AVIGNON UNIVERSITÉ

THÈSE DE DOCTORAT D'AVIGNON UNIVERSITÉ

École Doctorale N° 536
Sciences et agrosciences

Spécialité / Discipline de doctorat :
Informatique

Laboratoire Informatique d'Avignon

Présentée par
Yanis Labrak

Language Models at the Crossroads of Text and Speech for Healthcare Applications

---

Soutenue publiquement le 15/09/2025 devant le jury composé de :

Pierre Zweigenbaum, Directeur de recherche, LISN, Université Paris-Saclay	**Rapporteur**
Philippe Langlais, Professor, DIRO, Université de Montréal	**Rapporteur**
Elena V. Epure, Senior Research Scientist, Deezer Research	**Examinatrice**
Laurent Besacier, Principal Scientist, Naver Labs Europe	**Examinateur**
Asma Ben Abacha, Senior Scientist, Microsoft Health AI	**Examinatrice**
Mickaël Rouvier, Maître de conférence - HDR, LIA - Université d'Avignon	**Co-directeur**
Richard Dufour, Professeur, LS2N - Université de Nantes	**Co-directeur**
Julien Nave, R&D Director, Zenidoc	**Invité**

LABORATOIRE INFORMATIQUE D'AVIGNON

Abstract

The medical field presents unique *Natural Language Processing* (NLP) challenges through its specialized terminology, strict data regulations, and critical information needs. With the democratization of *Language Models* (LM) for assisting healthcare and clinical workers in their day-to-day work, the need for their adaptation to the domains of application became necessary to facilitate their accessibility to a broader audience, languages, and domains while reducing the computational cost of their usage.

On the other hand, traditional approaches to medical speech processing rely on cascade systems that convert speech to text, apply NLP system, and sometimes regenerate speech. While practical, these systems often lose paralinguistic features critical to clinical communication and suffer from error propagation between processing stages. Recent advances in self-supervised speech representation quantization have created new possibilities for integrating speech representation into other systems without intermediate text conversion, potentially preserving more communicative nuance.

In this thesis, I investigate among other things, how speech capabilities can be integrated into existing text-based *Pre-trained Language Models* (PLM) with healthcare-related capabilities, leveraging their embedded medical knowledge while enabling direct speech processing. The examination of alignment between speech and text representations at various abstraction levels reveals potential pathways for effective cross-modal knowledge transfer with limited training data, a crucial consideration given healthcare's data constraints.

Keywords: Speech Processing, Domain Adaptation, Cross-Modal Transfer, Healthcare Adaptation, Language Models, Multi-modal Speech-Text Modeling, Large Language Model (LLM)

Résumé

Le domaine médical présente des défis uniques en matière de *Traitement Automatique de la Langue Naturelle* (TALN) à travers sa terminologie spécialisée, ses réglementations strictes sur les données et ses besoins critiques en information. Avec la démocratisation des *Modèles de Langues* (ML) pour assister les professionnels de santé dans leur quotidien, leur adaptation aux domaines d'application est devenue nécessaire pour faciliter leur accessibilité à un public plus large, à différentes langues et domaines, tout en réduisant le coût computationnel de leur utilisation.

D'autre part, les approches traditionnelles du traitement de la parole médicale reposent sur des systèmes en cascade qui convertissent la parole en texte, appliquent un système de TALN, et parfois régénèrent la parole. Bien que pratiques, ces systèmes perdent souvent des caractéristiques paralinguistiques essentielles à la communication clinique et souffrent de la propagation d'erreurs entre les étapes de traitement. Les récentes avancées dans la quantification des représentations vocales auto-supervisées ont créé de nouvelles possibilités d'intégration de la représentation vocale dans d'autres systèmes sans conversion intermédiaire en texte, préservant potentiellement plus de nuances communicatives.

Dans cette thèse, j'examine comment les capacités vocales peuvent être intégrées aux *Modèles de Langue Pré-entraînés* (MLP) basés sur le texte et possédant des connaissances liées aux domaines de la santé, en exploitant leurs connaissances médicales acquises tout en permettant un traitement direct de la parole, sans étapes intermédiaires. l'analyse des capacitées d'alignement entre les représentations vocales et textuelles à différents niveaux d'abstraction ont révélé des méthodes plus optimales pour un transfert efficace de connaissances intermodales et savorisant ainsi l'apprentissage contraint par une quantité de données d'entraînement limitées, une considération cruciale étant donné les contraintes de données dans le domaine de la santé.

Mots-clés: Traitement de la Parole, Adaptation au Domaine, Transfert Intermodal, Adaptation aux domaines de la santé, Modèles de Langue, Modélisation Multimodale Parole-Texte, Grand Modèle de Langage (GLM)

Acknowledgement

I wish to express my sincere gratitude to Pierre Zweigenbaum and Philippe Langlais for generously accepting to review this manuscript as reporters. I am also deeply thankful to Elena V. Epure, Laurent Besacier, and Asma Ben Abacha for participating in my Ph.D. committee as examinators. Their expertise and time dedicated to evaluating this work are invaluable, and I am honored to have such distinguished researchers in my thesis committee.

I wish to thank my Ph.D. supervisors, Mickaël Rouvier and Richard Dufour, for their guidance, expertise, and support throughout this journey. Their mentorship has been invaluable in shaping my research path and academic growth. I am particularly grateful for their unwavering support during challenging times, their ability to help me navigate through difficulties, and their constant encouragement that helped me overcome obstacles and emerge stronger. Their patience and understanding went beyond academic supervision, making this journey not just academically enriching but also personally rewarding.

I am deeply grateful to the talented collaborators I had the privilege to work with during these years. Special thanks to Adrien Bazoge, Adel Moumen, Santiago Cuervo, Ricard Marxer, and Pierre Antoine Gourraud for their insightful collaborations and the knowledge we shared together.

My Ph.D. journey was enriched by wonderful travel experiences shared with amazing colleagues. I want to thank Adrien Bazoge, Arthur Amalvy, François Remy, Adel Moumen, and Oumaima El Khettari for the unforgettable moments we shared in Abu Dhabi, Toronto, Bangkok, Kos Island, and Paris.

I am thankful for my internship experience at Deezer under the guidance of Elena V. Epure and Gabriel Meseguer-Brocal. I want to extend my gratitude to the entire research team who welcomed me warmly: Bruno, Darick, Dorian, Gaspard, Guillaume, Karl, Lilian, Manuel, Marion, Romain, Viet Anh, and Yuexuan.

I would also like to express my appreciation to Julien Nave and Francis Tibermont from Zenidoc for their supervision and the freedom they granted me to explore and grow professionally during my time there.

I am grateful to all the colleagues and friends I met at the LIA and LS2N laboratories, who contributed to creating an enriching and collaborative research environment.

Je tiens à remercier particulièrement mes parents, Stéphanie et Saïd, pour leur soutien indéfectible et leurs encouragements constants tout au long de ce parcours. Merci également à toute ma famille et mes amis qui m'ont accompagné durant cette aventure.

Contents

Introduction

0.1 Motivations and Research Question

Healthcare natural language understanding faces unique challenges stemming from strict privacy requirements, limited data availability, and prohibitive annotation expenses. These barriers significantly restrict the development of accessible open-source solutions. Nevertheless, text and speech processing technologies hold tremendous potential for healthcare applications, from supporting clinical staff in daily activities to enhancing hospital revenue systems and enabling researchers to process vast amounts of unstructured data in the pursuit of novel treatments for both common and rare conditions.

Contemporary approaches in this domain predominantly leverage machine learning paradigms, utilizing LM either for semantic representation through vectors or direct interaction via autoregressive architectures such as *Generative Pre-trained Transformer* (GPT) models. This thesis investigates several pivotal questions at the intersection of LM, speech, and healthcare. We examine optimal strategies for encoding medical knowledge in LM for downstream application, comparing the efficacy of continual pre-training versus from-scratch approaches for adapting to resource-constrained domains like French medical language. My research also explores whether publicly accessible data can achieve comparable performance to private clinical datasets and challenges the assumption that more data invariably produces better outcomes. The thesis also addresses a fundamental tension, whether domain specialization compromises general language understanding.

Furthermore, we investigate how domain-specific tokenization affects both performance metrics and practical considerations like information density and computational efficiency, factors that ultimately determine accessibility across different languages and specialized fields. Through systematic comparative evaluation, this research identifies which approaches excel in specific contexts and where they fall short. The resource-intensive nature of healthcare data annotation prompted us to compare instruction-tuned *Large Language Models* (LLM), which were on the premises, with traditionally task-specific fine-tuned *Masked Language Models* (MLM). My findings reveal that LLMs demonstrate remarkable generalization to unfamiliar tasks, occasionally surpassing specialized models in areas like question answering. These insights informed that our adaptation methodology through continual pre-training on PubMed Central's open-access corpus, strategically leveraging the multilingual foundation of the model to facilitate knowledge transfer to medical applications and therefore French.

Recognizing the advancing capability of pre-trained LLMs to generalize across specialized domains through diverse training and reinforcement learning techniques, we expanded our research to incorporate speech modality into state-of-the-art LM, which showed good capabilities spoken tasks. This multimodal integration raises compelling questions about specialized processing requirements for speech, scaling benefits across data and model parameters, knowledge transfer between modalities, identifying which neural components contribute most significantly to speech comprehension, and preserving textual capabilities

while acquiring speech understanding.

0.2 Thesis Structure

This thesis is written around eight key articles, each chapters from Chapter 1 to 8 represent one article, they are structured into three main parts following the temporality of the thesis and showing the incremental efforts put in place to build language models adequate to process speech with healthcare related capabilities: the first one dedicated on the adaptation of MLM on the healthcare domains and their deep quantitative and qualitative analysis (Part II). The second part (Part III) presenting MLM limitations and how the adaptation of LLM can allow to go therefore those limitations. Finally, Part IV, about our how do we managed to align pre-trained LLMs with speech modality in order to leverage widely available textual knowledge.

0.3 Ressources

Personal publications

- Santiago Cuervo, Adel Moumen, **Yanis Labrak**, Sameer Khurana, Antoine Laurent, Mickael Rouvier, Ricard Marxer (2025)
  Text-Speech Language Models with Improved Cross-Modal Transfer by Aligning Abstraction Levels. - https://arxiv.org/abs/2503.06211
- **Yanis Labrak**, Richard Dufour, Mickael Rouvier (2025)
  TSD 2025, August 2025, Erlangen, Germany - *An Empirical Analysis of Discrete Unit Representations in Speech Language Modeling Pre-training: From Encoders to Phonemic Relations.* - https://drive.google.com/file/d/1JzZmWreVwZ2CTfPedvC_0T6xlkmp_HjL
- **Yanis Labrak**, Adel Moumen, Mickael Rouvier and Richard Dufour (2024).
  InterSpeech 2024, September 2024, Kos Island, Greece - *Zero-Shot End-To-End Spoken Question Answering In Medical Domain.* - https://www.isca-archive.org/interspeech_2024/labrak24_interspeech.html
- **Yanis Labrak**, Adrien Bazoge, Richard Dufour, Mickael Rouvier, Emmanuel Morin, et al. (2024).
  ACL 2024, July 2024, Bangkok, Thailand - *BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains.* - https://aclanthology.org/2024.findings-acl.348/
- **Yanis Labrak**, Mickael Rouvier and Richard Dufour. (2024)
  LREC-COLING 2024, May 2024, Turino, Italy - *A Zero-shot and Few-shot Study of Instruction-Finetuned Large Language Models Applied to Clinical and Biomedical Tasks.* - https://aclanthology.org/2024.lrec-main.185/
- **Yanis Labrak**, Adrien Bazoge, Béatrice Daille, Mickael Rouvier and Richard Dufour. (2024)
  LREC-COLING 2024, May 2024, Turino, Italy - *How Important Is Tokenization in French Medical Masked Language Models?* - https://aclanthology.org/2024.lrec-main.721/
- **Yanis Labrak**, Adrien Bazoge, Oumaima El Khettari, Mickael Rouvier, Pacome constant dit beaufils, Natalia Grabar, Béatrice Daille, Solen Quiniou, Emmanuel Morin, Pierre-Antoine Gourraud and Richard Dufour. (2024)
  LREC-COLING 2024, May 2024, Turino, Italy - *DrBenchmark: A Large Language Understanding Evaluation Benchmark for French Biomedical Domain.* - https://aclanthology.org/2024.lrec-main.478/

- **Yanis Labrak**, Adrien Bazoge, Richard Dufour, Mickael Rouvier, Emmanuel Morin, et al. (2023). **ACL 2023** - July 2023, Toronto, Canada - *DrBERT: A Robust Pre-trained Model in French for Biomedical and Clinical domains.* - https://aclanthology.org/2023.acl-long.896/
- **Yanis Labrak**, Adrien Bazoge, Richard Dufour, Mickael Rouvier, Emmanuel Morin, Béatrice Daille, Pierre-Antoine Gourraud (2023). **LOUHI @ EMNLP 2022** - December 2022, Abu Dhabi, United Arab Emirates - *FrenchMedMCQA: A French multiple-choice question answering dataset for medical domain.* - https://aclanthology.org/2022.louhi-1.5/
- Jason Fries, Leon Weber, Natasha Seelam, Gabriel Altay, Debajyoti Datta, **Yanis Labrak** (2022). **NeurIPS 2022** - December 2022, Abu Dhabi, United Arab Emirates - *Bigbio: A framework for data-centric biomedical natural language processing.* - https://dl.acm.org/doi/10.5555/3600270.3602140
- Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, **Yanis Labrak** et al. (2023). *Bloom: A 176b-parameter open-access multilingual language model.* - https://arxiv.org/abs/2211.05100

Research Artifacts

Models & Code

- **DrBERT** – Collection of French biomedical masked language model `https://huggingface.co/Dr-BERT`
- **BioMistral** – Collection of 7B LLM models adapted for healthcare domain `https://huggingface.co/BioMistral`
- **BioMedTok** – Collection of 17 tokenizers and models for biomedical domain `https://huggingface.co/BioMedTok`
- **SmolTolk** – Collection of LLM extended with speech capabilities (150M, 400M and 2B parameters) `https://huggingface.co/ParoleLM`

Datasets

- **NACHOS** – Large french crawled medical corpus `https://huggingface.co/datasets/Dr-BERT/NACHOS`
- **DrBenchmark** – Collection of 12 medical datasets and 20 tasks `https://huggingface.co/DrBenchmark/datasets`
- **SpokenMedicalQA** – Benchmark for medical spoken QA evaluation `https://huggingface.co/datasets/SpokenMedicalQA/SpokenMedicalQA`

Glossary

TALN	Traitement Automatique de la Langue Naturelle
NLP	Natural Language Processing
LM	Language Models
ML	Modèles de Langues
MLP	Modèles de Langue Pré-entraînés
PLM	Pre-trained Language Models
LLM	Large Language Models
MLM	Masked Language Models
CoT	Chain-of-Thought
BoW	bag-of-words
TF-IDF	Term Frequency-Inverse Document Frequency
GRUs	Gated Recurrent Units
LSTM	Long Short-Term Memory
GPT	Generative Pre-trained Transformer
NER	Named Entity Recognition
POS	Part-of-Speech
RE	Relation extraction
QA	question-answering
ICO	Intervention-Comparator-Outcome
NLI	Natural Language Inference
EMR	Exact Match Rate
EDRM	Euclidean Distance-based Relative Metric
SSL	Self-Supervised Learning
ASR	Automatic Speech Recognition
CNN	Convolutional Neural Network
RVQ	Residual Vector Quantization
STFT	Short-Time Fourier Transform
MFCC	Mel-Frequency Cepstral Coefficients
TTS	Text-To-Speech
WER	Word Error Rate
CER	Character Error Rate
SQA	Spoken Question Answering
MCQA	Multiple-Choice Question Answering
NLL	Negative Log Likelihood
NACHOS	opeN crAwled frenCh Healthcare cOrpuS
OCR	optical character recognition
BPE	byte-pair encoding
RAG	retrieval-augmented generation
TSLMs	Text-Speech Language Models
MLP	Multi-Layer Perceptron
NSP	Next Sentence Prediction
BERT	Bidirectional Encoder Representations from Transformers
SFT	Supervised Fine-Tuning
QLoRa	Quantized Low-Rank Adaptation
LoRA	Low-Rank Adaptation
AWQ	Activation-aware Weight Quantization
BnB	BitsandBytes
HPC	high-performance computing

Part I

Background and Related Work

0.4 Language Modeling

NLP has been transformed by large pre-trained language models based on the Transformer architecture [275]. These models, trained on massive text corpora and then fine-tuned for specific tasks, have become the standard approach in both general and specialized domains like medical NLP. Models such as GPT-4 [219] and Med-PaLM 2 [254] have demonstrated unprecedented performance, sometimes matching or exceeding human expert capabilities in medical tasks.

To understand the foundations of these powerful approaches, this chapter traces the evolution of text representation methods through four major paradigms: discrete representations that established basic computational text processing; statistical representations that introduced probabilistic approaches to word sequences; continuous but static word embeddings that enabled semantic relationships in dense vector spaces; and finally, current contextualized representations powered by transformer architectures.

NLP, whether in the general domain or in the medical field, develops and utilizes statistical methods, particularly machine learning or deep learning methods. The use of such methods requires a text representation adapted to these tools, generally in vector form. Various forms of vector representations of text have been proposed, ranging from discrete representations to numerical representations. The evolution of these statistical methods has been accompanied by an evolution in vector representations of words, with increasingly effective representations for incorporating word semantics.

0.4.1 Discrete Representations

Discrete representations form the foundation of early text processing methods in NLP. These representations encode text data into mathematical structures that computers can process, while maintaining the discrete nature of language units such as words or characters. Unlike continuous representations, discrete representations treat each word as a distinct, atomic unit, without an inherent notion of similarity or relationship between different words.

Bag-Of-Words

The most straightforward and historically significant approach to discrete text representation is the *bag-of-words* (BoW) model. This method, rooted in distributional semantics, as introduced by Zellig S. Harris [115], operates on a fundamental premise: the essential meaning within a text can be captured by considering word occurrences while disregarding grammatical structure and word order.

In its simplest form, a bag-of-words representation transforms a text into a vector whose dimension equals the size of the vocabulary. Each dimension corresponds to a specific word, and its value represents the frequency of that word in the text. Figure 1 illustrates this transformation process, where an input text is converted into a fixed-size vector based on a

predefined vocabulary. Note how words present in the vocabulary but absent from the text (such as "bird" and "park") receive zero frequency values, while frequent words like "the" have higher counts.

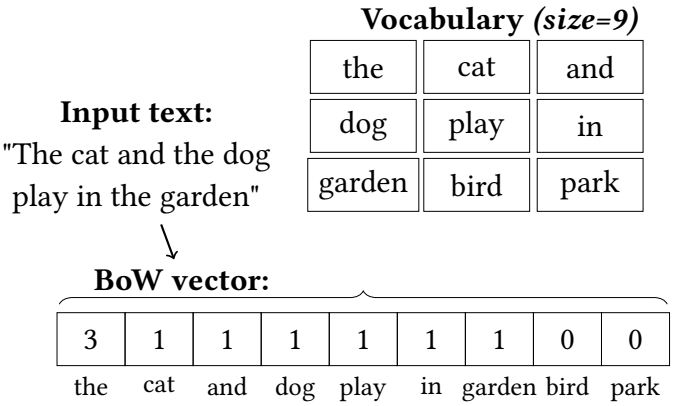


Figure 1: Illustration of the BoW representation. The input text is transformed into a fixed-size vector where each dimension corresponds to a word in the vocabulary, and the value represents the frequency of that word in the text. Note that words in the vocabulary that don't appear in the input text ("bird" and "park") have a frequency of 0.

Formally, let V denote the vocabulary size. Then, a text is represented as a V-dimensional vector T, where each component corresponds to a word in the vocabulary :

$$\vec{T} = (f_1, f_2, ..., f_{|V|})$$

where f_i represents the frequency of the i-th word in the vocabulary.

The BoW model can be extended beyond simple word frequencies to capture word co-occurrences within a context window. For a given word w, its representation considers surrounding words within a fixed window size f. Formally, a word c co-occurs with w if:

$$c \in [w_{-f}, w_{-1}] \cup [w_1, w_f]$$

This process results in a co-occurrence matrix $M \in \mathbb{R}^{|V|\times|V|}$, where each entry M_{ij} represents the number of times word i co-occurs with word j within the specified context window.

The BoW approach offers several advantages. Its implementation is straightforward and computationally efficient, making it well-suited for processing large text corpora. Moreover, as illustrated in Figure 1, the representations are inherently interpretable, with vector dimensions directly corresponding to vocabulary words.

Despite its advantages, the BoW approach faces several significant limitations. As illustrated in the figure, the resulting vectors are inherently sparse, with many dimensions containing zeros, particularly for rare words or those absent from the training text. This

sparsity issue is compounded by the high dimensionality of the representations, as the vocabulary size determines the vector dimensions and can reach hundreds of thousands in large-scale applications. The quality of word representations also varies considerably: while common words benefit from rich contextual information, leading to meaningful representations, rare words suffer from limited contextual data, resulting in less reliable vectors. Moreover, the fundamental premise of BoW, discarding word order and grammatical relationships, leads to a loss of important semantic nuances that could be crucial for understanding the text's meaning. For example, the vectors for the sentences *"The **vehicle** is **stationed** on the left."* and *"The **car** is **parked** on the left"* are far from each other despite having the same meaning.

Term Frequency-Inverse Document Frequency

While the basic BoW model captures word frequencies, it treats all words equally, regardless of their importance or discriminative power. The *Term Frequency-Inverse Document Frequency* (TF-IDF) weighting scheme addresses this limitation by balancing two factors: how frequently a term appears in a document (term frequency) and how unique that term is across the entire corpus (inverse document frequency).

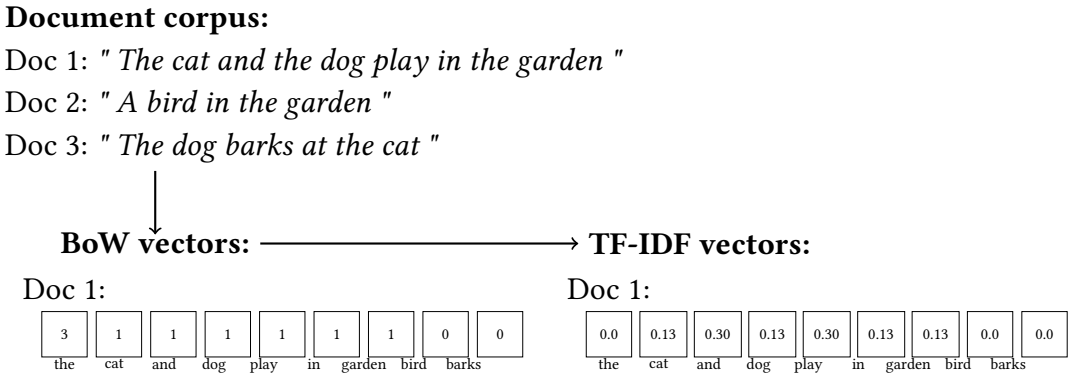


Figure 2: Illustration of TF-IDF transformation. The raw frequency counts from BoW are converted to weighted values that reflect term importance across the document corpus. Common words like "the" receive lower weights despite high frequency, while distinctive terms maintain higher importance.

Term Frequency (TF) measures how often a term occurs in a document, similar to the basic BoW approach:

$$\text{TF}(t, d) = \frac{\text{Number of times term } t \text{ appears in document } d}{\text{Total number of terms in document } d}$$

Inverse Document Frequency (IDF) penalizes terms that appear in many documents, as these are likely less informative, such as the stop words:

$$\text{IDF}(t, D) = \log \frac{\text{Total number of documents in corpus } D}{\text{Number of documents containing term } t}$$

The TF-IDF score for a term in a document is then calculated as:

$$\text{TF-IDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D)$$

This weighting scheme effectively reduces the importance of common words like "the" or "and" that appear in most documents while emphasizing rare, potentially more meaningful terms. Figure 2 demonstrates how TF-IDF transforms the raw frequency counts from the BoW model into weighted values that better reflect term importance.

The TF-IDF representation offers several advantages over the basic BoW model. It effectively reduces the impact of common, less informative words while emphasizing distinctive terms that better characterize document content. This weighting scheme has proven particularly effective for information retrieval and document classification tasks, where identifying discriminative features is crucial.

Despite these improvements, TF-IDF still inherits some limitations of the BoW approach. The representations remain sparse and high-dimensional, and the method continues to disregard word order and semantic relationships between terms. For instance, synonyms like "vehicle" and "car" are treated as entirely different dimensions despite their semantic similarity. Additionally, TF-IDF requires a predefined corpus to calculate the IDF component, making it unable to handle out-of-vocabulary terms.

These inherent limitations have spurred the development of more sophisticated word representation techniques, particularly continuous word embeddings. By projecting words into a fixed-dimensional dense vector space, these advanced methods effectively address the challenges of sparsity and high dimensionality while preserving or even enhancing the capture of semantic relationships between words. This evolution in representation techniques has provided more practical and efficient solutions for modern NLP applications.

0.4.2 Statistical Representations

While discrete representations like BoW provide a foundation for text processing, they lack the ability to model sequential patterns in language. Statistical representations, particularly n-gram language models, emerged as a way to capture local word dependencies and predict the probability of word sequences. These models build upon the distributional hypothesis while incorporating sequential information, making them particularly effective for tasks like speech recognition and machine translation.

N-gram Probabilistic Language Models

N-gram language models represent one of the most influential statistical approaches to language modeling. Unlike BoW, which treats words as independent units, n-gram models

like KenLM [119] consider sequences of n consecutive words to estimate the probability of the next word in a sequence. Formally, an n-gram model approximates the probability of a word sequence $W = (w_1, ..., w_m)$ as:

$$P(W) \approx \prod_{i=1}^{m} P(w_i | w_{i-n+1}, ..., w_{i-1})$$

where each word's probability depends on its $n - 1$ preceding words (its past context).

KenLM implements two efficient data structures for storing and querying these probabilities: PROBING and TRIE. The PROBING structure uses linear probing hash tables optimized for speed, while TRIE employs a trie with bit-level packing and interpolation search focused on memory efficiency. For a trigram model (n=3), probabilities are stored in the form:

$$\log P(w_i | w_{i-2}, w_{i-1})$$

A significant challenge with n-gram models is data sparsity: many possible n-gram sequences are never observed in training data, resulting in a sparse probability matrix. To address this issue, language models typically implement backoff mechanisms, which fall back to lower-order n-grams when a higher-order sequence is not observed. For example, if a specific trigram is not found in the training data, the model can back off to bigram or unigram probabilities using computed backoff weights. This technique allows the model to make reasonable probability estimates even for previously unseen sequences.

One advantage of n-gram models, including KenLM, is their ability to efficiently capture local word dependencies and idiomatic expressions while minimizing memory usage. For instance, in the phrase "New York City," the model learns that "City" is highly probable following "New York" and stores this information compactly. KenLM achieves this efficiency through different data structures: the PROBING structure allows fast hash-based lookups, while the TRIE structure reduces memory footprint using careful bit-packing and interpolation search.

