity: ['mégalérythème', 'infectieux'] (2 words)
TransBERT: ['__méga', 'l', 'érythème', '__infectieux'] (4 tokens)
CamemBERT: ['__m', 'égal', 'éry', 'thème', '__', 'infect', 'ieux'] (Δ+3)
DrBERT: ['__mé', 'gal', 'éryth', 'ème', '__infectieux'] (Δ+1)

Entity: ['oesophage', 'embryonnaire'] (2 words)
TransBERT: ['__o', 'es', 'ophage', '__embryonnaire'] (4 tokens)
CamemBERT: ['__o', 'es', 'oph', 'age', '__', 'embryon', 'naire'] (Δ+3)
DrBERT: ['__oes', 'ophage', '__embryonnaire'] (Δ-1)

Entity: ['inclusions', 'érythrocytaires'] (2 words)
TransBERT: ['__inclusions', '__', 'érythrocyt', 'aires'] (4 tokens)
CamemBERT: ['__inclus', 'ions', '__', 'éry', 'thro', 'cy', 'taires'] (Δ+3)
DrBERT: ['__inclusions', '__érythrocyt', 'aires'] (Δ-1)

Entity: ['problèmes', 'urinaires', 'et', 'intestinaux'] (4 words)
TransBERT: ['__problèmes', '__urinaires', '__et', '__intestinaux'] (4 tokens)
CamemBERT: ['__problèmes', '__urinaire', 's', '__et', '__', 'intestin', 'aux'] (Δ+3)
DrBERT: ['__problèmes', '__urinaires', '__et', '__intestinaux'] (Δ0)

Entity: ['Dysplasie', 'vasculaire', 'complexe'] (3 words)
TransBERT: ['__Dys', 'plasie', '__vasculaire', '__complexe'] (4 tokens)
CamemBERT: ['__D', 'y', 's', 'pla', 'sie', '__vasculaire', '__complexe'] (Δ+3)
DrBERT: ['__Dys', 'plasie', '__vasculaire', '__complexe'] (Δ0)

Entity: ['maladie', 'thromboembolique', 'veineuse'] (3 words)
TransBERT: ['__maladie', '__', 'thromboembolique', '__veineuse'] (4 tokens)
CamemBERT: ['__maladie', '__thrombo', 'e', 'mbo', 'lique', '__veine', 'use'] (Δ+3)
DrBERT: ['__maladie', '__thrombo', 'embolique', '__veineuse'] (Δ0)

Entity: ['glomérulaire', '"', 'dans', 'le', 'rein'] (5 words)
TransBERT: ['__glomérulaire', '__"', '__dans', '__le', '__rein'] (5 tokens)
CamemBERT: ['__g', 'lom', 'ér', 'ulaire', '__"', '__dans', '__le', '__re', 'in'] (Δ+4)
DrBERT: ['__glomérulaire', '__', '¡unk¿', '__dans', '__le', '__rein'] (Δ+1)

Ratio: 1.7 (168 entities)

Entity: ['sensibilité', 'aux', 'mutagènes'] (3 words)
TransBERT: ['__sensibilité', '__aux', '__mutagènes'] (3 tokens)
CamemBERT: ['__sensibilité', '__aux', '__mu', 'ta', 'gènes'] (Δ+2)
DrBERT: ['__sensibilité', '__aux', '__mutag', 'ènes'] (Δ+1)

Entity: ['erythrocytaire'] (1 word)
TransBERT: ['__erythr', 'o', 'cytaire'] (3 tokens)
CamemBERT: ['__', 'ery', 'thro', 'cy', 'taire'] (Δ+2)
DrBERT: ['__eryth', 'rocyt', 'aire'] (Δ0)

Entity: ['protéine', 'Hsp70'] (2 words)
TransBERT: ['__protéine', '__Hsp', '70'] (3 tokens)
CamemBERT: ['__protéine', '__H', 's', 'p', '70'] (Δ+2)
DrBERT: ['__protéine', '__Hsp', '70'] (Δ0)

Entity: ['tête', 'du', 'pancréas'] (3 words)
TransBERT: ['__tête', '__du', '__pancréas'] (3 tokens)
CamemBERT: ['__tête', '__du', '__pan', 'cré', 'as'] (Δ+2)
DrBERT: ['__tête', '__du', '__pancréas'] (Δ0)