However, like all n-gram models, these methods face inherent challenges. The sparsity problem remains significant: as n increases, the number of possible n-grams grows exponentially, making it impossible to observe all valid combinations in the training data. Additionally, n-gram models are constrained by their fixed context window size, limiting their ability to capture long-range dependencies or semantic relationships beyond their order.

Another major limitation is their inability to generalize beyond observed sequences. For example, if a model is trained on the phrase "the car's color is red" it will recognize this exact sequence but struggle to generalize to variations like "the car's color is magenta" This rigidity stems from the discrete nature of n-gram representations, which lack the ability to infer relationships between words beyond their explicit occurrence in the training data.

These constraints, particularly the difficulty in capturing semantic similarities and long-

range dependencies, led to the development of continuous representations, such as word embeddings and neural language models, which offer greater flexibility and generalization capabilities.

0.4.3 Continuous and Static Representations

Traditional n-gram language models, while effective for local patterns, struggle with data sparsity and discrete word representations. Word embedding methods were introduced to address these limitations. These methods map words into continuous vector spaces where each word is represented by a dense, real-valued vector of fixed dimension n. These vectors are learned on large amounts of data using neural network approaches, capturing semantic relationships more effectively than discrete vectors: semantically similar words will have similar vectors and will be close to each other in the representation space, where this would not be possible previously.

This fundamental shift toward a fixed dimension vector that allows a semantic comparison between words was, at the time of the release of Word2Vec in 2009, a game-changing approach that heavily influenced the future of modeling approaches of words and sequences.

Importantly, these word embeddings serve as crucial building blocks in modern neural architectures, where they are used to initialize the hidden layers of various neural network systems. Before the transformer architecture became dominant, word embeddings were generally pre-trained and therefore extensively employed in recurrent neural networks such as **LSTMs!** (**LSTMs!**) and *Gated Recurrent Units* (GRUs), bootstrapping these models with rich semantic representations that significantly improved their performance on various NLP tasks. This integration of pre-trained word embeddings into neural architectures established a fundamental paradigm that continues to influence how we represent and process language in deep learning systems.

Word2Vec embeddings: CBOW and Skip-gram architectures

Word2Vec, introduced by Mikolov et al. [205], revolutionized word representations by proposing two neural architectures for learning n-dimensional word embeddings: Continuous Bag-of-Words (CBOW) and Skip-gram. As illustrated in Figure 3, these architectures approach the learning task from opposite directions.

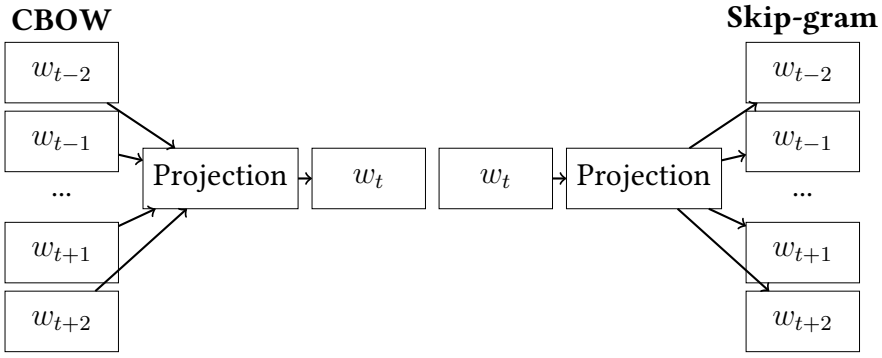


Figure 3: Word2Vec architectures: CBOW predicts the target word from context words, while Skip-gram predicts context words from the target word.

The CBOW architecture (left side of Figure 3) predicts a target word given its context. Initially, both the target word and its context are converted to one-hot vectors (vectors where all components are 0 except for one position which is 1). The input layer is the sum of the context's one-hot vectors, while the output layer is the target word's one-hot vector. The hidden layer forms the embedding layer where each vocabulary word is represented by a real-valued vector. The model is trained by comparing predicted and actual embeddings of the target word and adjusting the vector representations through backpropagation.

Conversely, the Skip-gram architecture (right side of Figure 3) attempts to predict the context words given a target word. However, with large vocabularies, this approach faces a computational challenge: for each positive pair (target word, context word), the model must generate numerous negative pairs (target word, vocabulary words not in the context) for training. To address this optimization problem, the authors introduced negative sampling, which stochastically samples only a subset of negative pairs for each positive example, significantly accelerating the training process.

GloVe embeddings

GloVe (Global Vectors for Word Representation) [229] combines the advantages of two approaches: statistical word co-occurrence methods and neural word embeddings. As shown in Figure 4, GloVe builds upon a word co-occurrence matrix and learns embeddings through matrix factorization.

The model first constructs a co-occurrence matrix from the text corpus, measuring how often each word pair appears within a given context window. From this matrix, word co-occurrence probabilities are calculated, representing the conditional probability of word j co-occurring with word i. The model is then trained by minimizing:

$$\sum_{i,j=1}^{V} f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log(X_{ij}))^2$$

where X_{ij} represents the co-occurrence counts and $f(X_{ij})$ is a weighting function.

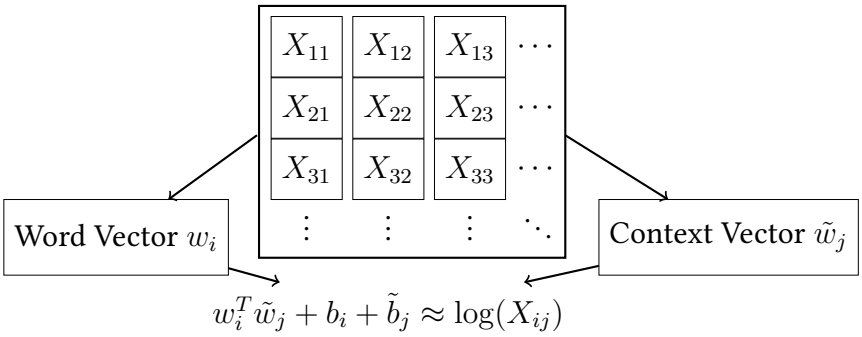


Figure 4: GloVe word vectors based on factorizing the word co-occurrence matrix.

FastText embeddings

Unlike Word2Vec and GloVe, which treat words as atomic units, FastText [33] extends the skip-gram model by representing each word as a bag of character n-grams, called subword units, as shown in Figure 5. Instead of learning a single vector per word, FastText learns representations for character n-grams (typically of length 3-6) and represents words as the sum of these n-gram vectors.

For example, the word "where" would be broken down into character n-grams: <wh, whe, her, ere, re> plus the special sequence <where>. The model adds special boundary tokens < and > to mark the beginning and end of words, helping distinguish between n-grams that appear in different positions. Like skip-gram, the model is trained to predict context words, but uses the sum of n-gram vectors instead of word vectors. The final word embedding is computed as:

$$s(w) = \sum_{g \in G_w} z_g$$

where G_w is the set of character n-grams in word w and z_g are the learned n-gram vectors.

This approach offers two key advantages: First, it can generate embeddings for out-of-vocabulary words by combining their character n-gram vectors. Second, it better captures morphological relationships between words, particularly beneficial for morphologically rich languages, as words sharing similar character sequences will have similar representations.

These static word embedding approaches marked a significant advancement in word representation. However, they share a common limitation: each word has a single fixed representation regardless of its context.

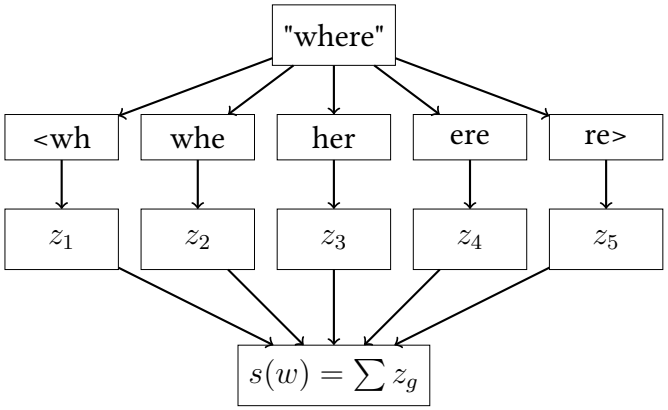


Figure 5: FastText word representation as the sum of its character n-gram vectors.

0.4.4 Continuous and Contextualized Representations

While static word embeddings represented a significant advancement in NLP, they face a fundamental limitation: each word has only one representation regardless of its context [90]. This limitation becomes particularly apparent when dealing with polysemous words or context-dependent meanings. For instance, the word "bank" in "river bank" versus "financial bank" should ideally have different representations reflecting their distinct meanings. Contextualized representations address this limitation by dynamically generating word representations based on their surrounding context [230].

The emergence of contextualized representations was made possible by several key developments: advanced tokenization methods, the transformer architecture [276], and novel self-supervised training objectives. These innovations collectively enabled the creation of powerful language models that could capture nuanced semantic relationships and generate context-aware word representations.

Tokenization Algorithms

Modern language models process text at a subword level rather than treating words as atomic units. This approach addresses the vocabulary size limitations and out-of-vocabulary problems faced by word-level models [249]. Three main tokenization algorithms have emerged as standards in the field: Byte-Pair Encoding (BPE) [249], WordPiece [295], and SentencePiece [158]. Figure 6 illustrates the general pipeline these algorithms follow, from raw text to final subword tokens.

c-e-p-h-a-l-o-g-r-a-p-h-y	
c-e-p-h-**al**-o-g-r-a-p-h-y	Merge: a+l → al
c-e-**ph**-al-o-g-r-a-p-h-y	Merge: p+h → ph
c-e-ph-al-o-**gr**-a-p-h-y	Merge: g+r → gr
c-e-ph-al-o-gr-a-**ph**-y	Merge: p+h → ph
ce-ph-al-o-gr-a-ph-y	Merge: c+e → ce
ce-**phal**-o-gr-a-ph-y	Merge: ph+al → phal
ce-phal-o-**graph**-y	Merge: gr+aph → graph
ce-phal-o-**graphy**	Merge: graph+y → graphy
cephal-o-graphy	Merge: ceph+al → cephal

Red: current merge

- : possible merge points

Figure 6: BPE segmentation process for the medical term "cephalography". Each step shows a merge operation (highlighted in red) based on the frequency of character pairs in the vocabulary. The process demonstrates how BPE can identify meaningful medical morphemes: "cephal" (head), "o" (combining vowel), and "graphy" (process of recording/imaging). Hyphens indicate possible merge points for the next iteration.

Byte-Pair Encoding, originally developed for data compression [95], iteratively merges the most frequent pairs of bytes or characters to build a vocabulary of subword units. As shown in Figure 6, the algorithm begins with character-level splitting and progressively applies merge operations based on frequency statistics from a training corpus. These merge operations combine frequent character sequences into larger subword units, creating a vocabulary that efficiently represents the training data.

WordPiece follows a similar pipeline but modifies the merging criteria. Instead of using pure frequency counting, it employs a likelihood-based approach for merge operations. This modification helps create more linguistically meaningful subword units, particularly useful for morphologically rich languages. The algorithm evaluates potential merges based on how much they would improve the likelihood of the training data given the current vocabulary.

Finally, SentencePiece implements a language-agnostic tokenization approach by treating input text as a sequence of Unicode characters. Unlike BPE and WordPiece, which typically operate on pre-tokenized text, SentencePiece applies its tokenization process directly to raw text. This makes it particularly suitable for languages without clear word boundaries, such as Chinese or Japanese, as it learns word segmentation and subword tokenization jointly.

Transformer Architecture

The Transformer architecture, introduced by Vaswani et al. [276], revolutionized NLP by replacing recurrent neural networks such as long short-term memory (LSTMs) [124] and gated recurrent neural networks [61] with self-attention mechanisms [18]. This mechanism enables parallel processing of input sequences and captures long-range dependencies more effectively than previous approaches.

At its core, the Transformer uses self-attention to compute representations of input tokens by considering their relationships with all other tokens in the sequence [276]. Each token's representation is computed as a weighted sum of all tokens' values, where the weights are determined by learned attention patterns. The multi-head attention mechanism allows the model to capture different types of relationships simultaneously, such as syntactic dependencies and semantic associations.

The architecture consists of multiple layers of self-attention and feed-forward neural networks, combined with residual connections [117] and layer normalization [12]. Position information is incorporated through learned positional encodings, allowing the model to understand token order despite its parallel processing nature.

Self-supervised Training Objectives

The effectiveness of contextualized representations largely depends on their training objectives. Self-supervised learning enables models to learn from vast amounts of unlabeled text by creating supervised learning tasks from the data itself [192]. Three primary training objectives have emerged: masked language modeling, autoregressive language modeling, and encoder-decoder language modeling, which we will explore in the following parts:

Encoder-only models: BERT and variants *Bidirectional Encoder Representations from Transformers* (BERT) [81] involves two tasks during pre-training: MLM and *Next Sentence Prediction* (NSP).

Masked language modeling, mainly popularized by BERT, consists of randomly masking tokens in the input sequence and training the model to predict these masked tokens, as shown in Figure 7. This objective forces the model to develop a deep understanding of bidirectional context and linguistic patterns. The masking strategy typically includes replacing tokens with a special `[MASK]` token in 15% of the cases, random tokens, or leaving them unchanged, helping the model learn robust representations [192].

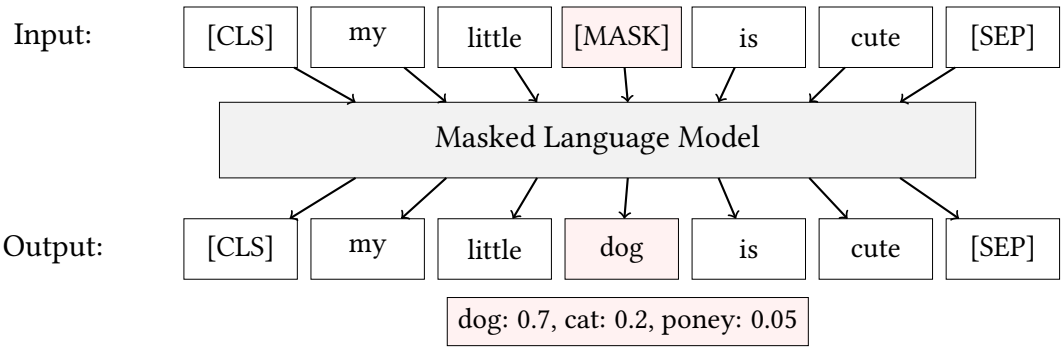


Figure 7: Masked Language Modeling (MLM) predicts masked tokens using bidirectional context.

The second training objective, NSP, consists of training the model to understand relationships between sentences by predicting whether two sentences appear consecutively

in the original text, using a special [CLS] token prepended to the input to capture this relationship.

The model architecture consists of multiple transformer encoder layers that process input text in both directions simultaneously. BERT's architecture comes in two variants: $\text{BERT}_{\text{BASE}}$ (12 layers, 12 attention heads, 768 hidden size, 110M parameters) and $\text{BERT}_{\text{LARGE}}$ (24 layers, 16 attention heads, 1024 hidden size, 340M parameters), both pre-trained on 3.2B words from Wikipedia and BooksCorpus [317].

BERT's input representation combines three embeddings as shown in the Figure 8: token embeddings (using WordPiece tokenization with a vocabulary of size 30K), position embeddings (encoding token position in the sequence), and segment embeddings (distinguishing between sentence pairs).

RoBERTa (Robustly Optimized BERT Approach) [192] represents a significant enhancement of the BERT architecture through several carefully designed optimizations. One of the key innovations is the implementation of dynamic masking, where the model generates new masking patterns each time a sequence is presented during training. This contrasts with BERT's static masking approach and helps prevent the model from memorizing specific mask patterns, leading to more robust learning.

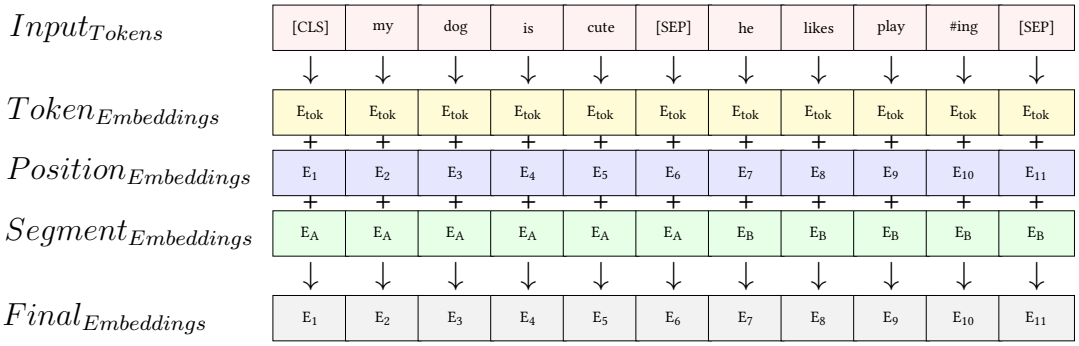


Figure 8: BERT input representation showing the combination of token, position, and segment embeddings. The final embedding for each token is the sum of its corresponding token embedding, position embedding, and segment embedding. [MASK] tokens are used for masked language modeling training.

A crucial modification was the elimination of BERT's NSP task. This seemingly simple change had profound implications: it not only streamlined the training process but also allowed for larger batch sizes, significantly improving training efficiency. The experiments also show that NSP's benefits were minimal compared to the computational overhead it introduced.

RoBERTa also refined the tokenization strategy by implementing *byte-pair encoding* (BPE) with a larger vocabulary of 50K tokens, enabling more nuanced text representation. While maintaining the same architectural scale as $\text{BERT}_{\text{LARGE}}$ with 355M parameters, RoBERTa's training process was substantially more extensive, utilizing a massive 160GB text corpus (compared to BERT's 16GB). This corpus included not only BERT's original

training data but also additional datasets like Common Crawl News [114] and OpenWebText [102], contributing to its improved performance across various NLP tasks.

The success of RoBERTa's optimized training approach inspired several language-specific adaptations, particularly for French. CamemBERT [199] emerged as a significant French language model, applying RoBERTa's architecture and training methodology to French text from the OSCAR corpus [221]. Unlike multilingual models that often compromise performance due to vocabulary distribution across languages, CamemBERT was specifically designed for French, achieving state-of-the-art performance on French NLP benchmarks. Similarly, FlauBERT [174] represents another notable French adaptation, incorporating both RoBERTa's architectural improvements and training strategies while being trained on diverse French corpora. Both models demonstrate the effectiveness of adapting proven architectural innovations to specific linguistic contexts, validating the transferability of RoBERTa's key improvements across different languages and domains.

These modifications, combined with optimized hyperparameters and longer training times, resulted in a model that consistently outperformed BERT on benchmark tasks.

A key limitation of BERT and its variants is the quadratic computational complexity of self-attention with respect to sequence length, restricting input sequences to 512 tokens.

Decoder-only / Auto-regressive Models: GPT and variants Autoregressive language modeling, used in GPT-style models [236, 237], trains the model to predict the next token given all previous tokens in the sequence, as shown in the Figure 9. This objective naturally aligns with the way humans process language left-to-right (in most languages), making it particularly effective for text generation tasks. The model learns to capture complex dependencies and patterns in language by repeatedly predicting the next token in context [38].

The *Generative Pre-trained Transformer* family of models [236] pioneered the use of transformer decoders for generative pre-training. The original GPT architecture consists of 12 transformer decoder layers, 12 attention heads, and a hidden size of 768, using BPE tokenization with a 40K vocabulary. Like BERT, GPT follows a two-stage approach: pre-training and fine-tuning. During pre-training, the model uses Causal Language Modeling on continuous sequences of 512 tokens, initially trained on the BooksCorpus dataset.

GPT-2 [237] introduced significant innovations in multi-task learning and zero-shot transfer. Rather than traditional fine-tuning, GPT-2 treated task-specific learning as unsupervised pre-training examples. The model was trained on WebText, a dataset introduced in the same article and carefully curated from web pages taken from Reddit's outbound links that received positive feedback from the community, chosen for content quality and for the naturally occurring demonstrations of various tasks in varied domains and contexts. This dataset results in 8 million documents for a total of 40 GB of text after de-duplication and cleaning.

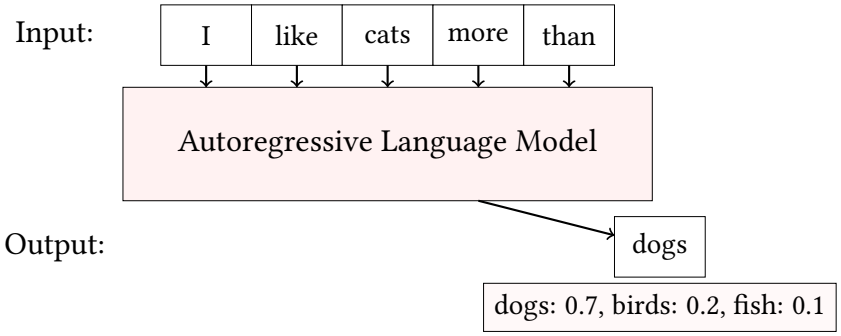


Figure 9: Autoregressive Language Modeling predicts the next token based on previous context.

GPT-2 pioneered the use of prompting or conditioning, where task instructions are prepended to the input sequence, enabling zero-shot task adaptation to perform multiple tasks such as reading comprehension, translation, summarization and question answering without explicit fine-tuning.

GPT-3 [38] scaled this approach dramatically to 175 billion parameters, trained on 300 billion tokens from diverse sources including filtered Common Crawl (410B tokens), WebText2 (19B tokens), Books1 and Books2 (67B tokens combined), and English Wikipedia. The model introduced various prompting paradigms: few-shot (using multiple examples), one-shot (single example), and zero-shot (task description only) learning.

The rapid evolution of increasingly large autoregressive models led to important questions about the relation between data and parameters, also called scaling laws. One of the first contributions in this direction is the Chinchilla scaling law [125] which suggests that model parameters and training tokens should scale proportionally for optimal compute efficiency, revising earlier assumptions about faster parameter scaling. This insight influenced the development of subsequent models like BLOOM [292], PaLM [56], OPT [312], and later on LLaMA [267], each accompanied by instruction-tuned variants (BloomZ [207], Flan-PaLM [59], OPT-IML [134] and Alpaca [259]) optimized for zero- and few-shot performance with natural language instructions.

These developments demonstrated that scale, combined with sophisticated prompting techniques, enables powerful general-purpose language capabilities [219, 138, 28]. The emergence of prompt-based fine-tuning, also called in-context learning (ICL), has made the need for adapting the model parameters to newer domains, data distribution, or tasks less obvious [86], has further enhanced the adaptability of these models to specific tasks while maintaining their general-purpose capabilities.

Encoder-Decoder Models: T5 and BART Encoder-decoder models are a fundamental architecture in NLP, where an encoder transforms an input sequence into an intermediate representation, and a decoder generates an output sequence from this representation. This architecture is particularly effective for tasks such as machine translation, text summarization, and text generation.

Text-to-Text Transfer Transformer (T5) [239] unified various NLP tasks into a single text-to-text format, where both inputs and outputs are treated as text strings. This unification is achieved through task-specific prefixes (e.g., "summarize:" for summarization, "sst2 sentence:" for sentiment analysis) that are prepended to input texts, building upon GPT's prompting approach. T5 was pre-trained on both unsupervised and supervised tasks, using the massive C4 (Colossal Clean Crawled Corpus) [239] dataset made of 750GB of cleaned Common Crawl data, along with supervised tasks from GLUE [279] and SuperGLUE [278] benchmarks.

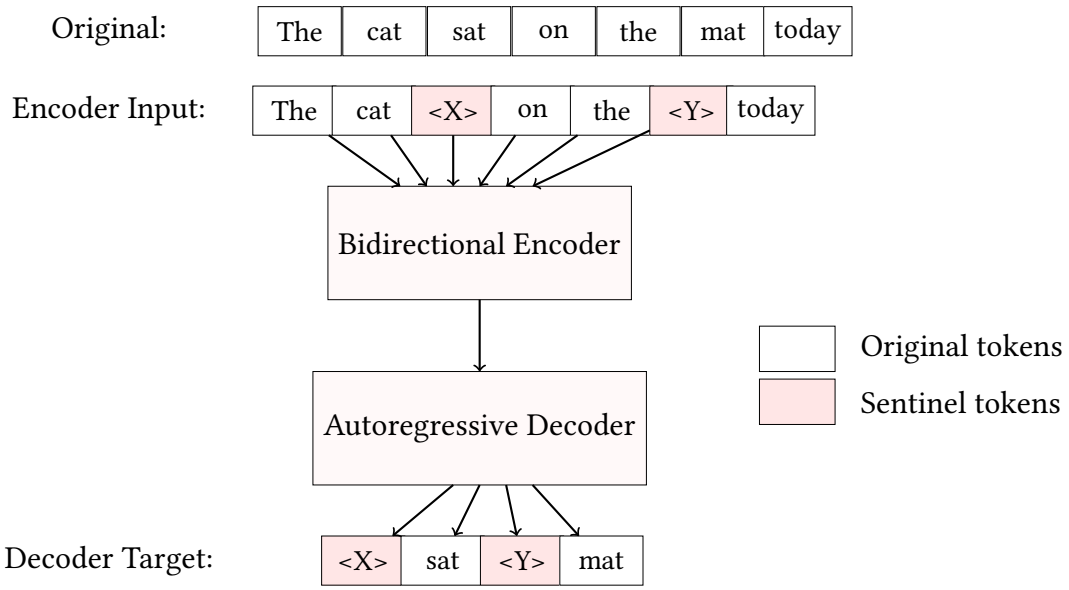


Figure 10: Encoder-Decoder training objective with span corruption. The encoder processes text with corrupted spans (replaced by sentinel tokens <X>, <Y>), while the decoder learns to reconstruct the original spans. This approach combines BERT-style masking with sequence-to-sequence learning.

T5's pre-training introduced span corruption, where randomly sampled spans covering 15% of input tokens are replaced with unique sentinel tokens, offering a more structured alternative to BERT's random masking. During training, the encoder processes the corrupted text while the decoder reconstructs the original spans, with targets consisting of the removed tokens delimited by their associated sentinel tokens as shown in Figure 10. The model has evolved into several variants, including mT5 [299] for multilingual tasks (supporting 101 languages), UL2 [260] with diverse pre-training objectives, and instruction-tuned versions like Flan-T5 [59], Tk-Instruct [283], T0 [246] and Flan-UL2, available in sizes ranging from millions to billions of parameters.

BART [178] combined the bidirectional encoder of BERT with the autoregressive decoder of GPT, featuring 12 layers in both encoder and decoder with a 1024 hidden size. Its pre-training involves sophisticated text corruption strategies: text infilling, where spans of text with lengths drawn from a Poisson distribution ($\lambda = 3$) are replaced with single mask tokens, and sentence permutation, which randomly reorders document sentences. Using the same BPE tokenization and training data as RoBERTa, BART excels at both un-

derstanding and generation tasks, particularly in sequence-to-sequence applications like summarization and translation.

The encoder-decoder architecture's versatility stems from its natural separation of understanding and generation: the encoder captures semantic and structural information from the source text into a continuous representation (context), while the decoder generates target text by considering both this encoded context and its previously generated tokens. This architecture has proven particularly effective for tasks requiring both deep understanding and structured generation, such as machine translation, text generation, and summarization, making it a more effective approach which allows to obtain better performances until the release of ChatGPT and Llama.

Pre-training and Model Adaptation Strategies for Healthcare

Adapting LLM for healthcare applications presents unique challenges in bridging general language capabilities with specialized medical knowledge. These adaptation strategies must address the complexity of medical terminology while ensuring models can process diverse clinical document formats, including progress notes, discharge summaries, and diagnostic reports. Successful adaptation enables models to generate outputs that align with medical reasoning patterns and documentation standards.

Pre-training Approaches Two principal paradigms guide the adaptation of language models for healthcare applications:

Continual pre-training offers an alternative strategy that builds upon existing general-purpose language models by extending their training with medical domain data. This transfer learning approach preserves the model's general language understanding and tokenization process while incorporating specialized medical knowledge using sources like PubMed or MIMIC [142]. This approach is relatively affordable and was used as the first method to obtain a domain-specific variant of BERT in healthcare with BioBERT [176], ClinicalBERT [5], and BlueBERT [228].

Pre-training from scratch domain-specific models, on the other hand involves training them exclusively on medical corpora, enabling them to develop specialized vocabulary and embed domain-specific knowledge. This resource-intensive approach requires substantial medical text data, typically drawn from sources like PubMed abstracts, complete medical articles, and when available, clinical documentation from electronic health records. This foundational training establishes broad medical knowledge and terminology comprehension, creating a base for specialized applications. A significant challenge in this approach is developing effective tokenization strategies to handle complex medical terminology, abbreviations, and specialized nomenclature. Among the first language models, architecture adaptation from-scratch to the healthcare domains arrived with BERT and its variants like PubMedBERT [109] and SciBERT [26].

In-Context Learning In-context learning has emerged with GPT-2 [37] as a powerful paradigm for learning new capabilities and knowledge on-the-fly, making it quite aligned with sparse and diversified medical applications, allowing models to adapt to specific medical tasks without fine-tuning. Models like GPT-4 [220] and Med-PaLM 2 [254] have demonstrated remarkable capabilities in few-shot medical reasoning [217], where they can leverage a small number of examples to perform complex medical tasks (See Figure 11). This approach has been particularly effective in the clinical context, where models can analyze patient cases by referencing similar examples provided in the prompt.

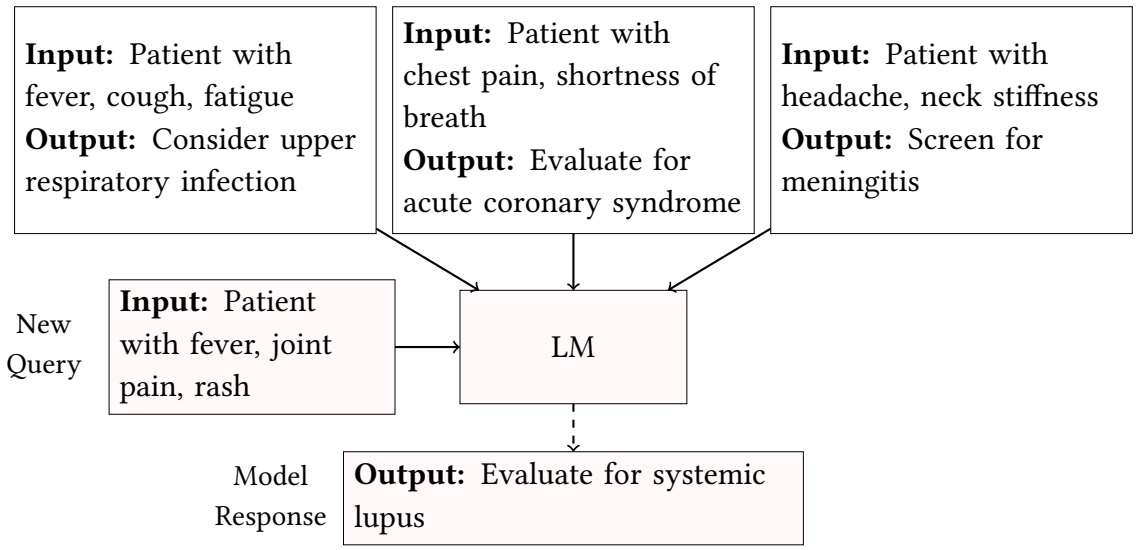


Figure 11: Few-shot learning in medical diagnosis with a number of examples set to three. The language model learns from a few example cases (also called "shots") and applies this knowledge to classify the input or generate appropriate responses for new medical cases. Each example contains an input-output pair showing symptoms and corresponding diagnostic considerations.

Recent studies have shown that carefully crafted medical examples can significantly improve diagnostic accuracy and clinical reasoning. For instance, ChatDoctor [181] and DoctorGLM [298] have demonstrated that providing structured medical examples with detailed symptom-diagnosis relationships helps models better understand clinical presentations and generate more accurate medical assessments.

Chain-of-Thought Reasoning *Chain-of-Thought* (CoT) [285] prompting has revolutionized medical reasoning in LLM by enabling step-by-step clinical reasoning processes (see Figure 12). This approach mirrors the systematic thinking patterns of healthcare professionals, breaking down complex medical decision-making into logical steps.

Medical reasoning benefits significantly from CoT, where models explicitly articulate the progression from symptoms to the final task (e.g, differential diagnoses or codification), considering various factors such as patient history, lab results, and potential complications.

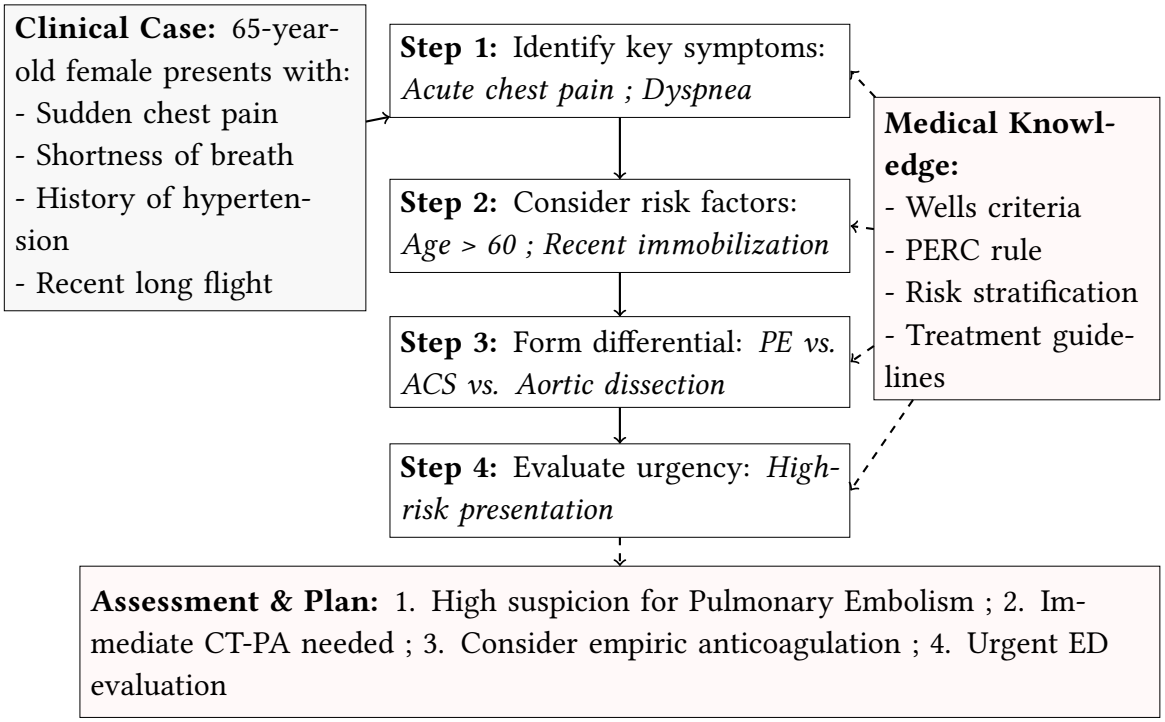


Figure 12: Chain-of-Thought reasoning in medical diagnosis. The model follows a systematic approach to clinical reasoning, breaking down the diagnostic process into logical steps while incorporating relevant medical knowledge. Each step builds upon previous observations and considerations, leading to a comprehensive assessment and management plan.

0.4.5 Downstream Tasks

NLP encompasses a wide range of computational tasks designed to understand, analyze, and generate human language. These tasks form the foundation for various applications in text processing and analysis, ranging from basic linguistic analysis to complex reasoning tasks. While many NLP tasks were initially developed for general domain text, they have been adapted and specialized for various domains, including healthcare.

Named Entity Recognition *Named Entity Recognition* (NER) is a fundamental NLP task that aims to identify and classify named entities in text into predefined categories such as person names, organizations, locations, medical codes, time expressions, quantities, and more (as shown in Figure 13). In the biomedical domain, NER is particularly valuable for extracting structured information from unstructured clinical notes, research papers, and other medical texts. This task presents unique challenges in the biomedical context due to the domain's specialized terminology, frequent abbreviations, complex naming conventions, and the high cost of annotation requiring expert knowledge.

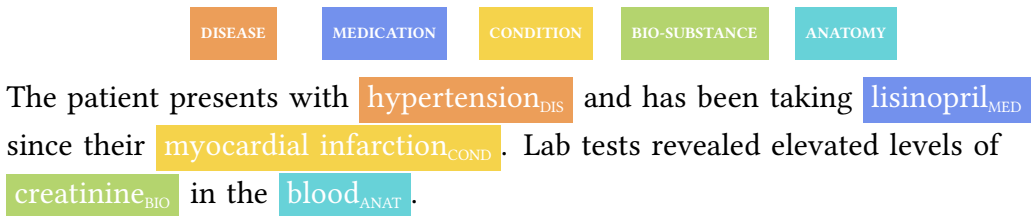


Figure 13: Examples of Named Entity Recognition (NER) in healthcare domains.

For French biomedical texts, several datasets are available: QUAERO [211] contains 103,056 words from drug leaflets and biomedical titles, annotated with 10 entity categories corresponding to UMLS [188] Semantic Groups, with 26,409 entity annotations mapped to 5,797 unique UMLS concepts. E3C [196] provides clinical entity and temporal information annotations, split into 70/10/20 for train/validation/test as shown in Table 1. Mantra-GSC [155] offers biomedical NER annotations from three sources (Medline with 11 classes, EMEA and Patents with 10 classes), similarly split 70/10/20. DEFT-2021 [107] contains 275 clinical cases with 13 types of entities. The PxCorpus [153] includes 1,981 transcribed dialogues with 38 NER classes. For English, BC5CDR [180] serves as a benchmark for chemical and disease entity recognition, while the NCBI-disease [85] corpus targets disease name recognition and normalization.

Subset	Train	Validation	Test
Clinical	87.38 % of layer 2	12.62 % of layer 2	100 % of layer 1
Temporal	70 % of layer 1	10 % of layer 1	20 % of layer 1

Table 1: Description of the sources for E3C.

Relation Extraction *Relation extraction* (RE) is a natural language processing task that aims to identify and classify semantic relationships between entities mentioned in text as shown in the Figure 14. In the biomedical domain, this typically involves detecting meaningful associations between biomedical entities such as genes, proteins, diseases, drugs, and symptoms. For example, a relation extraction system might identify that a particular gene "causes" a disease, a drug "treats" a condition, or a protein "interacts with" another protein. This task is more complex than named entity recognition as it requires understanding not just what entities are present, but how they relate to each other semantically. Relation extraction is crucial for building structured knowledge bases from unstructured text, enabling advanced biomedical applications like drug discovery, clinical decision support, and literature-based discovery.

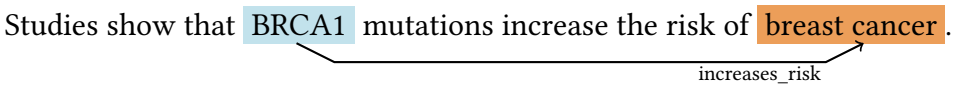


Figure 14: Example of Relation Extraction with an angled arrow showing the relationship between entities.

Currently, relation extraction tasks are primarily represented by English-language datasets, with the Genetic Association Database (GAD) [36] serving as a comprehensive resource of human genetic association studies, providing annotations of gene-disease relationships extracted from biomedical literature.

Text Classification Text classification is a fundamental NLP task that involves categorizing text documents into predefined classes or categories. In the biomedical domain, this typically includes assigning medical specialties, disease codes, or thematic categories to clinical notes, research papers, or patient records. Text classification tasks generally fall into two main types: multi-class and multi-label classification. In multi-class classification, each document belongs to exactly one category from a set of mutually exclusive classes (e.g., assigning a single primary diagnosis). In contrast, multi-label classification allows documents to simultaneously belong to multiple categories (e.g., a clinical case exhibiting multiple conditions or relevant to several medical specialties). The latter is particularly common in biomedical contexts, where patients often present with comorbidities and documents frequently span multiple medical domains.

French healthcare classification datasets include MorFITT [171] (3,624 biomedical abstracts annotated across 12 medical specialties, totaling 5,116 annotations), DiaMed [166] (739 clinical cases annotated with 22 ICD-10 chapters, DEFT-2021 [107] (275 clinical cases annotated with 23 MeSH axes), and PxCorpus (1,981 recordings with 4 intent classes). For English, datasets include HoC [22], LitCovid [47], PubHealth [210], and the N2C2 2006 Smokers [272] dataset.

Question Answering The medical domain benefits from various *question-answering* (QA) datasets, each with distinct characteristics. These datasets cover a broad spectrum of tasks, ranging from pharmacy professional examinations (FrenchMedMCQA) to medical board

questions (MedMCQA), and biomedical research comprehension (PubMedQA). The formats vary considerably, including multiple-choice questions with single or multiple answers, yes/no/maybe questions, and questions requiring in-depth analysis of clinical trials. These datasets are also linguistically diverse, with some also available in French, English, or Chinese, reflecting the international nature of medical research.

FrenchMedMCQA [163] represents the first French medical QA dataset, containing 3,105 pharmacy specialization exam questions. Each question offers 5 options (A through E) and comes from real French pharmacy specialization diplomas. The dataset balances between single-answer (1,080) and multiple-answer (2,025) questions, providing a diverse testing ground for French medical language understanding.

MedMCQA [222] stands as a comprehensive medical dataset comprising 193,155 questions from AIIMS and NEET PG exams. It spans 21 medical subjects across 2.4k healthcare topics, with each question accompanied by detailed explanations. The dataset's diversity is reflected in its question types, including diagnosis (16.39%), treatment (14.36%), and logical reasoning (28.83%), supporting both single and multiple correct answers.

PubMedQA [140] focuses on biomedical research comprehension, featuring 211.3k artificially generated questions alongside 1,000 expert-annotated ones. Questions follow a yes/no/maybe format and are derived from PubMed research articles and abstracts. The dataset emphasizes complex reasoning, with 57.5% of questions requiring inter-group comparisons and 96.5% demanding quantitative reasoning skills.

MMLU's [120] medical component consists of 1,089 questions across 6 medical subjects. As part of a broader evaluation benchmark, this multiple-choice dataset tests both basic and advanced medical knowledge, designed specifically to evaluate model capabilities against human expertise levels.

MedQA [139] offers a multilingual perspective on medical board exams, featuring USMLE questions in English (10,178 training + 1,273 test samples) with parallel datasets in simplified and traditional Chinese. The dataset emphasizes clinical reasoning and knowledge retrieval, complemented by accompanying medical textbook knowledge sources.

SciQ [287] presents a crowdsourced approach to science questions, including medical topics. Its multiple-choice format derives questions from science textbooks, with most questions including their source passages, ensuring context-rich learning opportunities.

BioASQ 7b [268] specializes in biomedical question-answering, incorporating various question types including factoid, yes/no, and list-type questions. The dataset demands precise answer extraction and tests comprehensive biomedical domain expertise.

Evidence Inference 2.0 [82] targets clinical trial analysis through 12,616 prompts derived from 3,346 articles. It employs *Intervention-Comparator-Outcome* (ICO) triplets, requiring sophisticated understanding of clinical trial reports and their implications for treatment effects.

Semantic Similarity Semantic similarity is a NLP task that aims to quantify the degree of semantic relatedness between texts, ranging from words and phrases to entire docu-

ments. In the biomedical domain, this task is particularly valuable for identifying related medical concepts, finding similar clinical cases, or determining if two medical descriptions refer to the same condition despite using different terminology. Unlike classification, which assigns discrete categories, semantic similarity produces continuous scores that reflect the gradation of relatedness. These scores typically range from 0 (completely unrelated) to a maximum value (identical or perfectly related). Semantic similarity assessment in medical texts is especially challenging due to the domain's complex terminology, where similar concepts may be expressed using entirely different vocabularies, and subtle differences in description might indicate significant clinical distinctions.

In French, CLISTER [123] provides 1,000 manually annotated clinical case pairs with similarity scores (0-5), based on three dimensions: surface similarity, semantic similarity of medical concepts, and clinical compatibility. DEFT-2020 [40] offers similarity scoring (0-5) across different medical text types including clinical texts, encyclopedia entries, and drug labels, with annotations based on annotator intuition. The corpus contains 1,010 sentence pairs from the CLEAR corpus [104].

Natural Language Inference *Natural Language Inference* (NLI), also known as textual entailment, is a task that evaluates the logical relationship between a premise (a given statement) and a hypothesis (a potential conclusion). The goal is to determine whether the hypothesis can be inferred from the premise. Typically, the relationship is classified into three categories: entailment (the hypothesis logically follows from the premise), contradiction (the hypothesis contradicts the premise), or neutral (the premise neither confirms nor contradicts the hypothesis). In the biomedical domain, NLI is particularly valuable for verifying clinical reasoning, checking if conclusions drawn from patient information are valid, and assessing whether medical texts contain contradictory information. This task requires deep semantic understanding and often domain-specific knowledge to correctly identify logical relationships between medical statements.

The English healthcare-specific MedNLI dataset [251] focuses on clinical domain inference using MIMIC-III notes, with premises drawn from the Past Medical History sections and hypotheses generated by clinicians. The dataset contains 14,049 sentence pairs (11,232 train, 1,395 dev, 1,422 test) with entailment annotations. SciTail [149] provides 27,000 entailment pairs derived from science question answering tasks, where hypotheses are created from science questions and correct answer candidates, while premises come from relevant web sentences. The dataset is unique in using naturally occurring sentences rather than artificially created ones. Currently, there are no prominent French datasets specifically dedicated to natural language inference in the medical domain.

Part-Of-Speech *Part-of-Speech* (POS) tagging is a NLP task that involves labeling each word in a text with its corresponding grammatical category, such as noun, verb, adjective, or adverb as shown in Figure 15. In the biomedical domain, POS tagging serves as a preprocessing step for many advanced NLP applications, including *Named Entity Recognition*, syntactic parsing, and information extraction. Medical texts present unique challenges for POS tagging due to their specialized vocabulary, complex noun phrases, abbreviated terms, and domain-specific syntactic patterns. Accurate POS tagging in clinical and biomedical texts

enables better understanding of the grammatical structure of medical language, which in turn improves the performance of downstream tasks like relation extraction and semantic analysis.

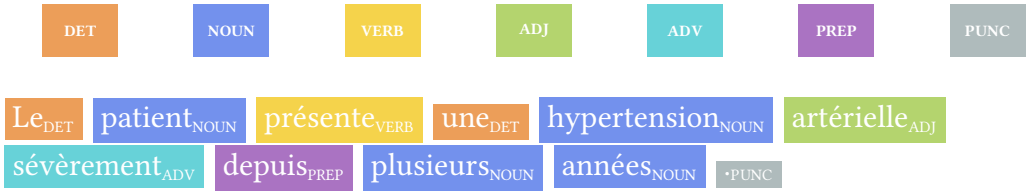


Figure 15: Example of Part-of-Speech (POS) tagging in French medical text. Each word is highlighted with a color corresponding to its grammatical category, with tiny subscripts indicating the specific part of speech.

Two significant French corpora provide part-of-speech annotations for medical texts. The CAS corpus [105] consists of 3,790 clinical cases with 31 distinct POS tags. These annotations were automatically generated using the Tagex tool[1] and subsequently validated against manual annotations, achieving a high precision of 98%. The ESSAI corpus [73] encompasses 7,247 clinical trial protocols and CAS corpus [105] comprises 3,790 clinical cases, annotated with a more granular set of 41 POS tags using TreeTagger [247]. Neither dataset originally included predefined data splits; therefore, both were randomly partitioned following the same distribution: 70% for training, 10% for validation, and 20% for testing purposes.

[1] https://allgo.inria.fr/app/tagex

0.4.6 Downstream Tasks Metrics

Evaluating NLP downstream tasks requires specialized metrics that capture the unique characteristics of each task. Here, we present the primary metrics used to assess model performance across various textual NLP tasks.

Named Entity Recognition For NER, the SeqEval F1 score is the standard metric, which extends traditional F1 to sequence labeling by considering entity-level rather than token-level performance:

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1}$$

where Precision measures the percentage of predicted entities that are correct:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{2}$$

and Recall measures the percentage of actual entities that were correctly identified:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{3}$$

Unlike token-level metrics, SeqEval only counts an entity as correct if both its span boundaries and entity type are correctly predicted.

Accuracy is also sometimes reported, measuring the proportion of correctly predicted entities among all predictions:

$$\text{Accuracy} = \frac{\text{Number of correctly predicted entities}}{\text{Total number of entities}} \tag{4}$$

Text Classification For text classification tasks, Weighted F1 balances precision and recall while accounting for class imbalance:

$$\text{Weighted F1} = \sum_{i=1}^{n} w_i \times \text{F1}i \tag{5}$$

where w_i represents the proportion of samples belonging to class i, and F1_i is the F1 score for that class i. This metric is particularly valuable in medical contexts where certain conditions or specialties may be underrepresented but equally important to identify correctly.

Question Answering Question answering tasks utilize specialized metrics depending on the answer format. For multiple-choice questions, *Exact Match Rate* (EMR) measures the percentage of questions where the model's prediction exactly matches the correct answer:

$$\text{EMR} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(\hat{y}i = y_i) \tag{6}$$

where $\mathbb{1}$ is the indicator function, $\hat{y}i$ is the predicted answer, and y_i is the correct answer.

For multiple-answer questions, Hamming Score evaluates partial correctness:

$$\text{Hamming Score} = \frac{1}{N}\sum_{i=1}^{N} \frac{|\hat{Y}i \cap Y_i|}{|\hat{Y}_i \cup Y_i|} \tag{7}$$

where $\hat{Y}i$ is the set of predicted answers and Y_i is the set of correct answers for question i.

Semantic Similarity For semantic similarity tasks, *Euclidean Distance-based Relative Metric* (EDRM) measures how well a model's predicted similarity scores align with human judgments:

$$\text{EDRM} = 1 - \frac{\sum_{i=1}^{N}(s_i - \hat{s}i)^2}{\sum i = 1^{N}(s_i - \bar{s})^2} \tag{8}$$

where s_i is the gold standard similarity score, $\hat{s}i$ is the predicted score, and $\bar{s}$ is the mean of all gold standard scores.

Spearman's rank correlation coefficient assesses how well the ranking of text pairs by predicted similarity aligns with human judgments:

$$\rho = 1 - \frac{6\sum_{i=1}^{N} d_i^2}{N(N^2 - 1)} \tag{9}$$

where d_i is the difference between the ranks of corresponding predicted and gold standard similarity scores, and N is the number of text pairs.

Part-Of-Speech Tagging POS tagging employs SeqEval F1 similar to NER, but with evaluation conducted at the token level rather than entity level:

$$\text{Accuracy} = \frac{\text{Number of correctly tagged tokens}}{\text{Total number of tokens}} \tag{10}$$

Additionally, per-class F1 scores are calculated to assess performance on specific grammatical categories, which is particularly important for identifying specialized medical terminology with the correct grammatical function.

These comprehensive metrics provide a robust framework for evaluating model performance across a diverse range of textual NLP tasks, enabling precise assessment of language understanding capabilities in specialized domains such as healthcare.

0.5 Speech Processing

0.5.1 Raw Signal

Audio signals are fundamentally represented as numerical sequences capturing sound amplitude variations over time. The temporal resolution of these signals is defined by the sampling rate, measured in Hz (samples per second), with higher rates corresponding to increased audio fidelity.

The precision of amplitude measurement is determined by bit depth, which defines the number of possible discrete values each sample can take. An 8-bit audio sample represents 2^8 = 256 distinct amplitude levels, while professional recording equipment typically uses 16-bit (65,536 levels) or 24-bit (16,777,216 levels) depth for higher fidelity. These higher bit depths capture nuances in vocal timbre and acoustic characteristics. Recording format also matters: mono (single channel) serves basic voice recordings, while stereo (dual channel) captures directional and spatial sound characteristics.

The information density of audio creates significant storage challenges. A 24-hour continuous recording at 16 kHz with 16-bit depth requires approximately 2.7GB, while its textual transcription might occupy only 100KB, a 27,000 reduction factor. Similarly, a 15-minute high-quality stereo recording at 44.1 kHz with 24-bit depth requires about 150MB, compared to just 5KB for a text summary. This disparity impacts storage infrastructure, computational resources for processing, research budgets, data pipeline design, and implementation of efficient storage solutions for speech projects.

0.5.2 Spectrogram

While raw audio signals capture amplitude variations over time, they don't directly represent critical features such as frequency components and phonetic characteristics. In speech analysis, the spectrogram provides a sophisticated visualization of frequency distributions over time, offering crucial insights for various linguistic and acoustic applications. This frequency-time representation is particularly valuable in speech processing, where subtle acoustic patterns can indicate specific phonetic elements.

The spectrogram is constructed through sequential frequency analysis of short time windows, typically using the Fourier transform [93]. Each spectrum represents the frequency composition within a specific time window, revealing linguistically significant patterns. In speech analysis, vowels typically show formant frequencies with the first formant (F1) between 273-805Hz and the second formant (F2) between 770-2524Hz (as measured in French vowels produced by female speakers [201]), while fricative consonants like /s/ produce distinctive energy in higher frequency bands around 4000-8000Hz [50].

The temporal evolution of these frequency patterns creates a comprehensive spectrogram, visualized as a frequency/time image. In speech applications, this representation enables to identify and track various phonetic elements and speech characteristics. For ex-

ample, in speaker identification or emotion recognition, the fundamental frequency (F0) patterns differ significantly between speakers: adult male voices typically show F0 around 100-150Hz, while females range from 170-220Hz [92, 234]. Variations in these patterns can also indicate different speaking styles [8], emotional states [25], or accent characteristics [262].

For enhanced perceptual relevance, frequency components are often converted to the mel scale, creating mel-spectrograms that better align with human auditory perception. This transformation is particularly valuable in speech recognition and synthesis, as it better represents how listeners actually perceive sound. The *Mel-Frequency Cepstral Coefficients* (MFCC) [76] provide an even more refined representation by applying a discrete cosine transform to the mel-scaled frequencies. MFCCs have proven especially useful across multiple speech domains.