Entity: ['scolioses', 'graves'] (2 words)
TransBERT: ['__scoliose', 's', '__graves'] (3 tokens)
CamemBERT: ['__s', 'col', 'ios', 'es', '__graves'] (Δ+2)
DrBERT: ['__scoli', 'oses', '__graves'] (Δ0)

Entity: ['tractus', 'génital', 'femelle'] (3 words)
TransBERT: ['__tractus', '__génital', '__femelle'] (3 tokens)
CamemBERT: ['__tract', 'us', '__', 'génital', '__femelle'] (Δ+2)
DrBERT: ['__tractus', '__génital', '__femelle'] (Δ0)

Entity: ['Hypophosphatasie', 'congénitale'] (2 words)
TransBERT: ['__Hypo', 'phosph', 'ata', 's', 'ie', '__congénitale'] (6 tokens)
CamemBERT: ['__Hy', 'po', 'pho', 's', 'pha', 'tas', 'ie', '__con', 'génital', 'e'] (Δ+4)
DrBERT: ['__Hyp', 'ophosph', 'at', 'asie', '__congénitale'] (Δ-1)

Entity: ['ostéomalacies'] (1 word)
TransBERT: ['__ostéo', 'malacie', 's'] (3 tokens)
CamemBERT: ['__os', 'té', 'oma', 'lac', 'ies'] (Δ+2)
DrBERT: ['__osté', 'omal', 'ac', 'ies'] (Δ+1)

Entity: ['formation', 'lipomateuse'] (2 words)
TransBERT: ['__formation', '__lipo', 'mateuse'] (3 tokens)
CamemBERT: ['__formation', '__l', 'ip', 'omat', 'euse'] (Δ+2)
DrBERT: ['__formation', '__lip', 'omateuse'] (Δ0)

Entity: ['cholangio-wirsungographie', 'endoscopique'] (2 words)
TransBERT: ['__cholangio', '-', 'wi', 'rs', 'ung', 'ographie', '__endoscopique'] (7 tokens)
CamemBERT: ['__cho', 'lang', 'io', '-', 'wi', 'r', 's', 'ung', 'ographie', '__en', 'do', 'scopique'] (Δ+5)
DrBERT: ['__chol', 'angio', '-', 'w', 'irs', 'ung', 'ographie', '__endoscopique'] (Δ+1)

Ratio: 1.6 (60 entities)

Entity: ['staphylococcus', 'aureus'] (2 words)
TransBERT: ['__', 'sta', 'phyl', 'ococcus', '__aureus'] (5 tokens)
CamemBERT: ['__sta', 'phyl', 'oco', 'c', 'cus', '__au', 're', 'us'] (Δ+3)
DrBERT: ['__staphyl', 'ococcus', '__aureus'] (Δ-2)

Entity: ['barrière', 'hémato-encéphalique'] (2 words)
TransBERT: ['__barrière', '__hémato', '-', 'en', 'céphalique'] (5 tokens)
CamemBERT: ['__barrière', '__', 'hémat', 'o', '-', 'enc', 'épha', 'lique'] (Δ+3)
DrBERT: ['__barrière', '__hémato', '-', 'encéphalique'] (Δ-1)

Entity: ['infiltrat', 'polymorphe', 'de', 'cellules'] (4 words)
TransBERT: ['__', 'infiltrat', '__polymorphe', '__de', '__cellules'] (5 tokens)
CamemBERT: ['__', 'infiltr', 'at', '__poly', 'morph', 'e', '__de', '__cellules'] (Δ+3)
DrBERT: ['__infiltra', 't', '__polymorphe', '__de', '__cellules'] (Δ0)

Entity: ['veine', 'porte', 'pré-duodénale'] (3 words)
TransBERT: ['__veine', '__porte', '__pré', '-', 'duodénale'] (5 tokens)
CamemBERT: ['__veine', '__porte', '__pré', '-', 'du', 'o', 'dé', 'nale'] (Δ+3)
DrBERT: ['__veine', '__porte', '__pré', '-', 'duodénale'] (Δ0)

Entity: ['Endométriome', 'ovarien'] (2 words)
TransBERT: ['__Endo', 'mé', 'tri', 'ome', '__ovarien'] (5 tokens)
CamemBERT: ['__En', 'dom', 'é', 'tri', 'ome', '__', 'ova', 'rien'] (Δ+3)
DrBERT: ['__En', 'dom', 'ét', 'ri', 'ome', '__ovarien'] (Δ+1)