0.5.3 Speech Feature Extraction

Self-Supervised Learning (SSL) represents a paradigm shift in machine learning [277, 66] where models learn from the inherent structure of data rather than human-annotated labels, fundamentally transforming speech feature extraction. While traditional approaches relied on hand-crafted features like MFCCs, SSL has emerged as a powerful technique for learning robust speech representations from vast amounts of unlabeled audio data [190, 46]. These learned feature extractors capture more nuanced aspects of speech signals, forming the foundation of modern speech processing systems. SSL objectives can be broadly categorized into contrastive approaches, which maximize similarity between related speech segments while minimizing similarity with unrelated ones [274], and non-contrastive approaches, which often involve reconstruction tasks or predictive modeling of speech features [62]. This self-supervised paradigm has proven particularly effective when combined with deep learning architectures, enabling systems to extract meaningful representations that capture both acoustic and linguistic information without requiring extensive labeled datasets [288].

In the following sections, we will explore two main types of learned representations: continuous representations, which capture information in a dense and continuous manner, and discrete representations, which encode information as distinct, quantized units.

Continuous Representation

Recent advances in speech SSL models have enabled the extraction of high-quality continuous representations from speech signals. These learned representations, or embeddings, capture rich acoustic and linguistic information that can be leveraged for downstream tasks.

Wav2Vec. Wav2vec 2.0 [17] represents a significant advancement in speech SSL, introducing a powerful architecture that combines feature extraction with contextual representation learning. The model employs a *Convolutional Neural Network* (CNN) encoder to extract latent speech representations from a raw signal, followed by a quantization module

and a transformer encoder. Its training objective involves predicting quantized latent representations of masked regions, using a contrastive loss function where negative examples are sampled from other masked regions within the same sequence. This architecture has demonstrated remarkable performance, particularly in *Automatic Speech Recognition* (ASR) tasks, achieving state-of-the-art results with minimal labeled data, as little as 10 minutes of transcribed speech [15].

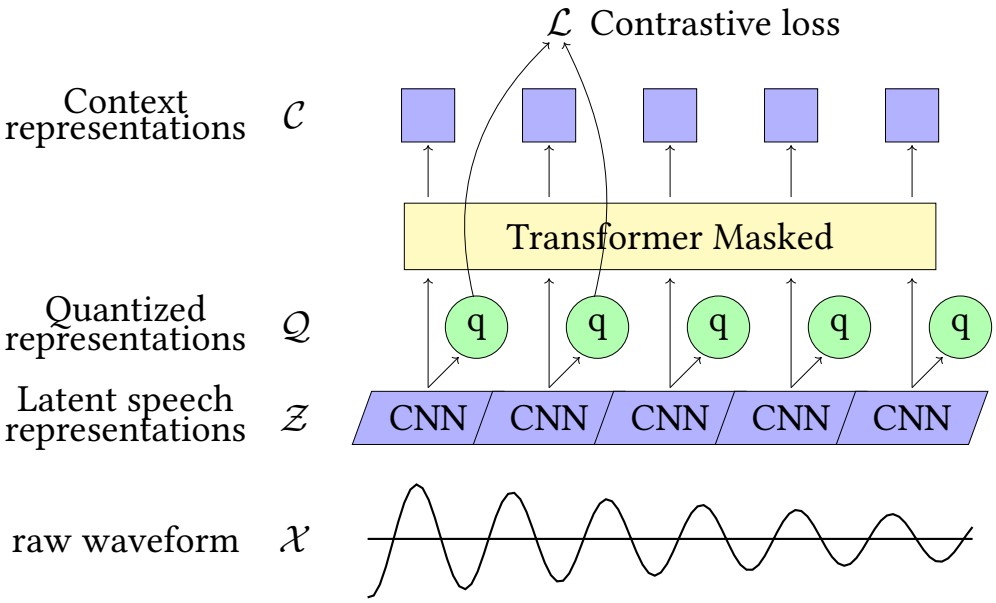


Figure 16: Architecture of wav2vec 2.0. The model processes raw waveform through a CNN encoder to obtain latent speech representations, which are then quantized. A Transformer processes these representations with masked prediction, using a contrastive loss between the context representations and quantized targets.

HuBERT. HuBERT (Hidden-Unit BERT) [127] introduces an innovative approach to speech SSL by incorporating iterative pseudo-labeling. While sharing architectural similarities with Wav2Vec 2.0, HuBERT distinguishes itself through its unique training objective [127]. Instead of contrastive learning, it employs a prediction-based approach where the model learns to predict cluster assignments of masked regions [127]. These clusters are initially derived from MFCC features and later refined using the model's own representations in subsequent iterations. This iterative refinement process enables HuBERT to capture increasingly sophisticated speech patterns, leading to superior performance in various speech processing tasks [288].

WavLM. WavLM [48] builds upon HuBERT's framework while introducing significant enhancements for real-world speech scenarios. It maintains the masked prediction objective but extends it with a denoising component and explicit training on diverse speech conditions, including clean, noisy, and overlapping speech. Architecturally, WavLM introduces a gated relative position bias in its self-attention mechanism to better model multi-speaker scenarios. Like its predecessors, WavLM employs a CNN encoder that converts raw

waveforms into latent representations with a 20ms stride, followed by transformer layers for contextualization. The resulting embeddings (768 or 1024-dimensional, depending on model size) capture both acoustic and linguistic features, with different layers specializing in different aspects of speech [250].

Whisper. Unlike the previously discussed models that focus on self-supervised pre-training objectives, Whisper [235] employs a supervised approach trained on 680,000 hours of labeled multilingual and multitask data. While it shares the encoder-decoder transformer architecture common in modern speech systems, Whisper distinguishes itself through its input processing and training methodology. Rather than operating directly on raw waveforms like Wav2Vec 2.0, HuBERT, and WavLM, Whisper first converts audio into mel-spectrogram features. Its encoder processes these spectrograms to produce contextualized embeddings where each vector represents 25ms of speech with a 10ms stride, offering denser temporal coverage than WavLM's 20ms stride. The encoder generates 1024-dimensional feature vectors (in the large model) that capture both local acoustic properties and broader linguistic context. What truly sets Whisper apart is its multitask training approach, which enables a single model to perform speech recognition, language identification, and translation across 99 languages.

Discrete Units

Speech Units

Traditional approaches to discrete speech representation often relied on phonetic units or hand-crafted features. More recently, self-supervised learning has enabled the discovery of discrete speech units directly from data [17, 52]. These learned units can capture meaningful acoustic and linguistic patterns without requiring explicit phonetic annotations [173]. For instance, Wav2Vec 2.0 introduced quantized speech representations through Gumbel-Softmax quantization [17], while HuBERT employed iterative k-means clustering to discover discrete units [127].

However, these discrete representations are primarily limited by their dependence on vocoders for speech reconstruction [154]. Vocoders are required to convert the discrete units back into continuous audio waveforms, introducing additional complexity and potential quality degradation in the reconstruction process [154]. This limitation has motivated the development of neural codec approaches that can both discretize and reconstruct speech signals efficiently.

Neural Codecs

Neural codecs represent a significant advancement in speech processing, offering efficient discrete representations while maintaining high-quality reconstruction capabilities

[307]. These models combine the benefits of discrete representations with end-to-end training, eliminating the need for separate vocoders.

EnCodec. EnCodec [87] is a neural audio codec that achieves state-of-the-art results across multiple sampling rates (8 to 48 kHz) and bitrates (1.5 to 24 kbps). The model employs a streaming convolutional encoder-decoder architecture with a quantized latent space using *Residual Vector Quantization* (RVQ). A key innovation is its training approach that combines reconstruction losses (in both time and frequency domains) with adversarial losses from a multi-scale *Short-Time Fourier Transform* (STFT) discriminator, effectively reducing artifacts in the reconstructed audio. EnCodec introduces a novel loss balancer mechanism that automatically adjusts the weights of different loss terms based on their gradients, stabilizing the challenging multi-objective training process. The model achieves real-time encoding and decoding on a single CPU core while maintaining superior perceptual quality compared to established codecs like Opus and EVS across various audio domains, including speech, noisy speech, and music. Additionally, a lightweight Transformer model can be used for entropy coding to further compress the representation by up to 40% while maintaining audio quality.

SpeechTokenizer. SpeechTokenizer [313] introduces a unified approach to speech tokenization that hierarchically disentangles different aspects of speech information across RVQ layers. The model adopts an encoder-decoder architecture with RVQ, where the first layer captures semantic content while subsequent layers encode paralinguistic information like timbre and prosody. Unlike previous approaches requiring separate tokenizers, SpeechTokenizer employs semantic distillation from HuBERT to guide the first RVQ layer, enabling better alignment between tokens and linguistic content. The model uses a convolution-based encoder-decoder network with a two-layer Bi*Long Short-Term Memory* (LSTM) replacing EnCodec's LSTM to enhance semantic modeling capabilities. SpeechTokenizer introduces a novel "D-axis" continuous distillation loss that calculates cosine similarity across each dimension rather than at each timestep, providing richer supervision signals. Experiments on their proposed SLMTokBench benchmark demonstrate that SpeechTokenizer achieves comparable reconstruction quality to EnCodec while showing stronger performance on speech language modeling tasks. The first-layer tokens exhibit high mutual information with text (significantly outperforming EnCodec's first-layer tokens on Phone-Normalized Mutual Information), while the complete token set maintains high-quality audio reconstruction capabilities, making it particularly suitable for unified speech language models.

Mimi. Mimi [88] is a neural audio codec developed specifically for the Moshi speech-text foundation model, designed to enable real-time dialogue applications. Unlike previous approaches requiring separate encoders for semantic and acoustic tokens, Mimi introduces a novel "split RVQ" architecture that addresses the semantic-acoustic trade-off. The model builds on SoundStream and EnCodec's SeaNet autoencoder design but adds Transformer modules in the bottleneck (8 layers, 8 heads) to improve both audio quality and semantic information capture. Mimi operates at 12.5Hz (one token per 80ms) with 8 quantizers of 2048 codes each, resulting in a bitrate of 1.1kbps. A key innovation is its distillation approach, where non-causal semantic information from WavLM is distilled into the first level of quantization, while the remaining levels capture acoustic details. To prevent the

semantic-acoustic trade-off from degrading audio quality, Mimi employs a split RVQ where semantic information is distilled into a plain VQ, with a parallel 7-level RVQ for acoustic information, and their outputs are summed. This architecture maintains causal processing for streaming applications while effectively combining semantic and acoustic information. Another notable finding is that training with adversarial losses only (removing reconstruction losses) significantly improves subjective audio quality despite degrading objective metrics [88].

0.6 Multi-Modal Speech-Text Language Modeling

Speech constitutes a primary mode of human communication, yet it remains significantly underrepresented in digital data compared to text. While text-based language models can train on trillions of tokens harvested from the web, high-quality speech datasets are orders of magnitude smaller. This data disparity creates a fundamental challenge for developing robust speech understanding systems, particularly for domain-specific applications where speech data is even more limited.

This asymmetry in data availability, however, opens up promising research directions. By developing methods to effectively transfer the rich semantic knowledge embedded in text-based models to the speech domain, we can potentially overcome the inherent limitations of speech data scarcity. Recent breakthroughs in self-supervised learning have made significant progress toward this goal, enabling models to extract powerful representations from unlabeled speech. These approaches have yielded impressive results across various speech processing tasks, including automatic speech recognition, speaker identification, and emotion detection.

In this section, I present how these speech representations can be effectively integrated with text-based language models to create unified multi-modal systems. By leveraging the discrete speech tokens discussed earlier, we can develop architectures that process both speech and text within a common framework. This integration enables more natural human-machine interactions and unlocks new capabilities in speech understanding, generation, and translation that surpass what is possible with unimodal approaches.

0.6.1 Modality Integration

The integration of speech modalities into language models has evolved through two distinct phases, each articulated around the encoder-decoder architecture, marking significant progress in the field and addressing different challenges in multi-modal processing.

Phase 1: Continuous Representation Integration

Early approaches to speech-text integration primarily relied on encoder-decoder architectures like SpeechT5 [7] and Whisper [235]. These models established the initial bridge between speech and text processing, treating the conversion as a sequence-to-sequence task. SpeechT5 introduced a unified-modal encoder-decoder framework that could handle various speech-text tasks, including ASR, *Text-To-Speech* (TTS), speech translation, and voice conversion through a shared encoder-decoder architecture with task-specific adapters. Whisper demonstrated remarkable robustness by training on 680,000 hours of multilingual and multitask supervised data, achieving strong zero-shot generalization to unseen datasets.

While these models achieved impressive results for their specific tasks, they suffered from several limitations when considered as general-purpose speech-text interfaces. A significant constraint was their decoder architecture, which typically lagged behind the rapid

advancements in text language modeling. As the NLP community progressed from GPT-2-scale models to much larger and more capable architectures, speech-text models continued to use relatively simple decoders that couldn't match the sophisticated generation and understanding capabilities of state-of-the-art text LLMs. This architectural gap limited their ability to produce nuanced, contextually appropriate responses or to leverage the rich world knowledge embedded in modern language models.

Models like Qwen-Audio [57] and SALMONN [258] demonstrated that continuous speech features could be effectively processed by modern transformer architectures through specialized audio encoders and showed improvement by leveraging textual knowledge. Qwen-Audio employed a unified architecture that processed both audio and text inputs by connecting a single audio encoder (initialized from Whisper-large-v2) to a large language model (Qwen-7B [19]), enabling zero-shot generalization across modalities. This approach allowed Qwen-Audio to handle diverse audio types, including human speech, natural sounds, music, and songs within a single model architecture.

A key innovation in Qwen-Audio was its multi-task training format framework, which addressed the challenge of one-to-many mapping caused by variations in textual labels across different datasets. By conditioning the decoder on a sequence of hierarchical tags (including transcription tags, audio language tags, task tags, text language tags, and timestamp tags), Qwen-Audio enabled knowledge sharing between similar tasks while avoiding interference. The model also incorporated speech recognition with word-level timestamp prediction, which improved performance on grounding-based QA tasks.

SALMONN further advanced this approach by introducing a dual encoder structure with a speech encoder from Whisper and a BEATs audio encoder, enabling the processing of various audio types while maintaining alignment with text representations through a window-level Q-Former connection module. This continuous representation approach offered several advantages, including the ability to process various audio types beyond speech (such as environmental sounds and music) and the preservation of rich signal information, including prosody, speaker characteristics, and emotional content.

However, these continuous representations posed significant challenges for efficient training and inference. Their high dimensionality, often hundreds or thousands of times larger than text embeddings, resulted in substantial computational requirements and memory usage. This dimensionality gap created architectural challenges when integrating with text-based models, often requiring complex projection layers or dimension reduction techniques that could introduce information loss. Additionally, the lack of discretization made it difficult to leverage the advances in text-based language modeling that rely on discrete token prediction objectives.

Phase 2: Discrete Speech Tokens Integration

The current state-of-the-art approaches represent a significant evolution, utilizing discrete speech tokens that enable more efficient integration with traditional language modeling architectures. This discretization approach transforms the continuous speech signal into a sequence of tokens from a finite vocabulary, similar to text tokenization, allowing speech

to be processed using the same mechanisms developed for text language modeling and is more suitable for causal language modeling.

Discrete tokens offer substantial computational efficiency by reducing the dimensionality of speech representations, leading to faster training and inference compared to continuous representations. Their architectural compatibility with existing LLM frameworks is notable, as they align naturally with architectures optimized for processing discrete symbols, enabling seamless integration without significant modifications. This unified vocabulary approach allows multiple modalities to be handled within a common framework, enabling models to learn cross-modal relationships more effectively while significantly reducing memory requirements due to the compact nature of discrete representations.

Modern systems like GSLM [4], TWIST [116], and SpiritLM [215] demonstrate the effectiveness of discrete representations for speech language modeling:

GSLM (Generative Spoken Language Model) pioneered the approach of building purely speech-based language models without relying on text supervision. The system consists of three components: a speech tokenizer that converts raw audio into discrete units, a language model trained on these units, and a unit-to-speech module for generation. GSLM used HuBERT for feature extraction, followed by k-means clustering to create discrete tokens, achieving the best performance with 200 clusters at 50Hz. The model was trained on 6,000 hours of speech and demonstrated the ability to generate coherent speech continuations while preserving speaker characteristics and prosodic elements that are typically lost in text-based approaches. GSLM established benchmarks for evaluating speech language models, including sWUGGY and sBLIMP for lexical and syntactic modeling.

TWIST (Textually Warm-Initialized Speech Transformer Language Models) built upon GSLM's foundation by introducing a novel approach that leverages pretrained text language models to improve speech language modeling. Despite the different granularity levels between speech tokens (phoneme-state level) and text tokens (subword level), TWIST demonstrated that initializing a speech language model from a pretrained text model like OPT or LLaMA provides consistent performance improvements. The authors conducted extensive empirical analysis on various aspects of the model design, including the effect of model scale (from 125M to 13B parameters), dataset size (from 1% to 100% of their 150,000 hours corpus), speech tokenizer configurations, and model architectures. Their findings showed that both model and data scaling significantly improve performance, with their largest 13B parameter model achieving state-of-the-art results on speech benchmarks. TWIST also contributed two spoken versions of the StoryCloze benchmark to better evaluate contextual understanding in speech models. Human evaluations confirmed that TWIST models generate more coherent and natural speech compared to cold-start models of equivalent size.

SpiritLM (Interleaved Spoken and Written Language Model), the most recent advancement, extended speech language modeling by creating a multimodal foundation model that freely mixes text and speech. Built on a 7B pretrained text language model, SpiritLM was

continuously trained on interleaved speech and text data, with sequences concatenated as a single stream of tokens using a word-level interleaving method. The model comes in two versions: a BASE version using HuBERT phonetic units and an EXPRESSIVE version that incorporates pitch and style tokens to model vocal characteristics. For text, both versions use subword BPE tokens. This architecture allows SpiritLM to maintain the semantic capabilities of text models while adding expressive speech generation. The model demonstrates impressive few-shot learning capabilities across modalities, enabling tasks like ASR, TTS, and speech classification without task-specific fine-tuning. SpiritLM's bitrate efficiency makes it particularly suitable for applications requiring both content preservation and expressive speech generation.

0.6.2 Alignment Datasets

Developing multi-modal text-speech models relies heavily on specialized datasets that serve different purposes. These datasets can be broadly categorized into unsupervised, massive-scale collections and carefully labeled, task-specific datasets.

Unsupervised, massive-scale datasets prioritize quantity and diversity over precise annotations. LibriLight [144] stands as a prime example with 60,000 hours of unlabeled speech data derived from audiobooks, enabling self-supervised learning at scale. Similarly, Spotify Podcasts [65] offers approximately 100,000 hours of conversational audio with rough transcript alignments. VoxPopuli [280] contributes 400,000 hours of parliamentary speech across 23 languages, while YouTube-derived datasets provide virtually unlimited, though noisy, speech data. These massive collections are crucial for pre-training foundation models like GSLM, TWIST, and SpiritLM, allowing them to learn general speech representations and patterns. However, they often contain alignment errors, background noise, and varying recording qualities that can introduce challenges during training.

In contrast, task-specific labeled datasets offer high-quality annotations but at a much smaller scale.

LibriSpeech [223] provides 1,000 hours of carefully transcribed audiobook readings with precise word-level alignments.

VCTK [137] contains studio-quality recordings from 109 speakers with exact transcriptions for TTS applications.

Fisher [64] and **Switchboard** [101] offer conversational speech with detailed turn-taking annotations for dialogue modeling.

CommonVoice [9] is a crowdsourced multilingual speech corpus where volunteers contribute recordings of text prompts and validate others' contributions. It contains over 33,500 hours of speech across 134 languages and 350,000 distinct speakers, with varying degrees of coverage. The dataset is designed to democratize speech technology by providing freely available data that represents diverse accents, demographics, and linguistic contexts.

These datasets enable supervised learning for specific applications like ASR, TTS, or speech translation, but their limited size and domain specificity can restrict model gener-

alization.

Weakly supervised datasets bridge the gap between fully unsupervised and carefully labeled collections.

GigaSpeech [45] exemplifies this approach with 10,000 hours of multi-domain English speech from audiobooks, podcasts, and YouTube, covering both read and spontaneous speaking styles across diverse topics. It employs forced alignment and segmentation to create sentence-level training data while filtering out low-quality transcriptions.

Similarly, **People's Speech** [97] provides 30,000 hours of diverse English speech with commercial-use licensing, collected from appropriately licensed Internet sources. Unlike many datasets focused on read speech in clean environments, People's Speech includes content from government recordings, interviews, health discussions, and more, with natural background noise that better represents real-world conditions.

The trade-off between these dataset types is evident in how different models utilize them. GSLM primarily leveraged unsupervised data to learn speech representations without text supervision. TWIST demonstrated that combining massive unsupervised pretraining (150,000 hours) with the structural knowledge from text models yields superior performance. SpiritLM took a hybrid approach, using a small but carefully curated parallel corpus for speech-text alignment while continuing to train on larger unsupervised collections.

Overall, the observations suggest that the optimal approach combines multiple dataset types: initial pre-training on massive unsupervised data to learn general patterns, followed by fine-tuning on weakly supervised and high-quality labeled datasets to enhance performance on specific tasks. This strategy allows models to benefit from the scale of unsupervised data while maintaining the precision offered by carefully annotated collections.

0.6.3 Instruction Tuning

Instruction tuning has significantly enhanced model performance on diverse tasks without task-specific fine-tuning, while also improving robustness. Given these notable advances in the text domain, researchers have naturally sought to extend these benefits to speech processing, where similar improvements could transform human-machine interaction. Recent advances in cross-modal instruction tuning have produced several innovative architectures, with particular emphasis on multi-stage training approaches to effectively bridge modalities.

SpeechGPT [310], built on LLaMA-13B, introduced a comprehensive three-stage training methodology. The first stage focuses on paired speech-text data training, where the model learns to process and align speech inputs with corresponding textual representations. This foundation stage establishes basic cross-modal understanding capabilities. The second stage introduces speech instruction data, where the model learns to follow specific commands and instructions in the speech domain. This stage utilizes carefully curated datasets containing diverse speech-based instructions and their corresponding responses, enabling the model to understand and execute spoken commands effectively. The third

stage, termed chain-of-modality instruction training, represents a significant innovation. In this stage, the model learns to handle complex interactions involving multiple modality transitions. This approach represents a substantial improvement over previous models like GSLM and TWIST, which primarily focused on learning speech representations or aligning speech with text, but lacked the ability to follow complex instructions across modalities. Unlike SpiritLM, which used interleaved speech-text training, SpeechGPT's staged approach allows for more systematic acquisition of cross-modal capabilities, resulting in more robust performance on instruction-following tasks and more natural interactions with users. Technically, SpeechGPT employs discrete speech representations using mHuBERT [306] to tokenize speech into 1024 discrete units obtained using k-Means, which are then incorporated into the LLM's vocabulary.

Complementing these developments, Moshi [89] introduced breakthrough capabilities in real-time spoken dialogue, featuring ultra-low latency processing and multi-stream architecture enabling natural interruptions and overlapping speech. Its integration with Helium LLM and Mimi neural codec represents a significant advance in natural human-AI interaction, achieving theoretical latency as low as 160ms. The model's architecture specifically addresses the challenges of real-time interaction, allowing for more natural conversation flow and immediate response generation. While previous models like Qwen-Audio made important strides in multimodal understanding, Moshi fundamentally reimagines the interaction paradigm by prioritizing real-time responsiveness. Unlike earlier approaches that processed complete utterances before generating responses, Moshi's streaming architecture enables it to begin formulating responses while still receiving input, similar to human conversation patterns.

0.6.4 Downstream Tasks

Evaluation of multi-modal language models relies on several key benchmarks spanning linguistic competence, reasoning abilities, and modality transfer capabilities. The sBLIMP benchmark [214] measures syntactic competence by evaluating models' ability to distinguish between grammatically correct and incorrect spoken utterances, providing crucial insights into the model's understanding of language structure across modalities. Similarly, sWuggy [214] assesses phonological processing by testing discrimination between real words and phonologically plausible non-words in the speech domain.

For ASR capabilities, models are typically evaluated on LibriSpeech [223], which offers both "clean" and "other" (more challenging) test sets to measure transcription accuracy across varying acoustic conditions. This provides a standardized measure of a model's ability to convert speech to text accurately. CommonVoice [9] serves as another critical ASR benchmark, offering evaluation across 100+ languages with diverse accents and recording conditions, making it particularly valuable for assessing multilingual and cross-dialect performance.

The sStoryCloze and Topic-sStoryCloze benchmarks introduced in TWIST [116] assess semantic understanding and common sense reasoning through continuation tasks, while tStoryCloze evaluates similar capabilities in the text domain. These evaluations span mul-

tiple modality combinations: Speech-to-Speech, Text-to-Speech, Speech-to-Text, and Text-to-Text. This comprehensive evaluation framework ensures models maintain coherence and understanding across modality transitions.

This multi-faceted evaluation approach provides a holistic assessment of models' abilities to process, understand, and generate content across modalities, ensuring they meet the requirements for practical applications in diverse real-world scenarios.

0.6.5 Downstream Tasks Metrics

Standard metrics for downstream tasks provide quantitative measures of model performance across different capabilities. For ASR tasks, *Word Error Rate* (WER) measures the percentage of words incorrectly transcribed, with lower values indicating better performance. *Character Error Rate* (CER) functions similarly to WER but at the character level, proving useful for languages without clear word boundaries.

$$\text{WER} = \frac{S + D + I}{N} \times 100$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions, and N is the number of words in the reference. Similarly, CER uses the same formula but operates at the character level rather than the word level.

For linguistic competence tasks, accuracy serves as the primary metric for sBLIMP and sWuggy, where models are evaluated on their ability to correctly distinguish between minimal pairs. The reasoning tasks in sStoryCloze and its variants typically report accuracy in selecting the correct continuation. For *Spoken Question Answering* (SQA) tasks, accuracy measures how often the model provides the correct answer to questions presented in spoken or textual form.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \times 100$$

Negative Log Likelihood (NLL) provides a probabilistic assessment of model predictions, measuring how confidently the model assigns probability to correct outputs. Lower NLL values indicate that the model assigns higher probability to the ground truth, suggesting stronger predictive performance.

$$\mathcal{L} = -\sum_{t=1}^{T} \log p(x_t | x_{<t}) \tag{11}$$

where $p(x_t|x_{<t})$ represents the probability of token x_t given all previous tokens $x_{<t}$ in the sequence, and T is the total sequence length. This metric is particularly valuable for evaluating generative models and their ability to accurately predict next tokens in a sequence.

Part II

Masked Language Models

Chapter 1

DrBERT: A Robust Pre-trained Model in French for Biomedical and Clinical domains

As we described in the previous chapters, language modeling has become a fundamental component in the making of state-of-the-art task-specific models in NLP [230, 81]. More specifically, large-scale MLM based on transformer architecture [275] and trained on large raw text corpora have impressively extended the performance of NLP models on most tasks.

However, while these models have shown remarkable success in general domains, recent works have demonstrated that optimal performance in specialized domains, such as finance, medical, or travel, can only be achieved using PLM adapted to the targeted conditions. This is particularly challenging for languages other than English, where domain-specific data are generally difficult to obtain, resulting in quite a few specialized PLMs available.

Several key research questions arise when considering domain adaptation of language models:

- What is the optimal trade-off between using healthcare related data from publicly accessible internet sources (which often differ significantly in structure and complexity from real-world applications) versus using data collected from private data warehouses when considering domain adaptation?
- How important is the variety of data sources compared to the volume of data?
- Can low-resource domain-specific data outperform large general-domain models?
- Is cross-lingual transfer effective for domain adaptation?