Entity: ['Cholestases', 'intrahépatiques'] (2 words)
TransBERT: ['__Chol', 'est', 'ases', '__intra', 'hépatiques'] (5 tokens)
CamemBERT: ['__Cho', 'les', 'tas', 'es', '__intra', 'hé', 'pa', 'tiques'] (Δ+3)
DrBERT: ['__Chol', 'est', 'ases', '__intra', 'hép', 'atiques'] (Δ+1)

Entity: ['crise', 'd', "'éclampsie"] (3 words)
TransBERT: ['__crise', '__d', '__', '"', 'éclampsie'] (5 tokens)
CamemBERT: ['__crise', '__d', '__', '"', 'é', 'cl', 'amp', 'sie'] (Δ+3)
DrBERT: ['__crise', '__d', "__", 'éclampsie'] (Δ-1)

Entity: ['spectre', "d'", 'amplitudes', 'de', "l'", 'électroencéphalogramme'] (6 words)
TransBERT: ['__spectre', '__d', "'", '__amplitudes', '__de', '__l', "'", '__électro', 'encéphalogramme'] (9 tokens)
CamemBERT: ['__spectre', '__d', "'", '__', 'amplitude', 's', '__de', '__l', "'", '__électro', 'enc', 'épha', 'l', 'ogramme'] (Δ+5)
DrBERT: ['__spectre', '__d', "'", '__amplitudes', '__de', '__l', "'", '__électro', 'encéphal', 'ogramme'] (Δ+1)

Entity: ['lamyotrophie', 'spinale', 'infantile'] (3 words)
TransBERT: ['__la', 'myo', 'trophie', '__spinale', '__infantile'] (5 tokens)
CamemBERT: ['__la', 'my', 'o', 'trophi', 'e', '__spin', 'ale', '__infantile'] (Δ+3)
DrBERT: ['__lam', 'yotroph', 'ie', '__sp', 'inale', '__infantile'] (Δ+1)

Entity: ['5', '-', 'hydroxytryptamine'] (3 words)
TransBERT: ['__5', '__-', '__', 'hydroxytryptamin', 'e'] (5 tokens)
CamemBERT: ['__5', '__-', '__hydro', 'x', 'y', 'try', 'pt', 'amine'] (Δ+3)
DrBERT: ['__5', '__-', '__hydroxy', 'tr', 'ypt', 'amine'] (Δ+1)

Ratio: 1.5 (511 entities)

Entity: ['discarthrose'] (1 word)
TransBERT: ['__disc', 'arthrose'] (2 tokens)
CamemBERT: ['__dis', 'c', 'arthrose'] (Δ+1)
DrBERT: ['__disc', 'arthrose'] (Δ0)

Entity: ['cliniques', 'vétérinaires'] (2 words)
TransBERT: ['__cliniques', '__vétérinaires'] (2 tokens)
CamemBERT: ['__cliniques', '__vétérinaire', 's'] (Δ+1)
DrBERT: ['__cliniques', '__vétérinaires'] (Δ0)

Entity: ['filarienne'] (1 word)
TransBERT: ['__fil', 'arienne'] (2 tokens)
CamemBERT: ['__fil', 'a', 'rienne'] (Δ+1)
DrBERT: ['__fil', 'arienne'] (Δ0)

Entity: ['ganglionnaire'] (1 word)
TransBERT: ['__ganglion', 'naire'] (2 tokens)
CamemBERT: ['__gang', 'lion', 'naire'] (Δ+1)
DrBERT: ['__ganglionnaire'] (Δ-1)

Entity: ['symptômes', 'cognitifs'] (2 words)
TransBERT: ['__symptômes', '__cognitifs'] (2 tokens)
CamemBERT: ['__symptômes', '__cognitif', 's'] (Δ+1)
DrBERT: ['__symptômes', '__cognitifs'] (Δ0)

Entity: ['infestation'] (1 word)
TransBERT: ['__', 'infestation'] (2 tokens)
CamemBERT: ['__inf', 'est', 'ation'] (Δ+1)
DrBERT: ['__inf', 'estation'] (Δ0)