This chapter presents our work on developing and evaluating the first biomedical and clinical transformer-based language models for French healthcare domains, based on the

RoBERTa architecture [193]. This work, which received an Honorable Mention at ACL 2023 [165], addresses several of the aforementioned research questions through extensive experimentation and analysis. Our main contributions are:

- We demonstrate that pre-training on constrained resources (4GB) of web-crawled medical data can compete with, and often outperform, models trained on specialized clinical data.
- We show that continual pre-training of English domain-specific models on French data is more effective than adapting French general-domain models.
- We release DrBERT[1], the first open-source French biomedical language model, along with NACHOS[2], a large French medical corpus. To ensure reproducibility and facilitate future research, we also make available all the code[3] for training, preprocessing and high-scale distributed learning.

Our results demonstrate that pre-training on constrained resources of web-crawled medical data can compete with, and even frequently surpass, models trained with specialized data from medical reports. We also show that continual pre-training on an existing domain-specific English model (PubMedBERT) is a more viable solution than on a French domain-generalist model when targeting French biomedical downstream tasks.

1.1 Pre-Training Datasets

In the biomedical domain, previous works [109] on PLMs highlighted the importance of matching the data sources used for its training to the targeted downstream tasks. Due to their sensitive nature (protection of user data, protected health information of patients, etc.), medical data are extremely difficult to obtain. Massive collection of web data related to this domain appears to be a solution that can overcome this lack. However, these web documents vary in terms of quality. No comparison has been made between PLMs based on specific domain data from the web and those on private documents from clinical data warehouses, whose quality can be controlled.

We extracted two different medical datasets for French. In the first one, I gathered data crawled from a variety of free-of-use online sources, and the second one, collected by Adrien, gather private hospital stays reports from the Nantes University Hospital.

Table 1.1 gives a general overview of the two collected corpora. The public web-based data, detailed in Section 1.1.1, allowed the constitution of a corpus, called NACHOS$_{large}$, containing 7.4 GB of data. The private dataset, called NBDW$_{small}$ is described in Section 1.1.2 and contains 4 GB of data. In order to perform comparable experiments, we extracted a NACHOS sub-corpus (NACHOS$_{small}$) of the same size as the private data. Finally, Section 1.1.3 describes the pre-processing applied to both datasets.

[1] https://huggingface.co/Dr-BERT/DrBERT-7GB
[2] https://huggingface.co/datasets/Dr-BERT/NACHOS
[3] https://github.com/qanastek/DrBERT

Corpus	Size	#words	#sentences
NACHOS$_{large}$ (public)	7.4 GB	1.1 B	54.2 M
NACHOS$_{small}$ (public)	4 GB	646 M	25.3 M
NBDW$_{small}$ (private)	4 GB	655 M	43.1 M
NBDW$_{mixed}$ (both)	4+4 GB	1.3 B	68.4 M

Table 1.1: Overview of the public (NACHOS) and private (NBDW) collected datasets.

1.1.1 Public corpus - NACHOS

We collected the *openN crAwled frenCh Healthcare cOrpuS* (NACHOS), a French medical open-source dataset compiled by crawling a variety of textual sources around the medical topic. It consists of more than one billion words, drawn from 24 French-speaking high-quality websites. The corpus includes a wide range of medical information: descriptions of diseases and conditions, information on treatments and medications, general health-related advice, official scientific meeting reports, anonymized clinical cases, scientific literature, thesis, French translation pairs, university health courses and a large range of data obtained from raw textual sources, web scrapping, and *optical character recognition* (OCR). Table 1.2 summarizes the different data sources of NACHOS.

We use heuristics to split the texts into sentences and aggressively filter out short or low-quality sentences like those obtained from OCR. Finally, we classified them into languages by using our own classifier trained on the multilingual Opus EMEA [263] and MASSIVE [91] corpora to keep only the sentences in French.

For the 4 GB version of NACHOS (NACHOS$_{small}$), we shuffled the whole corpus and selected randomly 25.3M sentences to maximize data source homogeneity. The full NACHOS corpus is now freely available online[4].

[4] https://drbert.univ-avignon.fr/

Resource name	# words
HAL	638,508,261
Haute Autorité de Santé (HAS)	113,394,539
Drug leaflets	74,770,229
Medical Websites Scrapping	64,904,334
ANSES SAISINE	51,372,932
Public Drug Database (BDPM)	48,302,695
ISTEX	44,124,422
CRTT	26,210,756
WMT-16	10,282,494
EMEA-V3	6,601,617
Wikipedia Life Science French	4,671,944
ANSES RCP	2,953,045
Cerimes	1,717,552
LiSSa	235,838
DEFT-2020	231,396
CLEAR	225,898
CNEDiMTS	175,416
QUAERO French Medical Corpus	72,031
ANSM Clinical Study Registry	47,678
ECDC	44,482
QualiScope	12,718
WMT-18-Medline	7,673
Total	**1,088,867,950**

Table 1.2: Word-count distribution across the 22 sources of the NACHOS biomedical corpus.

1.1.2 Private corpus - NBDW

The private corpus, called Nantes Biomedical Data Warehouse (NBDW), was obtained using the data warehouse from Nantes University Hospital. This data warehouse includes different dimensions of patients' related data: socio-demographic, drug prescriptions and other information associated with consultation or hospital stays (diagnosis, biology, imagery, etc.). The authorization to implement and exploit the NBDW dataset was granted in 2018 by the CNIL (*Commission National de l'Informatique et des Libertés*), the French independent supervisory authority in charge of application of national and European data privacy protection laws; authorization N°2129203.

For this work, a sample of 1.7 million de-identified hospital stays reports was randomly selected and extracted from the data warehouse. As described in Table 1.3, the reports are from various hospital departments, emergency medicine, gynecology and ambulatory care being the most frequent.

Each of the reports was split into tokens sequence with an average of 15.26 words per sequence. Then, all tokens sequences from all reports were shuffled to build the corpus. This corpus contains 655M words, from 43.1M sentences, for a total size of approximately 4 GB.

Medical Specialty	# documents	# words
Other	474,588	192,832,792
Emergency Medicine	235,579	90,807,406
Ambulatory Care	119,149	50,975,472
Consultation	95,135	38,335,804
Gynecology	132,983	38,204,495
Cardiology	29,633	22,654,583
Medical Oncology	45,603	22,587,869
Gastroenterology	46,600	21,340,794
Orthopaedic Surgery	82,084	18,983,791
Hematology	41,776	18,285,983
Critical Care Medicine	20,819	16,472,785
Otolaryngology	69,343	16,131,214
Dermatology	51,804	15,035,412
Rheumatology	31,527	14,647,543
Urology	51,535	14,272,231
Colon and Rectal Surgery	45,987	13,334,550
Internal Medicine	23,904	13,282,253
Psychiatry	26,628	12,496,503
Neurosurgery	34,481	10,360,533
Nephrology	19,171	9,548,533
Ophthalmology	19,700	4,464,515
Total	**1,698,029**	**655,055,061**

Table 1.3: Number of documents and total word counts by medical specialty in the NBDW corpus.

1.1.3 Pre-processing steps

The supplied text data has been split into subword units using SentencePiece [159], a language-independent subword tokenizer that does not require pre-tokenization, thereby avoiding the requirement for language-specific tokenizers. Unlike traditional approaches like Moses that rely on hand-crafted language-dependent rules, SentencePiece enables purely end-to-end text processing. We employ a vocabulary size of 32k subword tokens. For each model pre-trained from scratch (see Section 1.2.2), tokenizers were built using all the sentences from the pre-training dataset.

SentencePiece supports two distinct modes: BPE [249] and unigram language model [157]. Algorithm 1 outlines the core tokenization process, which begins with text normalization and treats the input as a sequence of Unicode characters, replacing whitespace with a special meta symbol "\" (U+2581).

Algorithm 1 SentencePiece Tokenization

Input: Text T, Vocabulary V, Model type $M \in \{\text{BPE}, \text{Unigram}\}$
Output: Tokenized sequence S
procedure Preprocess(T)
    Normalize Unicode characters with NFKC
    Replace whitespace with special token _ (U+2581)
    Treat input as sequence of Unicode characters
    return Preprocessed text T'
end procedure
if $M = \text{BPE}$ **then**
    $S \leftarrow \text{BPETokenize}(T', V)$
else if $M = \text{Unigram}$ **then**
    $S \leftarrow \text{UnigramTokenize}(T', V)$
end if
return S

The BPE variant (described in Algorithm 2) operates by iteratively merging the most frequent adjacent character pairs. SentencePiece implements an optimized $\mathcal{O}(N \log N)$ algorithm using binary heaps to manage merged symbols efficiently, compared to the $\mathcal{O}(N^2)$ complexity of naive implementations.

Algorithm 2 BPE Tokenization in SentencePiece

procedure BPETokenize(T', V)
    Initialize S as character sequence of T'
    while possible merges exist **do**
        Find most frequent adjacent token pair (a, b) in S using priority queue
        Replace all occurrences of (a, b) with merged token ab
        Add ab to vocabulary if not present
        if vocabulary size $= |V|$ **then**
            break
        end if
    end while
    return S
end procedure

The unigram language model (described in Algorithm 3) employs a probabilistic approach, finding the most likely segmentation using the Viterbi algorithm. This method enables multiple segmentation candidates during training, making it suitable for subword regularization.

Algorithm 3 Unigram Tokenization in SentencePiece

procedure UnigramTokenize(T', V)
  Initialize language model P with probabilities for tokens in V
  $S \leftarrow \emptyset$
  $x \leftarrow T'$
  while $x \neq \emptyset$ **do**
    $(s, x') \leftarrow \arg\max (s, x') : x = s + x'P(s)$ ▷ Find best token using Viterbi
    $S \leftarrow S \cup \{s\}$
    $x \leftarrow x'$
  end while
  return S
end procedure

For training the unigram model, SentencePiece employs a normalized likelihood-based pruning method (described in Algorithm 4) to iteratively reduce the vocabulary to the target size, typically removing tokens with lowest loss at each iteration.

Algorithm 4 Normalized Likelihood Pruning for Unigram Model

Input: Vocabulary V, Corpus C, Current model parameters θ, Pruning rate p (typically 20%)
Output: Reduced vocabulary
for each token $t \in V$ **do**
  Compute loss if token t is removed:
  $loss(t) = \sum_{x \in C} (\log P(x|\theta) - \log P(x|\theta \setminus t))$
  Normalize loss: $loss_{norm}(t) = \frac{loss(t)}{freq(t)}$
end for
Sort tokens by $loss_{norm}(t)$ in ascending order
Remove bottom $p\%$ of tokens from V
Re-estimate probabilities for remaining tokens
return Updated vocabulary V

A key advantage of SentencePiece is its lossless tokenization design, which preserves all information needed to perfectly reconstruct the original text. This is achieved by treating whitespace as a normal symbol (escaped with \) rather than as a boundary marker. The model is self-contained, with all normalization rules and parameters embedded in the model file, ensuring perfect reproducibility across environments.

In our implementation, we used the unigram model which has been shown to achieve superior performance for morphologically rich languages while maintaining competitive results across other language families. Experimental results reported by Kudo and Richardson demonstrate that SentencePiece can achieve comparable or better BLEU scores even without pre-tokenization, while providing significantly faster processing, especially for non-segmented languages.

1.2 Models Pre-Training

In this section, we describe the pre-training modalities of our studied models from two points of view: 1) the influence of the data used (size and nature), and 2) the pre-training strategies of the models. These two levels are respectively detailed in Sections 1.2.1 and 1.2.2. Section 1.2.3 finally presents the existing state-of-the-art pre-trained models that will be used for comparison purposes.

1.2.1 Influence of data

One issue is to identify the amount of data required to create a model that performs well and can compete with models trained on general domains. Recent studies, such as those by [314] and [200], discuss the impact of the size of pre-training data on model performance. According to these studies, some tasks are performing better with fewer data while others, such as commonsense knowledge and reasoning tasks, keep improving performance when pre-training data is added.

In the medical field, no study has been conducted to compare the impact of varying the amount of domain-specific data during pre-training, or to assess the impact of the supposedly variable quality of the data depending on their source of collection.

We thus propose to evaluate the pre-training of several language models on either NACHOS$_{small}$ or NBDW$_{small}$ corpus, as described in Section 1.1. Additionally, we propose a model pre-trained on NACHOS$_{large}$ to investigate if having almost twice as much data improves model performance. Finally, a combination of both public NACHOS$_{small}$ and NBDW$_{small}$ sources for a total of 8 GB (NBDW$_{mixed}$) is explored to demonstrate if combining private and public data is a viable approach in low-resource domains.

1.2.2 Pre-training strategies

In addition to the analysis on the size and the sources of data, we also seek to evaluate three training strategies of PLMs for the medical domain:

- Training a full model from scratch, including the subword tokenizer.
- Continuing the pre-training of the state-of-the-art language model for French, called CamemBERT, on our medical-specific data while keeping the initial tokenizer.
- Continuing the pre-training of a state-of-the-art domain specific language model for medical but here in English, called PubMedBERT, on our French data while keeping the initial tokenizer.

Regarding the last strategy, our objective is to compare the performance of an English medical model further pre-trained on our French medical data, against another one based on

a generic French model. Indeed, the medical domains shares many terms across languages that make relevant the mixture of resources from two languages.

Table 1.4 summarizes all the configurations evaluated in this chapter, integrating both the study of data size and pre-training strategies.

Model name	Strategy	Corpus
DrBERT	From scratch	NACHOS$_{large}$
DrBERT	From scratch	NACHOS$_{small}$
ChuBERT	From scratch	NBDW$_{small}$
ChuBERT	From scratch	NBDW$_{mixed}$
CamemBERT	continual pre-training	NACHOS$_{small}$
PubMedBERT	continual pre-training	NACHOS$_{small}$
CamemBERT	continual pre-training	NBDW$_{small}$

Table 1.4: List of studied pre-trained model configurations.

Model architecture All models pre-trained from scratch use the CamemBERT $_{\text{base}}$ configuration, which is the same as RoBERTa $_{\text{base}}$ architecture (12 layers, 768 hidden dimensions, 12 attention heads, 110M parameters). We did not train the large version of our models due to resource limitations.

Language modeling We train the models on the Masked Language Modeling (MLM) task using HuggingFace library [291]. It consists of randomly replacing a subset of tokens from the sequence with a special token and asking the model to predict them using cross-entropy loss. In BERT and RoBERTa models (including CamemBERT), 15% of the tokens are randomly selected. Of those selected tokens, 80% are replaced with the `<mask>` token, 10% remain unchanged, and 10% are randomly replaced by a token from the vocabulary. We keep this masking probability of 15% for the training of our models.

Optimization & Pre-training We optimize the models for 80k steps with batch sizes of 4,096 sequences, each sequence filled with 512 tokens, allowing us to process 2.1M tokens per step. The learning rate is warmed up linearly for 10k steps, going up from zero to the initial 5×10^{-5} learning rate. Models are trained on 128 Nvidia V100 32 GB GPUs for 20 hours on Jean Zay supercomputer. We use mixed precision training (FP16) [203] to reduce the memory footprint, allowing us to enlarge the batch size to 32 sequences on each GPU.

1.2.3 Baseline models

We describe some existing pre-trained models used as baselines in our comparative study.

CamemBERT [200] is a RoBERTa-based model pre-trained totally from scratch on the French subset of the OSCAR corpus (138 GB). In our case, this model is our main baseline to compare our results on, since it is the state-of-the-art model for French. We also use the 4 GB model's variants of CamemBERT to compare the impact of the nature and quantity of the data.

PubMedBERT [109] is a BERT-based biomedical-specific model pre-trained totally from scratch on the 3.1 billion words of PubMed corpus (21 GB).

ClinicalBERT [130] is a clinical-specific model based on BERT tokenizer and weights, which has been further pre-trained on the 0.5 billion words of MIMIC corpus (3.7 GB).

BioBERT v1.1 [176] is a biomedical-specific model based on BERT tokenizer and weights which has been further pre-trained using the 4.5 billion words of PubMed corpus.

1.3 Downstream Evaluation Tasks

Thematic / Corpus name	Task	Metric	Train	Dev	Test
Public Corpus					
ESSAIS [73]	POS Tagging	Macro F1	9,693	2,077	2,078
CAS: French Corpus with Clinical Cases [105]	POS Tagging	Macro F1	5,306	1,137	1,137
MUSCA-DET - Social Determinants of Health extraction (Task 1)	Nested NER	Macro F1	19,861	2,207	5,518
MUSCA-DET - Social Determinants of Health extraction (Task 2)	Multi-label Classification	Macro F1	19,861	2,207	5,518
QUAERO French Medical Corpus - EMEA [211]	Nested NER	Weighted F1	11	12	15
QUAERO French Medical Corpus - MEDLINE [211]	Nested NER	Weighted F1	833	832	833
FrenchMedMCQA [164]	MCQA	EMR / Hamming Score	2,171	312	622
Private Corpus					
Medical report acute heart failure structuration	Named Entity Recognition	Macro F1	2,527	281	703
Acute heart failure (aHF) classification	Binary Classification	Macro F1	1,179	132	328
Technical Specialties Sorting	Classification Multi-class	Macro F1	4,413	1,470	1,473
Medical report structuration prescriptions	Named Entity Recognition	Macro F1	61	15	26

Table 1.5: Corpus, tasks and metrics synthesis for evaluating medical-specific models.

To evaluate the different pre-training configurations of our models, a set of tasks in the medical domain is necessary. While this NLP domain-specific benchmark exists in English (BLURB [109]), none existed for French at the time we started working on this project. In this section, we describe an original benchmark, summarized in Table 1.5, integrating various NLP medical tasks for French. Among them, some are from publicly-available datasets (Section 1.3.1), allowing the replication of our experiments. Other tasks come from private datasets (Section 1.3.2) and cannot be shared. However, they are useful to evaluate our models more accurately.

1.3.1 Publicly-available tasks

The publicly available tasks used in these experiments are presented below.

ESSAIS / CAS: French Corpus with Clinical Cases The ESSAIS [73] and CAS [105] corpora respectively contain 13,848 and 7,580 clinical cases in French. Some clinical cases are associated with discussions. A subset of the whole set of cases is enriched with morpho-syntactic (POS tagging, lemmatization) and semantic (UMLS concepts, negation, uncertainty) annotations. In our case, we focus only on the POS tagging task.

FrenchMedMCQA The FrenchMedMCQA corpus [164] is a publicly available *Multiple-Choice Question Answering* (MCQA) dataset in French for the medical domain. It contains 3,105 questions coming from real exams of the French medical specialization diploma in pharmacy, integrating single and multiple answers.

QUAERO French Medical Corpus The QUAERO French Medical Corpus [211] introduces an extensive corpus of biomedical documents annotated at the entity and concept levels to provide NER and classification tasks. Three text genres are covered, comprising a total of 103,056 words obtained either from EMEA or MEDLINE. Ten entity categories corresponding to UMLS [32] Semantic Groups were annotated, using automatic pre-annotations validated by trained human annotators. Overall, a total of 26,409 entity annotations were mapped to 5,797 unique UMLS concepts. To simplify the evaluation process, we sort the nested labels in alphabetical order and concatenate them together into a single one to transform the task into a usable format for token classification with BERT based architectures.

MUSCA-DET MUSCA-DET is a French corpus of sentences extracted from the "Lifestyle" section in clinical notes from Nantes University Hospital biomedical data warehouse. The corpus contains 27,000 pseudonymized sentences annotated with 26 entities related to Social Determinants of Health (living, marital status, housing, descendants, employment, alcohol, smoking, drug abuse, physical activity). The corpus includes two tasks: nested NER and multi-label classification.

1.3.2 Private tasks

MUSCA-DET MUSCA-DET is a French corpus of sentences extracted from the "Lifestyle" section in clinical notes from Nantes University Hospital biomedical data warehouse. The corpus contains 27,000 pseudonymized sentences annotated with 26 entities related to Social Determinants of Health (living, marital status, housing, descendants, employment, alcohol, smoking, drug abuse, physical activity). The corpus includes two tasks: nested NER and multi-label classification.

Technical Specialties Sorting This classification task has to assign the specialty of a medical report based on its transcription. The dataset consists of 7,356 French medical reports that have been manually annotated and equally sampled across 6 specialties: Psychiatry, Urology, Endocrinology, Cardiology, Diabetology, and Infectiology.

Medical report structuration prescriptions (NER) The task seeks to identify named entities in a gold sample of 100 long medical reports obtained from French speech transcriptions. The named entities are annotated using the BIO format and fall into 12 classes: *O*, *AGE*, *CITY*, *DATE*, *EMAIL*, *HOSPITAL*, *PHONE*, *DOSAGE*, *DURATION*, *FORM*, *MEDICATION* and *POSOLOGY*.

Medical report acute heart failure structuration (NER) This corpus contains 350 hospital stay reports (divided into 3,511 sentences) from Nantes University Hospital. The reports are annotated with 46 entity types related to the following clinical information: cause of chronic heart failure, triggering factor for acute heart failure, diabetes, smoking status, heart rate, blood pressure, weight, height, medical treatment, hypertension and left ventricular ejection fraction. Overall, the corpus contains 6,116 clinical entities.

Acute heart failure (aHF) classification This task consists of the classification of hospital stays reports according to the presence or absence of a diagnostic of acute heart failure. This corpus consists of 1,639 hospital stays reports from Nantes university hospital, which are labeled as positive or negative to acute heart failure.

1.4 Results and Discussions

As previously described, we evaluate the performance of our pre-trained language models proposed for the biomedical domain on a set of public and private NLP downstream tasks related to the medical domain. We first propose to analyze the results according to the different pre-training strategies used (Section 1.4.1) then to focus on the impact of the pre-training data, whether in terms of size or nature (Section 1.4.2). Finally, we are interested in the generalization capacities of our domain-specific models by applying and comparing them on general domain NLP tasks (Section 1.4.3).

Note that all the PLMs have been fine-tuned in the same way for all downstream tasks and all the reported results are obtained by averaging the scores from four runs. Performance on biomedical downstream tasks is reported in Tables 1.7 and 1.6 for respectively private and public tasks. For readability reasons, the first part of each table presents the existing baseline model results, the second part our specialized models trained from scratch, and the last part our models using continual pre-training.

	MUSCA-DET T1			MUSCA-DET T2			ESSAI POS			CAS POS			FrenchMedMCQA		QUAERO-EMEA			QUAERO-MEDLINE		
	P	*R*	*F1*	*P*	*R*	*F1*	*P*	*R*	*F1*	*P*	*R*	*F1*	*Hamming*	*EMR*	*P*	*R*	*F1*	*P*	*R*	*F1*
CamemBERT OSCAR 138 GB	89.04	88.59	88.54	89.87	87.12	88.20	81.57	81.01	81.10	96.37	94.53	95.22	36.24	**16.55**	90.57	91.06	90.71	76.58	78.67	77.41
CamemBERT OSCAR 4 GB	86.09	85.45	85.43	92.68	90.34	91.27	84.01	83.51	83.69	98.15	95.34	96.42	35.75	15.37	90.75	91.16	90.83	78.55	79.33	78.76
CamemBERT CCNET 4 GB	91.12	89.91	90.33	93.10	90.42	91.38	85.60	85.63	85.42	**98.19**	**96.75**	**97.33**	34.71	14.41	90.31	90.59	90.33	78.06	78.11	77.61
PubMedBERT	93.04	91.45	91.99	84.41	80.60	81.97	88.43	87.93	87.78	97.40	94.86	95.90	33.98	14.14	86.89	87.33	86.79	77.33	77.28	77.09
ClinicalBERT	91.79	89.44	90.36	85.43	81.23	82.95	89.09	88.78	88.24	97.94	95.88	96.73	32.78	14.19	84.91	85.47	84.79	75.56	74.85	75.05
BioBERT 1.1	91.82	89.82	90.46	85.52	80.14	81.91	86.76	84.90	85.18	98.10	96.39	97.12	36.19	15.43	84.55	85.03	84.29	72.62	73.30	72.68
DrBERT NACHOS$_{large}$	92.10	90.27	91.04	**94.97**	90.41	92.24	**90.96**	**89.19**	**89.75**	97.37	94.49	95.65	36.66	15.32	**91.93**	**92.52**	**92.09**	77.85	78.54	77.88
DrBERT NACHOS$_{small}$	93.35	90.62	91.77	91.31	86.60	88.57	90.12	88.37	88.76	97.04	94.88	95.70	**37.37**	13.34	91.54	92.00	91.66	77.91	79.34	78.18
ChuBERT NBDW$_{small}$	**94.88**	90.79	92.23	94.77	90.27	92.17	88.53	87.73	87.71	97.00	94.65	95.61	35.16	14.79	88.11	88.78	88.15	75.05	76.57	74.94
ChuBERT NBDW$_{mixed}$	94.39	**91.93**	**92.73**	94.22	90.02	91.71	86.36	85.50	85.73	97.77	95.30	96.35	34.58	12.21	90.36	90.94	90.52	78.61	79.32	78.63
CamemBERT NACHOS$_{small}$	81.44	81.39	80.96	79.74	78.08	78.70	80.59	79.88	80.04	95.64	91.57	92.46	32.87	13.76	67.56	77.48	71.10	55.45	62.34	57.43
PubMedBERT NACHOS$_{small}$	92.51	91.49	91.53	94.95	**92.55**	**93.62**	84.73	83.80	83.85	97.82	96.12	96.81	35.88	15.21	90.97	91.27	91.03	**82.03**	**81.71**	**81.73**
CamemBERT NBDW$_{small}$	82.35	81.59	81.57	78.14	76.38	77.12	79.44	79.79	79.25	95.98	92.11	93.18	27.73	11.89	53.44	73.11	61.75	48.71	61.33	53.05

Table 1.6: Performance on public biomedical downstream tasks. Best model in bold and second is underlined.

1.4.1 Impact of pre-training strategies

As observed both in Tables 1.7 and 1.6, models pre-trained completely from scratch (DrBERT NACHOS and ChuBERT NBDW) tend to produce the best results for both types of data sources and tasks (*i.e.* private and public). Indeed, considering the F1-score, they obtain the best results on all private tasks and on almost all public ones (5 tasks out of 7). The two public remaining tasks (MUSCA-DET T2 and QUAERO-MEDLINE) are then better handled using PubMedBERT NACHOS$_{small}$, a model that has already been pre-trained on domain-specific data (biomedical English data) then further pre-trained with our French medical data (NACHOS$_{small}$).

We also observed that continual pre-training from domain generic models (CamemBERT NACHOS$_{small}$ or CamemBERT NBDW$_{small}$) does not allow reaching the performance of the other specific models, neither of these two models reaching the first or second place (in terms of performance) on any task.

Finally, the baseline models trained on generic data (CamemBERT OSCAR) and those trained on biomedical data in English (PubMedBERT, ClinicalBERT and BioBERT) remain competitive in few biomedical public tasks (CAS POS, FrenchMCQA or MUSCA-DET T2), while none of them are placed in first or second place on private tasks. This seems to highlight the difficulty of private tasks when non-matching data are used.

1.4.2 Effect of data

Regarding the amount of data used for pre-training models ($small$ vs. $large$ or $mixed$), results show that, the larger the data are, the better the model performs, no matter the pre-training strategy or the source of data (private or public). However, the difference is very low for most tasks, with $small$ systems often being ranked second behind large models,

even though they contain half as much data.

	aHF NER			aHF classification			NER Medical Report			Specialities Classification		
	P	***R***	***F1***	***P***	***R***	***F1***	***P***	***R***	***F1***	***P***	***R***	***F1***
CamemBERT OSCAR 138 GB	40.89	35.22	35.13	81.90	79.12	80.13	87.98	91.66	89.35	99.32	99.09	99.20
CamemBERT OSCAR 4 GB	46.32	43.17	42.66	81.49	81.42	81.41	87.79	90.74	88.78	99.53	99.69	99.61
CamemBERT CCNET 4 GB	47.25	42.2	43.11	82.02	79.30	79.98	87.61	92.28	89.34	99.54	99.55	99.55
PubMedBERT	52.61	46.30	47.22	78.17	76.18	76.86	87.07	92.61	89.20	99.25	99.51	99.37
ClinicalBERT	50.11	44.15	44.70	80.13	75.92	77.12	87.04	92.14	88.77	98.58	98.62	98.58
BioBERT v1.1	49.37	47.25	46.01	79.69	78.51	79.00	**88.17**	91.80	89.38	98.59	99.03	98.80
DrBERT NACHOS$_{large}$	55.29	46.66	48.22	81.33	81.25	81.25	87.99	**92.80**	**89.83**	99.82	**99.90**	**99.86**
DrBERT NACHOS$_{small}$	54.55	43.39	45.93	79.85	80.10	79.87	87.57	92.76	89.44	**99.85**	99.85	99.85
ChuBERT NBDW$_{small}$	**56.92**	47.46	49.01	81.03	**82.67**	81.56	87.76	92.63	89.58	99.76	**99.90**	99.83
ChuBERT NBDW$_{mixed}$	54.62	47.81	**49.14**	82.23	81.71	**81.98**	87.42	92.36	89.30	99.81	99.82	99.81
CamemBERT NACHOS$_{small}$	22.02	16.67	16.08	74.86	69.82	69.80	65.72	68.49	66.74	99.44	99.67	99.54
PubMedBERT NACHOS$_{small}$	53.44	**48.21**	48.72	**83.06**	80.39	81.40	87.35	92.69	89.36	99.52	99.58	99.55
CamemBERT NBDW$_{small}$	25.44	19.33	19.12	79.50	74.74	76.02	68.80	71.23	69.64	99.60	99.57	99.58

Table 1.7: Performance on our private biomedical downstream tasks. Best model in bold and second is underlined.

We notice a clear dominance of models that were pre-trained on web-based sources, specifically OSCAR and NACHOS, when applied to public tasks. Indeed, models relying on private NBDW data only achieve the best performance (in terms of F1-score) on the MUSCA-DET T1 task. This trend is not quite observed on private tasks, where NBDW-based models obtain more acceptable or even better performance when mixed with public biomedical data (ChuBERT NBDW$_{mixed}$), as seen in Table 1.7. We believe this discrepancy is mainly due to the different nature of processed data.

Finally, we observe that English-based models perform closely to the French-based CamemBERT model. This shows the usefulness of pre-training on domain specific data. For example, better results are obtained with continual pre-training of the PubMedBERT model with our specialized data in French (PubMedBERT NACHOS$_{small}$), corroborating our hypothesis about the effectiveness of cross-language knowledge transfer.

1.4.3 Performance on general-domain tasks

Table 1.8 gives the results obtained by all PLMs on general domain downstream tasks. These tasks come from [200] who used them to evaluate the CamemBERT model. The first four are POS tagging tasks (GSD, SEQUOIA, SPOKEN and PARTUT), the last being a natural language inference task (XNLI).

	GSD	SEQUOIA	SPOKEN	PARTUT	XNLI
CamemBERT OSCAR 138 GB	**98.28**	98.68	97.26	97.70	**81.94**
CamemBERT OSCAR 4 GB	98.14	**99.18**	**97.57**	97.86	81.76
CamemBERT CCNET 4 GB	98.18	98.92	97.20	**97.92**	81.26
PubMedBERT	96.48	96.49	90.00	93.97	73.79
ClinicalBERT	96.49	96.31	89.60	93.17	70.57
BioBERT v1.1	97.32	96.54	91.81	94.52	71.54
DrBERT NACHOS$_{large}$	96.94	98.05	95.92	96.54	72.18
DrBERT NACHOS$_{small}$	97.17	98.21	96.38	96.45	72.86
ChuBERT NBDW$_{small}$	96.45	97.38	94.90	95.83	69.00
ChuBERT NBDW$_{mixed}$	97.18	98.10	96.43	96.33	72.32
CamemBERT NACHOS$_{small}$	97.63	96.90	91.12	94.00	71.26
PubMedBERT NACHOS$_{small}$	97.41	98.71	95.54	97.01	77.35
CamemBERT NBDW$_{small}$	97.55	96.26	89.17	91.34	72.73

Table 1.8: Performance on public domain-general downstream tasks. Best model in bold and second is underlined.

All results of our models decrease in performance on all tasks. The most important drop is for the natural language inference task, with a performance of ChuBERT NBDW$_{small}$ almost 13% lower than CamemBERT 138 GB. We also observe that the specialized models in English are as efficient as our biomedical models in French. It seems quite clear from the previous observations that specialized models are difficult to generalize to other tasks, but that specialized information captured in one language could transfer to another language.

1.4.4 Vocabularies Inter-coverage

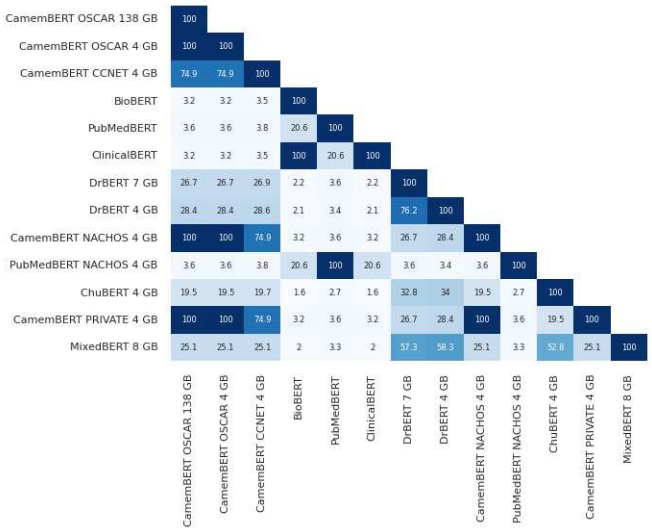
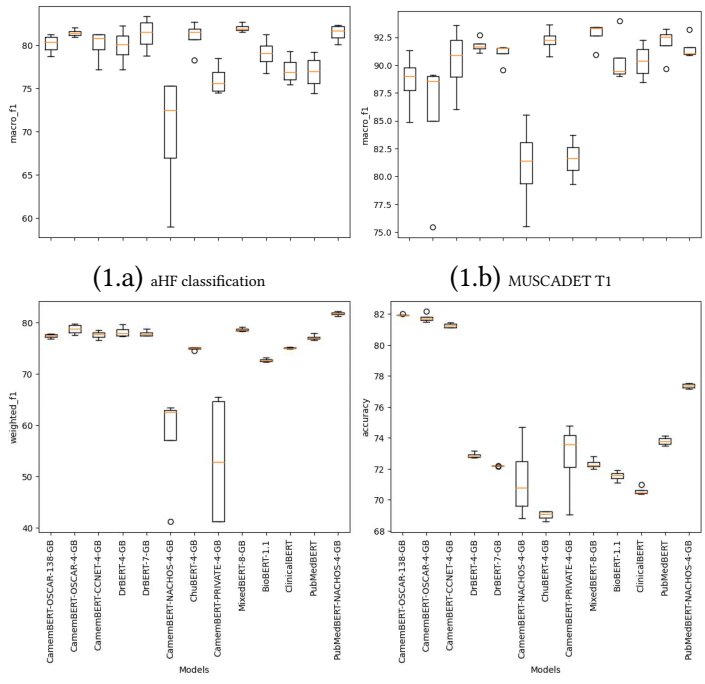


Figure 1.1: Vocabularies inter-coverage matrix for MLM models, showing pairwise percentage overlap in token vocabularies. Darker cells indicate higher shared coverage.

As we can see in Figure 1.1, the vocabulary inter-coverage matrix reveals interesting patterns in the shared vocabulary between different BERT-based models. CamemBERT models (138 GB, 4 GB, and CCNET 4 GB) show strong mutual vocabulary coverage (74.9-100%), indicating significant overlap in their tokenization despite different training data sizes. However, domain-specific models like BioBERT, PubMedBERT, and ClinicalBERT share relatively low vocabulary coverage (around 32-38%) with general-domain models, highlighting their specialized nature. DrBERT variants (7 GB and 4 GB) demonstrate moderate coverage (20-30%) with other models but maintain strong internal consistency (76.2-100%). Notably, the NACHOS variants of CamemBERT and PubMedBERT show distinct patterns, with CamemBERT NACHOS maintaining higher coverage with its parent model while PubMedBERT NACHOS shows more divergence. The MixedBERT 8 GB model exhibits particularly low coverage (2-33%) with most other models except DrBERT variants (57.3-58.3%), suggesting a unique vocabulary composition that reflects its mixed-domain training approach.

1.4.5 Models Stability

We observe during the evaluation phase that most of the models based on continual pre-training strategy from CamemBERT OSCAR 138 GB are suffering from bad consistency and stability during fine-tuning, which translates into fluctuation in performance between runs as shown in the Figure 1.2.



(1.a) aHF classification (1.b) MUSCADET T1

(2.a) QUAERO MEDLINE (2.b) XNLI

Figure 1.2: Box plot of the F1 score for each of the tasks and models.

We also notice during PubMedBERT NACHOS$_{small}$ pre-training that the model loss (Figure 1.3) is globally stable during almost all the duration of the pre-training, until reaching

the step 71,000, where the loss fall down until touching down zero at step 72,500.

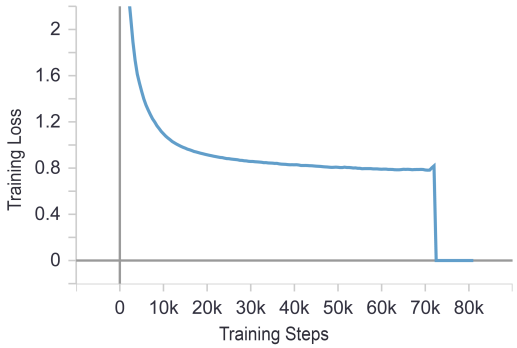


Figure 1.3: PubMedBERT NACHOS$_{small}$ loss.

While the DrBERT 4GB and 7GB variants trained from-scratch are showing a consistent loss decrease without any abnormal phenomenon as shown in Figure 1.4:

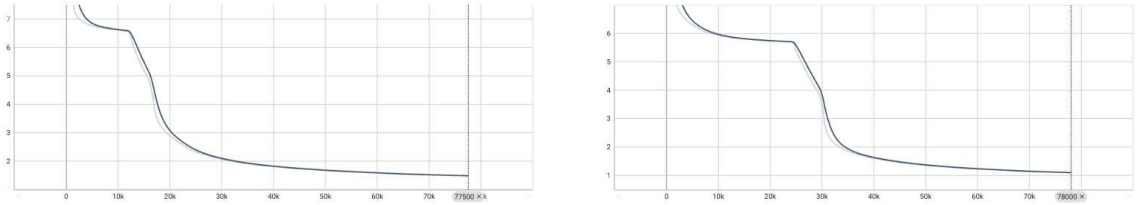


Figure 1.4: On the left hand side DrBERT NACHOS$_{small}$ loss and on the right hand side DrBERT NACHOS$_{large}$ loss.

1.5 Conclusion

In this work, we proposed the first biomedical and clinical Transformer-based language models, based on RoBERTa architecture, for the French language. An extensive evaluation study of these specific models has been performed on an aggregated collection of diverse private and public medical tasks. Our open-source DrBERT models improved the state of the art in all medical tasks against both the French general model (CamemBERT) and the English medical ones (BioBERT, PubMedBERT and ClinicalBERT). In addition, we showed that pre-training on constrained resources (4 GB) of web-crawled medical makes it possible to compete with, and even frequently surpass, models trained with specialized data from medical reports.

Results also highlighted that continual pre-training on an existing domain-specific English model, here PubMedBERT, is a more viable solution than on a French domain-generalist model while targeting French biomedical downstream tasks. It needs to further investigate the performance of this approach using more data, similar to what we have done with DrBERT NACHOS$_{large}$.

The pre-trained models as well as the pre-training scripts[5] have been publicly released online under an MIT open-source license. The main purpose of the NACHOS dataset is to promote the development of robust NLP tools by the community, so we have decided to make the corpora available for academic research.

[5] https://drbert.univ-avignon.fr/

Chapter 2

DrBenchmark: A Large Language Understanding Evaluation Benchmark for French Biomedical Domain

As demonstrated in previous chapters, the evaluation of language models is crucial for understanding their capabilities and limitations. While numerous benchmarks exist for general domain evaluation [279, 278], specialized domains like healthcare require dedicated evaluation frameworks that can assess both general language understanding and domain-specific knowledge [109, 111]. This is particularly challenging for languages other than English, where both the models and evaluation resources are scarce.

Several key research questions arise when considering the evaluation of domain-specific language models:

- How can we effectively evaluate the domain adaptation capabilities of language models in specialized fields?
- What metrics and tasks are most relevant for assessing medical language understanding?
- How do we ensure a comprehensive evaluation across different medical specialties and task types?
- Can we establish standardized evaluation protocols that facilitate fair comparisons between models?

This chapter presents DrBenchmark, the first large-scale evaluation framework for French biomedical language models [166]. Our work provides a systematic approach to assessing

the performance of language models in the French medical domain through a diverse set of tasks and metrics. Our main contributions are:

- We introduce a comprehensive collection of 12 medical datasets on HuggingFace[1] encompassing 20 diverse downstream tasks, including: POS tagging, NER, Multi-class and Multi-label classification, MCQA, *Semantic Textual Similarity* (STS)
- We conduct a comprehensive evaluation of 8 state-of-the-art masked language models:
  - French generalist models: CamemBERT [199], CamemBERTa [6], FlauBERT [174]
  - French biomedical models: DrBERT [165], CamemBERT-bio [265]
  - English biomedical model: PubMedBERT [108]
  - Cross-lingual generalist model: XLM-RoBERTa [67]
- We develop standardized evaluation protocols and metrics specifically designed for healthcare-related NLP tasks, ensuring reliable and reproducible model comparisons.
- We provide on GitHub an extensive evaluation framework with automated pipelines[2] for large-scale experiments, including support for *high-performance computing* (HPC) infrastructures.

Our benchmark incorporates datasets from various sources, including scientific literature, clinical trials, clinical cases, and speech transcriptions. These datasets represent a wide range of medical specialties and task types, from clinical case analysis to prescription understanding, providing a comprehensive assessment of medical language understanding capabilities.

The evaluation framework has been designed to be easily extensible, allowing for the integration of new datasets and metrics as they become available. This ensures that DrBenchmark can evolve alongside advances in medical NLP and continue to serve as a valuable resource for the research community.

2.1 DrBenchmark Overview

Our proposed benchmark comprises 20 French biomedical language understanding tasks, one of which is specifically created for this benchmark. The descriptions and statistics of these tasks are presented in Table 2.1. DrBenchmark encompasses the following overall aspects:

1. **A diverse set of tasks evaluating distinct model capabilities:** POS tagging assesses grammatical understanding and syntactic structures, NER evaluates lexical

[1] https://huggingface.co/DrBenchmark/datasets
[2] https://github.com/DrBenchmark/DrBenchmark

knowledge and domain-specific terminology comprehension, Multi-class and Multi-label classification test semantic categorization at different granularity levels, Intent classification measures understanding of medical context, MCQA evaluates medical reasoning capabilities, and STS assesses the model's ability to capture nuanced relationships between medical texts.

2. **A diverse range of data origins:** Scientific literature, clinical trials, clinical cases, speech transcriptions, and more as described in Table 2.2.

Dataset	Task	Metric	Train	Validation	Test	License
CAS	POS tagging	SeqEval F1	2,653	379	758	DUA
ESSAI	POS tagging	SeqEval F1	5,072	725	1,450	DUA
QUAERO	NER - EMEA	SeqEval F1	429	389	348	GFDL 1.3
	NER - MEDLINE	SeqEval F1	833	832	833	GFDL 1.3
E3C	NER - Clinical	SeqEval F1	969	140	293	CC BY-NC
	NER - Temporal	SeqEval F1	969	140	293	CC BY-NC
MorFITT	Multi-label Classification	Weighted F1	1514	1,022	1,088	CC BY-SA 4.0
FrenchMedMCQA	Question-Answering	Hamming / EMR	2,171	312	622	Apache 2.0
	Multi-class Classification	Weighted F1	2,171	312	622	Apache 2.0
Mantra-GSC	NER - EMEA	SeqEval F1	70	10	20	CC BY 4.0
	NER - Medline	SeqEval F1	70	10	20	CC BY 4.0
	NER - Patents	SeqEval F1	35	5	10	CC BY 4.0
CLISTER	Semantic Textual Similarity	EDRM / Spearman	499	101	400	DUA
DEFT-2020	Semantic Textual Similarity	EDRM / Spearman	498	102	410	DUA
	Multi-class Classification	Weighted F1	460	112	530	DUA
DEFT-2021	Multi-label Classification	Weighted F1	118	49	108	DUA
	NER	SeqEval F1	2,153	793	1,766	DUA
DiaMed	Multi-class Classification	Weighted F1	509	76	154	CC BY-SA 4.0
PxCorpus	NER	SeqEval F1	1,386	198	397	CC BY 4.0
	Multi-class Classification	Weighted F1	1,386	198	397	CC BY 4.0

Table 2.1: Descriptions and statistics of the 20 tasks included in DrBenchmark.

Please note that within DrBenchmark, we include classical tasks like NER and POS tagging, as well as more specific and challenging tasks like MCQA and multi-label classification. In Section 2.1.1, we provide an overview of the different French downstream tasks, while in Section 2.1.2, we offer insights into the pipeline and its reproducibility.

Dataset	Sources
CAS	Clinical cases
ESSAI	Clinical trial protocols
QUAERO	Drug leaflets & Biomedical titles
E3C	Clinical cases
MorFITT	Biomedical abstracts
FrenchMedMCQA	Pharmacy Exam
Mantra-GSC	Biomedical abstract / titles, drug labels, & patent
CLISTER	Clinical cases
DEFT-2020	Clinical cases, encyclopedia & drug labels
DEFT-2021	Clinical cases
DiaMed	Clinical cases
PxCorpus	Drug prescriptions transcripts

Table 2.2: Data sources covered by each dataset.

2.1.1 Downstream tasks

DEFT-2020 [40] contains clinical cases, encyclopedia, and drug labels introduced in the 2020 edition of an annual French Text Mining Challenge, called DEFT, and annotated for two tasks: (i) textual similarity and (ii) multi-class classification. The first task aims at identifying the degree of similarity within pairs of sentences, from 0 (the least similar) to 5 (the most similar). The second task consists of identifying, for a given sentence, the most similar sentence among the three sentences provided.

DEFT-2021 [107] is a subset of 275 clinical cases taken from the 2019 edition of DEFT. This dataset is manually annotated in two tasks: (i) multi-label classification and (ii) NER. The multi-label classification task focuses on identifying the patient's clinical profile based on the diseases, signs, or symptoms mentioned in the clinical cases. The dataset is annotated with 23 axes from Chapter C of the Medical Subject Headings (MeSH). The second task involves fine-grained information extraction for 13 types of entities (more details in Appendix 9.4).

E3C [196] is a multilingual dataset of clinical cases annotated for the NER task. It consists of two types of annotations (more details in Appendix 9.4): (i) clinical entities (e.g., pathologies), (ii) temporal information and factuality (e.g., events). While the dataset covers 5 languages, only the French portion is retained for the benchmark. Since the dataset does not come with pre-defined subsets, we performed a 70 / 10 / 20 random split, as described in Table 2.3.

Subset	Train	Validation	Test
Clinical	87.38 % of layer 2	12.62 % of layer 2	100 % of layer 1
Temporal	70 % of layer 1	10 % of layer 1	20 % of layer 1

Table 2.3: Description of the sources for E3C.

The QUAERO French Medical Corpus [211], simply referred to as QUAERO in this chapter, contains annotated entities and concepts in French for NER tasks. The dataset covers two text genres (drug leaflets and biomedical titles), consisting of a total of 103,056 words sourced from EMEA or MEDLINE. 10 entity categories corresponding to the UMLS Semantic Groups [188] were annotated (more details in Appendix 9.4). In total, 26,409 entity annotations were mapped to 5,797 unique UMLS concepts. Due to the presence of nested entities in annotations, we simplified the evaluation process by retaining only annotations at the higher granularity level from the BigBio [94] implementation, following the approach described in CamemBERT-bio [265], which translates into an average loss of 6.06% of the annotations on EMEA and 8.90% on MEDLINE. Additionally, considering that some documents from EMEA exceed the maximum input sequence length that most current language models can handle, we decided to split these documents into sentences.

MorFITT [170] is a multi-label dataset annotated with medical specialties. It contains 3,624 biomedical abstracts from PMC Open Access. It has been annotated across 12 medical specialties (more details in Appendix 9.4), for a total of 5,116 annotations.

FrenchMedMCQA [164] is a Multiple-Choice Question-Answering (MCQA) dataset for the biomedical domain. It contains 3,105 questions coming from real exams of the French medical specialization diploma in pharmacy, integrating single and multiple answers. The first task consists of automatically identifying the set of correct answers among the 5 proposed for a given question. The second task consists of identifying the number of answers (between 1 and 5) supposedly correct for a given question.

Mantra-GSC [155] is a multilingual dataset annotated for biomedical NER. From the 5 languages covered, we included only the French subset in this benchmark. The dataset is obtained from 3 sources, which have been partitioned to be evaluated separately by 2 annotation schemes (more details in Appendix 9.4): Medline (11 classes), and EMEA and Patents (10 classes). The sources cover different types of documents (biomedical abstracts/titles, drug labels, and patents). To ensure evaluation consistency, we randomly split the dataset into 3 subsets: 70% for training, 10% for validation, and 20% for testing.

CLISTER [122] is a French clinical cases STS dataset of 1,000 sentence pairs manually annotated by several annotators, who assigned similarity scores ranging from 0 to 5 to each pair. The scores were then averaged together to obtain a floating-point number representing the overall similarity. The objective of this dataset is to develop models that can automatically predict a similarity score that closely aligns with the reference score based solely on the two sentences provided.

CAS [105] comprises 3,790 clinical cases that have been annotated for POS tagging with 31 classes using automatic annotations through Tagex [3], with an evaluation conducted by comparing the automatic outputs against manual annotations. This evaluation yielded 98% precision. Since the dataset does not come with predefined subsets, we decided to randomly split it into 3 subsets of 70%, 10% and 20% of the total data for training, validation, and test, respectively.

ESSAI [73] contains 7,247 clinical trial protocols annotated in 41 POS tags using TreeTagger [247]. As the dataset was not originally divided into 3 subsets, we applied the same procedure as on the CAS corpus.

PxCorpus [153] is a spoken language understanding dataset in the domain of medical drug prescription transcripts. It includes 4 hours (1,981 recordings) of transcribed and annotated dialogues focused on drug prescriptions. The recordings were manually transcribed

[3] https://allgo.inria.fr/app/tagex

and semantically annotated. The first task involves classifying the textual utterances into one of the 4 intent classes (prescribe, replace, negate, none). The second task is a NER task where each word in a sequence is classified into one of 38 classes, such as drug, dose, or mode (more detail in Appendix 9.4).

DiaMed is an original dataset created specifically for DrBenchmark. 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 annotators, one of whom is a medical expert, into 22 chapters of the International Classification of Diseases, 10th Revision (ICD-10) [1]. These chapters provide a general description of the type of injury or disease. To ease the annotation process, only labels at the chapter level were used (more details in Appendix 9.4). The inter-annotator agreement between the 4 annotators has been computed for two annotation sessions (see Table 2.4), with 15 different clinical cases assessed per session.

	Session 1 - 0 to 15 docs		**Session 2 - 15 to 30 docs**	
Annotator ID	κ	$\mathcal{G}$	κ	$\mathcal{G}$
Annotator 1 & 2	0.538	0.566	0.697	0.705
Annotator 1 & 3	0.682	0.709	0.697	0.705
Annotator 1 & 4	0.397	0.429	0.548	0.558
Annotator 2 & 3	0.311	0.357	1.000	1.000
Annotator 2 & 4	0.472	0.497	0.672	0.707
Annotator 3 & 4	0.311	0.354	0.672	0.707
Average	0.452	0.485	0.714	0.730

Table 2.4: Inter-annotator agreement statistics. κ is referring to Kappa Cohen and $\mathcal{G}$ to Gwet's AC1.

2.1.2 Reproducibility and Usage

To facilitate the adoption of DrBenchmark and ensure consistency in implementations, we have developed a practical toolkit based on the HuggingFace Datasets library [179]. This toolkit includes data loaders that adhere to normalized schemes and predefined data splits. It also provides pre-training and evaluation scripts for each of the tasks, utilizing the HuggingFace Transformers [290] and PyTorch [227] libraries. For further guidance, we have integrated all the training details, including hyperparameters, in Appendix 9.3. This information will help users to reproduce and customize the experiments conducted with DrBenchmark[4].

[4] https://github.com/DrBenchmark/DrBenchmark

2.2 Language Models Studied

In this section, we outline the experimental protocol used to compare the performance of existing language models within DrBenchmark. To guarantee a fair comparison, we focus exclusively on pre-trained masked language models (MLMs) in this study. These MLMs are based on BERT-like architectures [81].

We first provide a brief overview in Section 2.2.1 of the 8 pre-trained language models that were studied: French generalist models (CamemBERT, CamemBERTa, and FlauBERT), cross-lingual generalist model (XLM-RoBERTa), French biomedical models (DrBERT and CamemBERT-bio), and English biomedical model (PubMedBERT). Subsequently, in Section 2.2.2, we describe the evaluation protocol employed to assess the performance of these models.

2.2.1 Pre-trained Masked Language Models

Table 2.5 summarizes the models and their parameters compared on DrBenchmark.

	Model	Tokenizer	Vocabulary	Pretraining	Corpus	Text Size
French Generalist	CamemBERTa	SentencePiece 32K	CCNET	from-scratch	CCNET	4 GB
	CamemBERT	SentencePiece 32K	OSCAR	from-scratch	OSCAR	138 GB
	FlauBERT	BPE 50K	Wiki + Web crawl	from-scratch	Wiki + Web crawl	71 GB
French Biomedical	DrBERT-FS	SentencePiece 32K	NACHOS	from-scratch	NACHOS	7.4 GB
	DrBERT-CP	WordPiece 30K	PubMed	continual pretraining	PubMed + NACHOS	21 + 4 GB
	CamemBERT-bio	SentencePiece 32K	OSCAR	continual pretraining	OSCAR + biomed-fr	138 + 2.7 GB
Cross-lingual Generalist	XLM-RoBERTa	WordPiece 30K	CC-100	from-scratch	CC-100	2.5 TB
English Biomedical	PubMedBERT	WordPiece 30K	PubMed	from-scratch	PubMed	21 GB

Table 2.5: Summary of the pre-training specifications for the different BERT-based models compared.