Entity: ['pharmacologiques'] (1 word)
TransBERT: ['__pharmacologique', 's'] (2 tokens)
CamemBERT: ['__pharmaco', 'logique', 's'] (Δ+1)
DrBERT: ['__pharmac', 'ologiques'] (Δ0)

Entity: ['bloqueur', 'adrénergique', 'bèta'] (3 words)
TransBERT: ['▁bloqueur', '▁', 'adrénergique', '▁b', 'è', 'ta'] (6 tokens)
CamemBERT: ['▁bloque', 'ur', '▁ad', 'ré', 'ner', 'gique', '▁b', 'è', 'ta'] (Δ+3)
DrBERT: ['▁bloque', 'ur', '▁ad', 'rénergique', '▁b', 'è', 'ta'] (Δ+1)

Entity: ['équilibre', 'acido-basique'] (2 words)
TransBERT: ['▁équilibre', '▁acido', '-', 'basique'] (4 tokens)
CamemBERT: ['▁équilibre', '▁ac', 'ido', '-', 'bas', 'ique'] (Δ+2)
DrBERT: ['▁équilibre', '▁acid', 'o', '-', 'b', 'asique'] (Δ+2)

Entity: ['oestrus'] (1 word)
TransBERT: ['▁o', 'estrus'] (2 tokens)
CamemBERT: ['▁o', 'est', 'rus'] (Δ+1)
DrBERT: ['▁oest', 'rus'] (Δ0)

Appendix J

CamemBERT Vs cTransBERT: Results aggregated by task

	CamemBERT			cTransBERT		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	74.65**	75.54	74.17**	**75.10****	**76.05**	**74.70****
Macro Avg	57.74**	56.94	55.66**	**60.58****	**61.08**	**59.31****
🥇 / 🥈	7/73	6/101	5/83	**10/121**	**13/110**	**12/110**
🥉 / 🏅	**10/95**	**11/67**	**12/85**	7/47	4/58	5/58
NRA	36.22**	45.68	40.81**	**63.78****	**54.32**	**59.19****

Table J.1: Model Evaluation for the Classification Task (Tokenizer Analysis) - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different classes/labels for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Wilcoxon test. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

	CamemBERT			cTransBERT		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	**81.23**	82.13	**81.55**	81.02	**82.13**	81.44
Macro Avg	66.23	66.45	65.60	**67.16**	**67.13**	**66.51**
🥇 / 🥈	**27**/140	**27**/170	**27**/132	19/**174**	**27**/**190**	19/**177**
🥉 / 🏅	19/**129**	**19**/**99**	19/**137**	**27**/95	**19**/79	**27**/92
NRA	45.87	46.83	44.13	**54.13**	**53.17**	**55.87**

Table J.2: Model Evaluation for the Named Entity Recognition Task (Tokenizer Analysis) - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different entities for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Wilcoxon test. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

	CamemBERT			cTransBERT		
	P	***R***	***F*$_1$**	***P***	***R***	***F*$_1$**
Weighted Avg	**98.31**	**98.29**	**98.29**	98.31	98.29	98.29
Macro Avg	91.43	89.17	89.89	**92.29**	**89.43**	**90.23**
🥇 / ⚪	**13/187**	**14/207**	**12/187**	7/179	7/193	8/166
🥈 / ⚪	7/88	6/68	8/88	**13/96**	**13/82**	**12/109**
NRA	**52.37**	**53.56**	**54.24**	47.63	46.44	45.76

Table J.3: Model Evaluation for the Part-of-Speech Tagging Task (Tokenizer Analysis) - This table presents the weighted and macro aggregations for Precision, Recall, and F_1 across different tags for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Wilcoxon test. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

	CamemBERT	cTransBERT
	***R*^{2}**	***R*^{2}**
Weighted Avg	83.38	**84.36**
Macro Avg	83.38	**84.36**
🥇 / ⚪	1/2	**3/4**
🥈 / ⚪	**3/4**	1/2
NRA	30.00	**70.00**

Table J.4: Model Evaluation for the Semantic Textual Similarity Task (Tokenizer Analysis) - This table presents the weighted and macro aggregations for R^2 for each dataset and fold. It also illustrates the ranking distribution through the medals system and the Normalized Ranking Average, whose statistical significance for difference has been evaluated using the Wilcoxon test. (*) and (**) indicate statistical significance at $\alpha = 0.05$ and $\alpha = 0.01$, respectively.