CamemBERT [199] is a RoBERTa-based model for French, pre-trained from-scratch on the generalist French 138 GB subset of OSCAR corpus [221].

CamemBERTa [6] is a DeBERTaV3 [118] based model pre-trained from-scratch on around 30% of the French subset of CCNET corpus [289] used for CamemBERT$_{CCNET}$, that had seen approximately 133 billion tokens during its pre-training.

FlauBERT [174] is a BERT-based model pre-trained from scratch using a subsample of 71 GB of the French Common Crawl and Wikipedia corpora.

XLM-RoBERTa [67] is a cross-lingual RoBERTa-based model trained on 116 languages, including French, by using 2.5 TB of the CommonCrawl corpus.

PubMedBERT [109] is a BERT-based biomedical-specific model pre-trained from scratch on the 3.1 billion words of the PubMed corpus (21 GB). This is the only model for English.

DrBERT-FS and DrBERT-CP [165] are French biomedical MLMs built using a from-scratch pre-training of RoBERTa (DrBERT-FS) and continual pre-training of PubMedBERT (DrBERT-CP) from the French public biomedical corpus NACHOS [165], integrating 1.08 billion words (7.4 GB) and 646 million words (4 GB) respectively.

CamemBERT-bio [265] is a French biomedical language model built using a continual pre-training of the CamemBERT$_{OSCAR-138GB}$ model. It was trained on the French public corpus biomed-fr [265] with 413 million words (2.7 GB) and a wide range of data collected on the web.

2.2.2 Models evaluation

All the models are fine-tuned according to a strict protocol using the same hyperparameters for each downstream task. The reported results are obtained by averaging the scores from four separate runs, thus ensuring robustness and reliability. We also report statistical significance computed using Student's t-test.

To ensure a fair and consistent comparison among systems for sequence-to-sequence tasks such as POS tagging and NER, we chose the SeqEval [208] metric in conjunction with the IOB2 format and the training of all the models to predict only the label on the first token of each word, as mentioned by [265]. It provides a tokenizer-agnostic evaluation and mitigates any correlation between models' performances and the tokenization process.

For STS tasks, the models' performance was assessed using two metrics: (1) the Spearman correlation, and (2) the mean relative solution distance accuracy (EDRM), as defined by the original authors of the DEFT-2020 dataset [40].

2.3 Experiments and Results

In Section 2.3.1, we compare the results obtained by each model within DrBenchmark, which permits positioning a wide range of state-of-the-art models in the biomedical field across various NLP tasks. Then, we propose to gain a comprehensive understanding of the models' behavior by examining areas such as low-resource fine-tuning scenarios (Section 2.3.2) and the analysis of word tokenization of the studied models (Section 2.3.3).

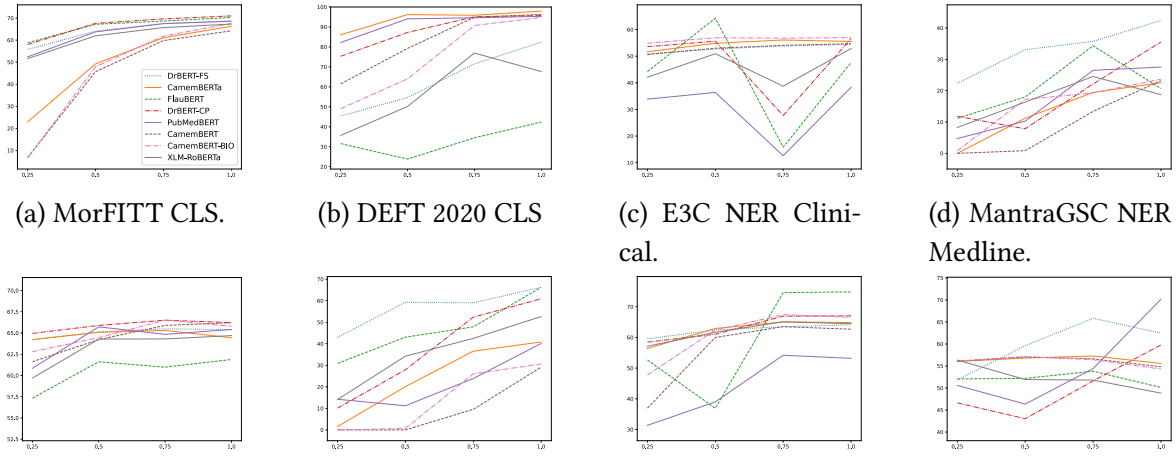
2.3.1 Comparison of Models' Performance

			French Generalist			French Biomedical			English Biomedical	Cross-lingual Generalist
Dataset	**Task**	**Baseline**	**CamemBERT**	**CamemBERTa**	**FlauBERT**	**DrBERT-FS**	**DrBERT-CP**	**CamemBERT-bio**	**PubMedBERT**	**XLM-RoBERTa**
CAS	POS	23.50	95.53**	96.56**	95.22**	**96.93**	96.46**	95.22**	94.82**	96.91
ESSAI	POS	26.31	97.38**	98.08**	97.05*	**98.41**	98.01**	97.39**	97.42**	98.34
QUAERO	NER EMEA	8.37	62.68**	64.86**	**74.86**	64.11**	67.05**	66.59**	53.19**	64.47**
	NER MEDLINE	4.92	55.25**	55.60**	48.98	55.82**	**60.10**	58.94	53.26**	51.12**
E3C	NER Clinical	4.47	54.70**	55.53	47.61	54.45	56.55	**56.96**	38.34	52.87**
	NER Temporal	21.74	**83.45**	83.22	61.64	81.48**	83.43	83.44	80.86**	82.6
MorFITT	Multi-Label CLS	3.24	64.21**	66.28**	70.25	68.70**	**70.99**	67.53**	68.58**	67.28**
FrenchMedMCQA	MCQA	21.83 / **11.57**	28.53 / 2.25**	29.77 / 2.57**	27.88 / 2.09**	31.07 / 3.22**	32.41 / 2.89**	**35.3** / 1.45	32.90 / 1.61**	34.74 / 2.09**
	CLS	8.37	66.21	64.44**	61.88	65.38	**66.22**	65.79	65.41*	64.69*
MantraGSC	NER FR EMEA	0.00	29.14**	40.84**	66.20	**66.23**	60.88	30.63**	40.14**	52.64*
	NER FR Medline	7.78	23.20**	22.55**	20.69	**42.38**	35.52	23.66**	27.53*	18.73*
	NER FR Patents	6.20	00.00**	44.16**	31.47**	**57.34**	39.68	00.00**	4.51**	8.58**
CLISTER	STS	0.44 / 0.00	0.55 / 0.33**	0.56 / 0.47**	0.50 / 0.29**	0.62 / 0.57**	0.60 / 0.49*	0.54 / 0.26**	**0.70** / **0.78**	0.49 / 0.23**
DEFT-2020	STS	0.49 / 0.00	0.59 / 0.58**	0.59 / 0.43**	0.58 / 0.51**	0.72 / 0.81*	0.73 / **0.86**	0.58 / 0.32**	**0.78** / **0.86**	0.60 / 0.26**
	CLS	14.00	96.31	**97.96**	42.37**	82.38	95.71*	94.78*	95.33*	67.66**
DEFT-2021	Multi-Label CLS	24.49	18.04**	18.04**	**39.21**	34.15**	30.04**	17.82**	25.53**	24.46**
	NER	0.00	62.76**	62.61**	33.51	60.44**	63.43*	**64.36**	60.27**	60.32**
DiaMED	CLS	15.36	30.40**	24.05**	34.08**	**60.45**	54.43**	39.57**	54.96**	26.69**
PxCorpus	NER	10.00	92.89**	95.05**	47.57	**95.88**	71.38	93.08**	94.66**	95.80
	CLS	84.78	94.41	93.95	93.45*	94.43	**94.52**	94.49	93.12	93.91

Table 2.6: Performance of the studied models over 4 runs. Best model in bold and second is underlined. Statistical significance is computed using Student's t-test: * stands for $p < 0.05$, ** stands for $p < 0.01$.

The results of the 8 models are reported in Table 2.6 and compared to a baseline obtained by considering the majority class for all predictions. Overall, although we might anticipate certain models to excel in all tasks, we discovered that no single model outperforms the rest in all application scenarios. Interestingly, most of the models examined manage to secure the top position in at least one of the French biomedical downstream tasks studied. The only exception pertains to the cross-lingual generalist model (`XLM-RoBERTa`), which manages to reach the second-best position on several tasks.

Despite this unexpected outcome, we observe that French biomedical language models (`DrBERT-FS`, `DrBERT-CP`, `CamemBERT-bio`), presumed to be the most aligned with the nature of the data of the benchmark, exhibit indeed superior performance across many tasks. More precisely, `DrBERT-FS` achieves the highest performance in 8 tasks, `DrBERT-CP` in 5 tasks, and `CamemBERT-bio` in 2 tasks. This indicates that domain and language-specialized models achieve the best performance in up to 75% of the DrBenchmark downstream tasks.



(a) MorFITT CLS. (b) DEFT 2020 CLS (c) E3C NER Clinical. (d) MantraGSC NER Medline.

(e) FrenchMedMCQA CLS (f) MantraGSC NER EMEA. (g) QUAERO NER EMEA. (h) CLISTER STS.

Figure 2.1: Performance with varying training subset sizes (25%, 50%, 75% and 100%). Results are reported on the full test set.

Biomedical vs. Generalist. The nature of the data appears to have an influence. Generalist models (`CamemBERT`, `CamemBERTa`, `FlauBERT`, and `XLM-RoBERTa`) are more suitable for tasks that require extensive linguistic knowledge but may not perform as well as specialized models nor even reach their level of performance. We observe that all generalist models obtain better performance only on 4 out of the 20 tasks, but remain competitive on most tasks. Furthermore, our experiments with `DrBERT-FS` indicate that biomedical models may require less pre-training data compared to generalist ones. However, it is important to note that this observation requires further confirmation. In some tasks, biomedical models that undergo continual pre-training from a generalist model, such as `CamemBERT-bio`, can prove to be the most effective, underscoring the value of pre-training on generalist datasets.

From-scratch vs. Continual Pre-Training. `DrBERT-CP` and `CamemBERT-bio`, pre-trained from `PubMedBERT` and `CamemBERT` respectively, demonstrate improved performance compared to their initial models. Notably, `DrBERT-CP` outperforms `CamemBERT-bio` in 15 out of 20 tasks. These findings suggest that when it comes to continual pre-training, starting with a specialized model in the specific domain (here, `PubMedBERT`) may be a better choice than a generalist model (here, `CamemBERT`), even with different languages. Additionally, we observe that `DrBERT-FS` achieves the highest performance in 8 tasks, suggesting that starting *from-scratch* can be a competitive strategy compared to *continual pre-training*.

French vs. Other language. French models generally achieve better performance compared to English or multilingual ones. When considering the English `PubMedBERT` model, we observe that its performance in most tasks is comparable to that of the French models, with the exception of NER tasks, where French models demonstrate superiority. Thus, we observe that the language appears to be less prominent when utilized in domain-specific tasks, such as those in the biomedical field.

RoBERTa vs. DeBERTaV3 architectures. Despite being trained on only 30% of the pre-training data used by CamemBERT$_{CCNET}$, CamemBERTa achieves identical or better performances in 68% of the tasks (12 out of 20), benefiting from the DeBERTaV3 architecture in domain-specific scenarios. However, all the models based on CamemBERT face difficulties in corpora with a limited amount of data, such as MantraGSC Patents, where they fail to generate labels other than 'O'. On the other hand, in the same low-resource scenarios, CamemBERTa models exhibit greater robustness and achieve superior performance. The architecture on which the models are based therefore, seems to play a role in the performance obtained.

		French Generalist			French Biomedical			English Biomedical	Cross-lingual Generalist
Dataset	**Task**	**CamemBERT**	**CamemBERTa**	**FlauBERT**	**DrBERT-FS**	**DrBERT-CP**	**CamemBERT-bio**	**PubMedBERT**	**XLM-RoBERTa**
CAS	POS	1.63	1.64	**1.34**	1.36	1.81	1.63	1.81	1.8
ESSAI	POS	1.55	1.56	**1.28**	1.29	1.78	1.55	1.78	1.75
QUAERO	NER EMEA	1.66	1.67	**1.37**	**1.37**	1.73	1.66	1.73	1.77
	NER Medline	2.01	2.01	**1.58**	1.64	1.97	2.01	1.97	2.18
E3C	NER FR Clinical	1.64	1.65	1.39	**1.32**	1.80	1.64	1.80	1.78
	NER FR Temporal	1.63	1.63	1.38	**1.31**	1.80	1.63	1.80	1.76
MorFITT	Multi-Label CLS	1.51	1.51	**1.33**	1.39	1.91	1.51	1.91	1.73
FrenchMedMCQA	MCQA	1.80	1.80	**1.55**	**1.55**	2.03	1.80	2.03	2.00
	CLS	1.80	1.80	**1.55**	**1.55**	2.03	1.80	2.03	2.00
MantraGSC	NER FR EMEA	1.50	1.46	**1.34**	1.37	1.99	1.50	1.99	1.71
	NER FR Medline	2.25	2.25	**1.88**	2.05	2.47	2.25	2.47	2.49
	NER FR Patents	1.58	1.58	**1.41**	1.51	2.06	1.58	2.06	1.86
CLISTER	STS	1.76	1.76	**1.55**	**1.55**	2.09	1.76	2.09	1.93
DEFT-2020	STS	1.43	1.43	**1.31**	1.45	1.92	1.43	1.92	1.64
	CLS	1.31	1.32	**1.20**	1.23	1.75	1.31	1.75	1.51
DEFT-2021	CLS	1.70	1.71	**1.48**	1.51	2.05	1.70	2.05	1.90
	NER	1.62	1.63	**1.35**	**1.35**	1.80	1.62	1.80	1.79
DiaMED	CLS	1.66	1.67	**1.45**	1.46	1.99	1.66	1.99	1.88
PxCorpus	NER	1.71	1.76	**1.63**	1.66	2.13	1.71	2.13	1.83
	CLS	1.71	1.76	**1.63**	1.66	2.13	1.71	2.13	1.83
Average		*1.67*	*1.67*	***1.43***	*1.47*	*1.90*	*1.67*	*1.90*	*1.85*

Table 2.7: Average sub-word units per word for each model and dataset. For each task, the lowest sub-word value is shown in bold, and the highest value is underlined. Models are grouped based on their tokenizer type. Cells in green indicate the best model in terms of performance for the task, while cells in red indicate the worst model.

2.3.2 Impact of Fine-Tuning with Limited Data

Unlike the process of training language models, the fine-tuning approach involves utilizing annotated data to adapt a pre-trained language model for solving specific downstream tasks. In the previous section, we observed that language models pre-trained on medical data generally achieved better performance on DrBenchmark compared to generalist models trained on much larger datasets. However, we now question the models' ability to be effectively applied to biomedical tasks when there is limited fine-tuning training data available. For this purpose, we conducted experiments by varying the amount of training data

during the fine-tuning process by randomly choosing four percentages of the training data: 25%, 50%, 75% and 100%. To make the experiment as fair as possible, we did four runs for each percentage, model and dataset combination. The validation and test sets have not been changed for the sake of comparison.

We observe that on certain datasets, some models capture information more quickly than others, like in Figures 2.1b, 2.1f and 2.1a. Unsurprisingly, in almost all scenarios, having the complete training set yields better results than having only 25% of it. However, we note a few exceptions in Figures 2.1a and 2.1h with `FlauBERT`, where we observe the opposite trend. For intermediate percentages, 50% and 75%, we observe a decrease in performance with certain models, such as `FlauBERT` in Figures 2.1a and 2.1g, and `DrBERT-CP` in Figures 2.1d and 2.1h. In NER tasks (Figures 2.1a, 2.1d, 2.1f and 2.1g), `DrBERT-FS` achieves the best performance in scenarios with very little data, indicating good model robustness.

2.3.3 Analysis of Word Tokenization

Tokenizers play a crucial role in MLMs by utilizing size-limited vocabularies to split texts into sub-units, aiming to handle out-of-vocabulary (OOV) words. Due to variations in the training data, vocabularies differ across different models, as illustrated in Figure 2.2. As a result, tokenizers segment words in distinct ways, yet remarkably achieve similar performance levels as previously noted in Table 2.6.

Figure 2.2: Vocabularies inter-coverage matrix.

So far, there has been a prevailing notion in the community that excessive segmentation of words in tokenization leads to a loss of morphological form and semantic meaning, introducing noise and adversely affecting performance [63, 126, 35]. However, our experiments,

as shown in Table 2.7, reveal that FlauBERT is the model with the least word segmentation (1.43 on average), while DrBERT-CP tends to have the highest average segmentation (1.90 on average). Surprisingly, when comparing the performance of these two models on the benchmark tasks, we observe that DrBERT-CP outperforms FlauBERT on 16 out of the 20 tasks, thus contradicting previous conclusions drawn by the community. Table 2.8 provides some examples of the tokenization done by each analyzed model, showcasing a list of commonly used biomedical terms. Yet, tokenization, as it is currently done in MLMs, seems to play a minor role in the performance of systems.

		French Generalist		French Biomedical	English Biomedical	Cross-lingual Generalist
Term	**CamemBERTa**	**CamemBERT CamemBERT-bio**	**FlauBERT**	**DrBERT-FS**	**PubMedBERT DrBERT-CP**	**XLM-RoBERTa**
asymptomatique	a-s-ym-pto-matique	a-s-y-mp-to-matique	as-ym-ptom-atique	✓	asympt-omat-ique	as-y-mp-tomat-ique
blépharorraphie	blé-phar-or-ra-phi-e	blé-phar-or-ra-phi-e	bl-é-phar-or-raph-ie	blé-ph-ar-or-ra-ph-ie	ble-pha-ror-ra-phi-e	b-lép-har-orra-phi-e
bradycardie	brad-y-cardi-e	brad-y-cardi-e	bra-dy-car-die	✓	brady-car-di-e	bra-dy-card-ie
bronchographie	bronch-ographie	bron-ch-ographie	bron-cho-graphie	bronch-ographie	bronch-ograph-ie	bron-ch-ographie
bronchopneumopathie	bronch-op-ne-um-opathie	bron-cho-p-ne-um-opathie	bron-chop-neu-mo-pathie	bronchop-neumopathie	bronch-op-neum-opath-ie	bron-chop-ne-umo-pathi-e
dysménorrhée	dys-mén-or-r-h-ée	dys-mén-or-r-h-ée	dys-mé-nor-rh-ée	dys-m-énorrhée	dysm-eno-rr-he-e	dys-mén-or-r-hé-e
glaucome	gla-uc-ome	gla-uc-ome	glau-come	✓	glauc-ome	gla-u-come
IRM	✓	✓	✓	✓	ir-m	I-RM
kystectomie	k-yst-ectomie	ky-st-ectomie	ky-st-ec-tomie	kys-tectomie	ky-st-ectom-ie	ky-st-ecto-mie
neuroleptique	neuro-le-p-tique	neuro-le-p-tique	neur-ol-ep-tique	neur-oleptique	neurol-ept-ique	neuro-lep-tique
nicotine	✓	✓	✓	✓	✓	nico-tine
poliomyélite	poli-om-y-élite	poli-om-y-élite	poli-omy-élite	poli-omyélite	poli-omyel-ite	poli-om-y-é-lite
rhinopharyngite	rh-ino-phar-y-ng-ite	rhin-oph-ary-ng-ite	rh-ino-phar-yn-gite	rhin-opharyng-ite	rhin-oph-aryng-ite	r-hin-op-har-y-ng-ite
toxicomanie	toxico-mani-e	toxico-mani-e	✓	✓	toxic-oman-ie	toxic-om-anie
vasoconstricteur	vas-oc-on-strict-eur	vas-oc-on-strict-eur	vas-o-cons-tri-cteur	vasoconstric-teur	vasoconstric-te-ur	vaso-con-strict-eur

Table 2.8: Visual comparison of models' tokenization on commonly used biomedical terms. A checkmark indicates that the word is present as a complete token, while hyphens separate subword units. ✓refers to the word being available as a unigram in the vocabulary of the tokenizer.

Table 2.9 summarizes the results obtained on average by the considered MLMs when aggregating the tasks into one of the five designated categories: POS, NER, MCQA, MCC (Multi-class classification), MLC (Multi-label classification), or STS tasks. Upon analyzing the average performance by task category, it becomes evident that the leading model, DrBERT-FS, does not excel in tasks such as MLC or STS. For example, the multilingual biomedical model PubMedBERT demonstrates a notable advantage, with nearly 18 EDRM points ahead of CamemBERT-bio in the STS tasks.

	Tasks					
Models	**POS**	**NER**	**MCQA**	**MCC***	**MLC***	**STS**
CamemBERT	96.45	51.52	28.53 / 2.25	71.83	41.12	0.57 / 0.45
CamemBERTa	97.32	58.16	29.77 / 2.57	70.10	42.16	0.57 / 0.45
FlauBERT	96.13	51.85	27.88 / 2.09	57.94	**54.73**	0.54 / 0.40
DrBERT-FS	**97.67**	**64.23**	31.07 / **3.22**	75.66	51.42	0.67 / 0.69
DrBERT-CP	97.23	59.84	32.41 / 2.89	**77.72**	50.51	0.66 / 0.67
CamemBERT-bio	96.30	53.06	**35.30** / 1.45	73.65	42.67	0.56 / 0.29
PubMedBERT	96.12	46.93	32.90 / 1.61	77.20	47.05	**0.74** / **0.82**
XLM-RoBERTa	97.62	54.21	34.74 / 2.09	63.23	45.87	0.54 / 0.24

Table 2.9: Average results obtained by the different MLMs for each type of task. MLC stands for Multi-label classification and MCC for Multi-class classification.

2.4 Conclusion

In this chapter, we introduced DrBenchmark, the first large language understanding benchmark tailored for the French biomedical domain. We conducted a qualitative evaluation of 8 state-of-the-art masked language models (MLMs) on this comprehensive benchmark, encompassing 20 diverse downstream tasks. Our findings illuminate the limitations of generalist models in tackling complex biomedical tasks, emphasizing the importance of employing domain-specific models to achieve peak performance. While the French biomedical models excel in most tasks, no single model emerges as universally superior. Remarkably, certain out-of-domain models or models trained in different languages exhibit superior performance in specific tasks and maintain competitiveness in others.

In conclusion, we have observed that several biomedical tasks in DrBenchmark exhibit relatively poor performance, even when utilizing specialized biomedical models. We postulate that the models examined in this study, here state-of-the-art MLMs, may not be the most effective choices for specific tasks such as question-answering or multi-label classification. In our future research, we intend to shift our focus towards generative approaches, such as LLaMA [267], OPT [312], or GPT-NeoX-20B [31], as well as their instruction-tuned counterparts [133]. These alternatives may offer more suitable solutions for addressing these types of tasks.

Chapter 3

How Important Is Tokenization in French Medical Masked Language Models?

In recent years, the success of language models in NLP has been intrinsically linked to their tokenization strategies, the process of converting raw text into meaningful subword units for model processing. While these strategies have proven effective for general domain tasks, their application to specialized domains like medical text, particularly in languages other than English, presents unique challenges. Medical terminology often follows distinct morphological patterns and contains specialized vocabulary that may not be optimally captured by conventional tokenization approaches.

Several key research questions arise when considering the importance of tokenization for domain-specific language models:

- Since morphemic decomposition is fundamental to human understanding, can it also enhance language modeling and understanding tasks for machines?
- How do different tokenization strategies, from purely statistical to linguistically-informed, impact model performance in a specialized domain?
- Is there an optimal level of tokenization granularity for complex medical terminology?
- To what extent does the source and size of the tokenizer's training data influence the final model's effectiveness?
- Can a morpheme-enriched tokenization approach outperform standard methods on French biomedical tasks?

This chapter, based on our work accepted at LREC-COLING 2024 [162], investigates these questions. Since medical terms are often complex compositions of meaningful subunits (morphemes), purely statistical tokenizers like BPE and SentencePiece risk segmenting them into semantically irrelevant pieces. To address this, we compare these traditional methods with a novel morpheme-enriched approach designed to preserve the linguistic building blocks of medical vocabulary. We evaluate the impact of these strategies by assessing model performance across the 23 biomedical NLP tasks from our DrBenchmark framework (Chapter 2). Our main contributions are:

- We introduce a novel morpheme-enriched tokenization strategy designed to produce more semantically coherent subword units by integrating domain-specific linguistic knowledge.
- We perform a large-scale comparative study of 7 different tokenization approaches, training 7 new French medical MLMs from scratch and evaluating them on the 23 biomedical NLP tasks from the DrBenchmark framework.
- We release a comprehensive suite of 17 tokenizers[1] on the Hugging Face Hub, encompassing various algorithms, data sources, and our novel morpheme-enriched variants.
- To ensure full reproducibility, we provide our complete experimental framework on GitHub[2], including code for tokenizer training, model pre-training, downstream evaluation, and result analysis.

Our findings reveal that while no single tokenization strategy is universally optimal, our morpheme-enriched approach often leads to more linguistically coherent tokenization of specialized medical terms. This work contributes to a deeper understanding of how tokenization choices influence model performance in specialized domains and offers practical insights for developing robust medical language models for French and other morphologically rich languages.

3.1 Tokenization Strategies

In the following section, we provide a brief overview of the two studied classical statistical-based tokenization approaches (Section 3.1.1), followed by the description of our original approach that integrates linguistic knowledge through morphemes into existing tokenizers' algorithms (Section 3.1.2).

3.1.1 Statistical Tokenization Algorithms

In this chapter, we compare two statistically based tokenization methods, BPE and SentencePiece (see Section 0.4.4). BPE begins with individual characters and progressively combines them into subword pairs based on their frequency in the training data. In contrast,

[1] https://huggingface.co/BioMedTok/models?sort=downloads
[2] https://github.com/BioMedTok/BioMedTok

SentencePiece employs two subword segmentation algorithms, Unigrams and BPE, offering flexibility in terms of segmentation granularity. While SentencePiece is widely used in French biomedical models [265, 165, 68, 29], its appropriateness for a specific language and domain may vary, potentially leading to suboptimal subword segmentation.

3.1.2 Morpheme-enriched Tokenization

In this chapter, focusing on improving the modeling of specialized medical terminology in the medical field and reducing the impact of unseen words during model pre-training, our primary emphasis is on lexical morphemes [266]. To achieve this, we created a manual list of around 600 frequently used lexical morphemes in the French medical domain, sourced from the book in [69]. Examples of these morphemes include terms like `céphal-`, `clinico-`, `-thérapie`, `thoraco-`, `-ome` and `-gène`.

We trained our morpheme-enriched tokenizers by modifying both the BPE and SentencePiece algorithms. During training, we introduced a predefined list of language-specific morphemes as tokens. These morphemes were enforced selections by the tokenizer when encountered, while the remaining text underwent the standard tokenization process of the chosen algorithm. This approach enabled us to combine traditional BPE and SentencePiece tokenizations with morpheme tokens, mitigating issues related to unseen words during training.

3.2 Experimental Protocol

In this section, we outline the experimental approach used to evaluate the impact of tokenization strategies on French biomedical PLMs. Firstly, in Section 3.2.1, we present the set of 23 selected biomedical NLP downstream tasks used in our study and taken from our previous work DrBenchmark [167]. Next, we describe the different training data sources employed to train the statistical tokenizers in Section 3.2.2. Following this, in Section 3.2.3, we explain the training procedure for the chosen BERT-based model architecture. Finally, in Section 3.2.4, we provide a comprehensive description of the evaluation methodology used to assess the performance of these models.

3.2.1 Downstream Tasks

We summarize the datasets of the 23 NLP biomedical downstream tasks from DrBenchmark [167], including NER, part-of-speech (POS) tagging, STS, and classification.

3.2.2 Tokenizers Data Sources

To ensure a fair and comprehensive comparison of training data sources used by the statistical tokenizers, we carefully curated a 1GB subset of raw, lowercase text data from a

variety of sources, including NACHOS [165], PubMed Central, CC100 [289], and the French Wikipedia. We then constructed tokenizers using both tokenization algorithms, resulting in a total of 16 tokenizers: 8 with the integration of morphemes and 8 without. These specific data sources were chosen for their diversity: NACHOS focuses on French biomedical content, PubMed Central on English biomedical content, Wikipedia on general French language, and CC100 on general multilingual content. Each tokenizer was configured with a vocabulary size of 32k tokens, consistent with the original hyperparameters used in other French biomedical models such as CamemBERT-BIO [265] and DrBERT [165].

3.2.3 Language Model Pre-Training

To assess the impact of introducing morphemes into tokenizers on the pre-training process of biomedical language models, we conducted pre-training from scratch using the 16 tokenizer combinations (see Section 3.2.2). Our choice of architecture was RoBERTa [193], which is based on the masked language modeling objective and configured with standard token masking percentages as introduced by the authors.

For the PLMs training data, we utilized the NACHOS corpus created by [165]. This corpus, already pre-processed and converted to lowercase, is consistent with the data sources used for training the tokenizers. It comprises 1.1 billion words, equivalent to 7.4GB of raw text data, sourced from a wide range of online resources focusing on the French biomedical and clinical domains.

The pre-training process was conducted uniformly across all models, employing the same hyperparameters and executed over a 20-hour period. We harnessed the computational power of 32 V100 32GB GPUs available on the Jean-Zay supercomputer for this purpose. By maintaining consistent procedures and employing a fixed seed to mitigate randomness during training, we ensured the reliability and reproducibility of our experiments.

3.2.4 Evaluation

All models undergo fine-tuning following a standardized protocol with identical hyperparameters for each downstream task, enabling a focused evaluation of tokenizers. We ensure robustness and reliability by averaging the results across four independent runs and performing statistical significance assessments using Student's t-test.

For consistent comparisons, especially in sequence-to-sequence tasks like POS tagging and NER, we employ the SeqEval [208] metric in conjunction with the IOB2 format. To align with established practices [265], our models are trained to predict only the label for the initial token of each word.

3.3 Results and Discussions

			BPE								SentencePiece							
			NACHOS		*PubMed*		*CC100*		*Wiki*		*NACHOS*		*PubMed*		*CC100*		*Wiki*	
Dataset	**Task**	**Metric**	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*
CAS	***CLS***	F1	94.2*	94.9	94.7	94.2*	95.2	**95.3**	94.8	94.8	94.8	94.7*	93.4**	93.6**	94.4	94.1	95.3*	95.1**
	NER Neg	SeqEval	**87.0**	83.3*	82.4**	81.3**	84.9	84.2	84.7	84.5*	86.1	86.4	83.6*	83.9	85.4	84.2**	85.6	83.2
	NER Spec	SeqEval	30.3*	30.6	35.0	28.2*	34.6	32.0	34.4	34.0	**36.1**	29.8	28.4*	22.2**	31.9	28.7*	32.1	27.0*
	POS	SeqEval	97.0**	96.9**	97.1	96.9**	97.1*	97.0**	**97.2**	96.9**	97.1*	97.0*	96.9**	96.9**	97.1	97.1	97.1	97.1
PxCorpus	***CLS***	F1	94.8	94.2	93.6	93.9	94.2	94.6	93.4	93.7	**94.9**	94.1	94.8	94.1	94.8	93.7	93.7	94.5
	NER	SeqEval	95.9	95.9	95.9	95.9	96.1	96.0	96.2	95.9	96.1	96.1	96.0	96.1	95.9	96.1	**96.2**	96.1
DEFT2020	***STS***	MSE	0.71	0.71	0.64*	**0.75**	0.70	0.67	0.71	0.69	0.72	0.71	0.63**	0.63	0.70	0.67*	0.70	0.67*
	CLS	F1	**91.0**	85.9	57.6**	73.7	79.5	76.3	77.1	66.0	83.0	85.3	80.9	66.7**	61.1*	66.3*	75.0*	77.4*
MORFITT	***CLS***	F1	68.6**	68.0**	66.5**	65.9**	68.4**	67.0**	68.7	67.3**	**69.6**	68.8*	66.8**	66.2**	68.2	67.5**	69.1**	67.7**
E3C	***NER Clinical***	SeqEval	**54.2**	53.1	52.4	48.6**	52.7	51.3**	51.1*	52.0*	54.2	52.4	52.1	51.1**	53.8	52.5*	53.2	51.7
	NER Temporal	SeqEval	82.0	81.2	80.9**	80.0**	81.8	81.2	**82.3**	80.6**	82.1	81.6	80.3**	79.8**	80.6**	81.1**	81.6*	81.73*
CLISTER	***STS***	MSE	0.63*	0.63	0.63	0.60**	0.65	0.63	0.62**	**0.66**	0.61*	0.64	0.61**	0.62**	0.62	0.60*	0.64**	0.63**
DEFT2021	***NER***	SeqEval	60.3	59.0**	58.1**	56.2**	59.4**	59.2**	60.1**	59.1**	**61.3**	60.1*	57.0**	56.6**	59.2**	59.9**	59.3**	58.9**
	CLS	F1	32.9	34.5*	33.4	32.3	34.5*	33.9	34.2	32.9	34.3	33.1	34.3	33.1	31.0	31.9*	34.2	**34.9**
ESSAI	***NER Spec***	SeqEval	60.5	60.9	56.4*	59.2	57.9	61.5	63.6	57.4	63.9	62.8	57.6	55.7*	**64.6**	62.0	61.4	63.1
	POS	SeqEval	98.4*	98.3	98.3	98.2**	98.4	98.4	98.3	98.3	98.4	98.4	98.3	98.2*	**98.4**	98.3	98.3*	98.3
	NER Neg	SeqEval	83.0	83.4	79.3	76.4	82.2	83.2	81.8	84.2*	81.3	84.0*	80.2	81.1	83.2	**84.2**	82.1	79.6*
	CLS	F1	97.3	97.1*	97.4	96.6**	97.4	96.7**	97.4	97.0**	97.3	97.3	**97.5**	97.2*	97.0	97.0	97.5*	97.0*
QUAERO	***NER Medline***	SeqEval	57.7	56.2**	55.4**	53.6**	57.9	55.0**	57.3	56.4**	**58.2**	55.5**	54.8**	52.9**	57.5*	55.8**	56.9	54.9**
	NER EMEA	SeqEval	65.6	65.1	63.9	63.1**	62.1**	62.7*	63.1**	62.6*	65.5	**65.9**	62.6**	63.8*	62.8**	63.1*	62.7*	62.0**
MantraGSC	***NER EMEA***	SeqEval	60.9	63.9	58.2*	60.6*	**69.3**	63.0	61.9*	62.3**	66.9	62.5*	56.8**	60.3	60.8*	59.5	64.0*	63.9**
	NER Medline	SeqEval	41.4*	42.9	39.3	36.2**	44.3	41.2	43.8	40.8*	41.9	39.5*	36.4**	37.8	46.4*	39.9	**47.1**	36.1*
	NER Patents	SeqEval	52.1*	53.3*	**57.0**	50.2*	57.0	53.9	53.6	52.3*	52.0	49.6*	50.7**	49.4	52.8*	48.0	50.6*	47.8*
Average performances per tasks																		
	CLS	F1	**79.80**	79.10	73.87	76.10	78.20	77.30	77.60	75.28	78.98	78.88	77.95	75.15	74.42	75.08	77.47	77.77
	NER	SeqEval	63.92	63.75	62.63	60.73	64.63	63.42	64.15	63.24	**65.05**	63.55	61.27	60.82	64.22	62.69	64.06	62.00
	POS	SeqEval	97.70	97.60	97.70	97.55	**97.75**	97.70	97.75	97.60	97.75	97.70	97.60	97.55	97.75	97.70	97.70	97.70
	STS	MSE	0,67	0,67	0,64	0,68	**0,68**	0,65	0,67	0,68	0,67	0,68	0,62	0,63	0,66	0,64	0,67	0,65

Table 3.1: Performance of the tokenization algorithms and different data sources used to train tokenizers (top). Average performance per type of task is also reported (bottom). *w/o* and *w/* denote models without and with morphemes. Best models are in bold, and the second-best are underlined. Statistical significance is determined using Student's t-test, where * indicates $p < 0.05$, and ** $p < 0.01$.

In this section, we present the results of our tokenization strategies on various biomedical NLP tasks, with a focus on key aspects. We investigate the impact of tokenization granularity (Section 3.3.1), the introduction of morphological information during tokenizer construction (Section 3.3.2), and the influence of data sources on tokenizers, including token sparsity, morpheme coverage, and the overall performance of different tokenization algorithms (Section 3.3.3).

Table 3.1 summarizes the performance of the BPE and SentencePiece strategies, both with (*w/*) and without our morpheme-enriched approach (*w/o*), across various French biomedical downstream tasks. Average performance per task type is also provided for clarity. It's worth noting that, before delving into detailed analysis, there is no consistent tokenization strategy that consistently yields the best results in all tasks, whether it employs a purely statistical algorithm or a statistical approach coupled with morpheme enrichment.

3.3.1 Impact of tokenization granularity

		BPE								SentencePiece								
		NACHOS		*PubMed*		*CC100*		*Wikipedia*		*NACHOS*		*PubMed*		*CC100*		*Wikipedia*		
Corpus	**Task**	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	*w/o*	*w/*	ρ
CAS	**CLS**	1.32	1.38	2.20	2.13	1.49	1.49	1.51	1.50	1.32	1.45	2.18	2.15	1.49	1.56	1.51	1.57	*-0.62*
	NER Neg	1.32	1.38	2.20	2.13	1.49	1.49	1.51	1.50	1.32	1.45	2.18	2.15	1.49	1.56	1.51	1.57	*-0.70*
	NER Spec	1.32	1.38	2.20	2.13	1.49	1.49	1.51	1.50	1.32	1.45	2.18	2.15	1.49	1.56	1.51	1.57	*-0.42*
	POS	1.32	1.38	2.20	2.13	1.49	1.49	1.51	1.50	1.32	1.45	2.18	2.15	1.49	1.56	1.51	1.57	*-0.36*
PxCorpus	**CLS**	1.54	1.62	2.26	2.27	1.76	1.72	1.73	1.72	1.54	1.67	2.24	2.30	1.72	1.77	1.77	1.82	*-0.22*
	NER	1.54	1.62	2.26	2.27	1.76	1.72	1.73	1.72	1.54	1.67	2.24	2.30	1.72	1.77	1.77	1.82	*-0.22*
DEFT2020	**STS**	1.41	1.45	2.27	2.24	1.42	1.45	1.43	1.45	1.41	1.49	2.24	2.23	1.41	1.48	1.42	1.49	*-0.47*
	CLS	1.21	1.26	2.13	2.09	1.31	1.34	1.33	1.36	1.20	1.32	2.05	2.04	1.25	1.34	1.29	1.37	*-0.41*
MorFITT	**CLS**	1.38	1.44	2.45	2.40	1.48	1.50	1.49	1.51	1.37	1.50	2.35	2.33	1.46	1.55	1.48	1.57	*-0.82*
E3C	**NER Clinical**	1.30	1.35	2.23	2.17	1.48	1.48	1.50	1.49	1.29	1.43	2.22	2.18	1.48	1.55	1.49	1.56	*-0.59*
	NER Temporal	1.29	1.35	2.22	2.16	1.48	1.48	1.48	1.49	1.29	1.43	2.22	2.18	1.47	1.54	1.48	1.55	*-0.75*
CLISTER	**STS**	1.52	1.59	2.65	2.57	1.73	1.72	1.74	1.72	1.51	1.65	2.56	2.49	1.71	1.77	1.71	1.77	−0.33
DEFT2021	**NER**	1.31	1.37	2.26	2.19	1.48	1.49	1.50	1.50	1.31	1.44	2.19	2.15	1.48	1.55	1.49	1.56	*-0.88*
	CLS	1.50	1.57	2.63	2.56	1.69	1.70	1.71	1.71	1.46	1.61	2.50	2.46	1.64	1.72	1.66	1.74	*-0.11*
ESSAI	**NER Spec**	1.29	1.34	2.20	2.14	1.42	1.43	1.45	1.45	1.29	1.41	2.21	2.16	1.41	1.49	1.46	1.52	*-0.68*
	POS	1.28	1.33	2.19	2.13	1.41	1.42	1.44	1.44	1.28	1.41	2.19	2.15	1.40	1.48	1.44	1.51	*-0.61*
	NER Neg	1.28	1.33	2.19	2.13	1.41	1.42	1.44	1.44	1.28	1.41	2.19	2.15	1.40	1.48	1.44	1.51	*-0.69*
	CLS	1.28	1.34	2.20	2.14	1.42	1.43	1.45	1.46	1.28	1.41	2.20	2.16	1.41	1.49	1.45	1.52	*-0.02*
QUAERO	**NER Medline**	1.53	1.63	2.35	2.26	1.78	1.78	1.77	1.78	1.52	1.76	2.36	2.35	1.77	1.89	1.76	1.89	*-0.77*
	NER EMEA	1.30	1.34	2.14	2.12	1.44	1.46	1.49	1.51	1.30	1.39	2.06	2.04	1.45	1.51	1.50	1.56	*-0.28*
MANTRAGSC	**NER EMEA**	1.33	1.40	2.47	2.41	1.49	1.51	1.50	1.52	1.32	1.43	2.33	2.30	1.46	1.53	1.49	1.55	*-0.63*
	NER Medline	1.89	2.01	2.84	2.70	2.06	2.13	2.14	2.14	1.89	2.09	2.84	2.78	2.06	2.22	2.10	2.22	*-0.64*
	NER Patents	1.54	1.59	2.34	2.30	1.61	1.63	1.59	1.62	1.43	1.52	2.20	2.20	1.50	1.58	1.51	1.60	*0.06*
Average per model		1.39	1.45	2.30	2.25	1.54	1.55	1.56	1.56	1.38	1.51	2.26	2.24	1.52	1.60	1.55	1.62	*-0.48*
Relative Difference (%)		0.0	4.5	65.9	61.8	11.2	11.8	12.3	12.6	−0.7	8.9	62.8	61.1	9.9	15.5	11.7	16.9	

Table 3.2: Average number of sub-word units per word for each tokenization strategy and data source training. Their Pearson correlation (ρ) with each task performance is reported (last column). Cells colored in red correspond to lower-performing models, while those in green represent higher-performing ones. The last row represents the relative difference in terms of average subwords per word compared to the NACHOS BPE without a morpheme baseline. *w/o* and *w/* denote models without and with morphemes.

To assess the impact of tokenization granularity, Table 3.2 presents the average number of sub-word units per word for each tokenization strategy and data source used in the studied tasks. While deriving overarching conclusions from these results can be challenging, we calculated Pearson correlation (ρ) between models' performances on the downstream tasks from Table 3.1 and the corresponding average number of sub-word units per word. These correlation scores range from -1 to $+1$, where -1 indicates a complete negative linear correlation, 0 represents no correlation, and $+1$ signifies a strong positive correlation. In the context of tokenization, a negative correlation implies that fewer subword units are associated with higher scores, while a positive correlation suggests that more subword units are linked to higher scores.

Overall, we observe in Table 3.2 an average ρ correlation of -0.48 between tasks and models, indicating that, in general, higher performance scores tend to be associated with fewer subword units. To our knowledge, this is the first time such a correlation has been

experimentally demonstrated. However, it's important to note that this correlation varies across the targeted tasks. Tasks like CLS show correlation close to zero, suggesting that they are less affected by the granularity of tokenization. In contrast, STS and sequence-to-sequence tasks, particularly NER, appear to be more influenced by tokenization granularity, likely due to their heavy reliance on immediate context for making predictions.

While the RoBERTa model's embeddings capture semantic meaning and the encoder module captures contextual information [243], we aimed to determine whether the observed correlations are attributed to a specific part of this architecture. To investigate this, we isolated and froze the embeddings and/or encoder of our BERT-based model, based on the NACHOS SentencePiece, during fine-tuning for various tasks. The experimental approach, as detailed in Table 3.3, involved several stages. Initially, we established a baseline for each task with no frozen components. Subsequently, we conducted experiments by freezing only the embedding layer, only the encoder, and both the embeddings and encoders. Our findings indicate a stronger dependence on RoBERTa's encoder for tasks such as POS tagging and STS, in contrast to other tasks, which corroborate the context dependency as an explanation for the correlation scores between segmentation granularity and models' performances for these tasks, but not for NER.

	CAS POS	PxCorpus NER	PxCorpus CLS	CLISTER STS
🔥 **Full Fine-tuning**	97.10	96.10	94.82	0.61
❄️ **Embedding**	97.03 ↓ 0.07	96.10 ↑ 0.00	94.73 ↓ 0.09	0.62 ↑ 1.63
❄️ **Encoder**	65.97 ↓ 32.05	83.95 ↓ 12.64	84.78 ↓ 10.58	0.45 ↓ 26.22
❄️ **Embedding + Encoder**	60.04 ↓ 38.16	79.62 ↓ 17.14	84.78 ↓ 10.58	0.44 ↓ 27.86

Table 3.3: Performance and relative loss (in %) of the PLMs based on SentencePiece NACHOS without morpheme with parts of the models being frozen.

As shown in Table 3.2, higher performance scores are associated with fewer subword units. To gain a linguistic perspective on how tokenization strategies behave, we analyzed the segmentation of 150 biomedical terms equally distributed across cardiology, dermatology, obstetric-gynecology, and ophthalmology, as presented in Table 3.4. Most models, except for those using SentencePiece NACHOS, struggle to precisely align with the official morphological segmentation established by the Académie Française (French Academy). However, upon closer examination, it is evident that these models often come very close to the desired segmentation. While the segmentations may exhibit slight variations, such as the relocation of a letter from one token to another, they maintain the same number of tokens as the official morphological segmentation. This observation is further supported when we analyze actual tokenizer outputs (see Table 3.5) and assess the segmentation statistics in Table 3.4. For example, BPE NACHOS tokenizes the term "ophtalmoscope" into the units "ophtalm oscope," whereas the morphological segmentation should be "ophtalmo scope," a segmentation achieved by its morpheme-enriched counterpart.

					Type of errors	
			EM*	Exact # Tok.	Under Seg.	Over Seg.
BPE	*NACHOS*	*w/o*	21.3	41.3	9.3	49.3
		w/	34.6	50.0	6.0	44.0
	PubMed	*w/o*	2.6	12.0	2.6	85.3
		w/	17.3	28.6	2.6	68.6
	CC100	*w/o*	8.0	28.0	2.6	69.3
		w/	23.3	38.6	2.6	58.6
	Wikipedia	*w/o*	8.6	24.6	3.3	72.0
		w/	22.0	36.6	4.6	58.6
SP	*NACHOS*	*w/o*	56.6	74.6	7.3	18.0
		w/	61.3	70.6	2.6	26.6
	PubMed	*w/o*	14.6	26.6	2.6	70.6
		w/	32.0	42.0	2.6	55.3
	CC100	*w/o*	24.0	42.0	4.0	54.0
		w/	36.6	49.3	2.6	48.0
	Wikipedia	*w/o*	18.0	42.0	3.3	54.6
		w/	34.0	54.0	4.6	41.3

Table 3.4: The average Exact Match (EM*) and portion of terms aligned with the official segmentation length (Exact # Tok.), both in %, are based on the gold segmentation from 150 biomedical terms. Both last columns are referring to the portion of terms suffering from under- and over-segmentation. *w/o* and *w/* denote without and with morphemes respectively. SP stands for SentencePiece.

In Table 3.4, we observed various types of errors in segmentation, with the most common issue being over-segmentation of units that are not present in our biomedical lexical morphemes list. This over-segmentation results in smaller, more numerous, and sparser tokens, which can impact the efficiency of pre-training. The reduced frequency of tokens and the faster filling of RoBERTa's 512-token context window with less meaningful tokens can be problematic.

Finally, Table 3.4 reveals an interesting distinction between BPE and SentencePiece using NACHOS training data. SentencePiece outperforms BPE in achieving segmentations that closely resemble correct ones, both in terms of the number of tokens and their semantic accuracy. SentencePiece excels at matching correct segmentations, particularly for medical terminology, in 56.6% of cases without morphemes and 61.3% when morphemes are used, while BPE NACHOS achieves only 34.6% accuracy.

Base	*cancérigène*	*ophtalmoscope*	*angiographie*
Correct	**cancér i gène**	**ophtalmo scope**	**angio graphie**
BPE Wiki	c anc éri gène	oph tal mos cope	ang i ographie
BPE PubMed	can c é rig è ne	o ph tal m oscope	angi ograph ie
BPE NACHOS	cancé rig ène	ophtalm oscope	angiographie
SentencePiece NACHOS	cancérigène	ophtalm oscope	angiographie
BPE NACHOS +Morpheme	**cancér i gène**	**ophtalmo scope**	**angio graphie**
SentencePiece NACHOS +Morpheme	**cancér i gène**	**ophtalmo scope**	**angio graphie**

Table 3.5: Instances of tokenization juxtaposed with their correct segmentation.

3.3.2 Impact of morphemes

One of our primary objectives was to approximate the correct morphological segmentation of words in the French biomedical language. Our analysis reveals that tokenizers, such as BPE and SentencePiece trained on NACHOS, enriched with morphemes, can often achieve this goal. Notably, SentencePiece NACHOS enriched with morphemes achieved the best performance, with a 61.3% exact match. Our morpheme-enriched approach offers the advantage of obtaining a tokenization that closely resembles what could be achieved through a complex rule-based method. This approach is easily adaptable to other languages with a list of lexical morphemes and similar principles.

As shown in Table 3.1, the introduction of morphemes (*w/*) may lead to performance enhancements in approximately 25% of the studied downstream tasks. However, it is noteworthy that the best results are primarily achieved by classical statistical tokenizers, BPE and SentencePiece, when not using morphemes, and when trained on our biomedical-specific data, NACHOS. This observation is intriguing because NACHOS-based tokenizers inherently contain a higher proportion of morphemes, as shown in Table 3.6, which presents the portion of correct morphemes already present in the tokenizers without introducing additional morphological information based on their length ranges. This suggests that introducing morphemes and other forms of morphological knowledge, such as grammatical endings, may have a more substantial impact in contexts that do not align directly with the target domains and languages. However, we can note that the results of this method are inconsistent and do not ensure an overall performance boost across all models or tasks.

Furthermore, it is worth noting that morphemes are often already present in the tokenizers in their complete form, as illustrated in Table 3.6, or with minor modifications based on token probabilities, as shown in Table 3.5. Notably, tokenizers based on NACHOS contain a significantly higher percentage of morphemes, with 47.23% for BPE and 43.59% for SentencePiece. Conversely, the source with the fewest morphemes is CC100, with percentages of 34.77% for BPE and 35.64% for SentencePiece. This observation aligns with the fact that CC100 has fewer connections to both the target language and domain.

		Coverage of the morphemes (%)			
Tokenizer	**Source**	**1 - 3**	**4 - 6**	**7 - 10**	**Global**
BPE	***NACHOS***	83.33	45.38	31.00	47.23
	PubMed	65.15	39.32	15.00	38.06
	CC100	78.78	34.46	7.00	34.77
	Wikipedia	87.87	34.95	10.00	36.67
SP	***NACHOS***	83.33	41.01	28.00	43.59
	PubMed	60.60	37.13	14.00	35.81
	CC100	83.33	34.70	8.00	35.64
	Wikipedia	93.93	37.37	12.0	39.44

Table 3.6: Percentage of the morphemes already present in the tokenizers' vocabularies per range of morpheme lengths. SP stands for SentencePiece.

In general, we observe that despite the significant improvement in segmentation quality (as shown in Table 3.4), tokenizers enriched with morphemes do not exhibit a strong

correlation with the results achieved in downstream tasks, as evident in Table 3.1. The ability to deliver satisfactory results despite encountering suboptimal segmentations, as seen in the case of PubMed, which frequently over-segments words, underscores the robustness of RoBERTa's architecture in handling noise and its capacity to compensate for such challenges.

3.3.3 Impact of data sources

As indicated in Table 3.1, the average performance across tasks demonstrates a significant impact of the training data source on the results obtained by the models. It becomes apparent that using data that is more suitable for the target language, even if it originates from various domains such as Wikipedia and CC100, is more effective than utilizing data from the target domains but from a different language. This is particularly evident in the CLS, NER, and STS tasks, where BPE PubMed achieves an average of 70.16% for classification, 0.63 MSE for STS, and 62.62% for NER, whereas CC100 outperforms with 74.14%, 0.67 MSE, and 64.62%, respectively.

The decrease in performance from PubMed can be attributed to over-segmentation, as seen in Table 3.2. This over-segmentation is primarily due to the significant differences between the data used to build the tokenizer and the language of the model's pre-training. These differences stem from distinct lexicons, writing styles, and morphological structures in French compared to English, particularly for specialized words like "Péricardite" (French) and "Pericarditis" (English), or "Orthophoniste" (French) and "Speech Therapist" (English). Furthermore, variations in alphabets, such as special French characters like "é" or "è," can lead to token sparsity when encountered in positions not seen during tokenizer construction on PubMed. This results in a lack of both language and domain-specific information for French, as only limited tokens can be used to form sentences.

Some data sources are surprisingly less affected by the introduction of morphemes. For instance, the CC100 source is not positively impacted by morphemes, despite having a lower proportion of morphemes in its original version, as shown in Table 3.6. This behavior may be explained by the increased granularity introduced by morphemes, which reduces the probabilities of other tokens appearing. This can lead to a poorer representation of words.

3.3.4 Tokenization Statistics

Our analysis of different tokenization approaches in the Table 3.7 reveals several key patterns in tokenizer encoding capabilities. To quantify these patterns, we employ two key metrics: Shannon entropy and fertility. The Shannon entropy $H(X)$ measures the uncertainty in token distribution, defined as $H(X) = -\sum_i p_i \log_2 p_i$ where p_i represents the probability of token i in the vocabulary. We also compute a normalized version of the entropy by dividing by $\log_2(vocab\_size)$, which gives a value between 0 and 1, where 1 indicates perfectly even token distribution. Meanwhile, fertility F is computed as $F = \frac{|T|}{|W|}$, where $|T|$ is the total number of tokens and $|W|$ is the total number of words in the corpus, providing insight into tokenization efficiency.

BPE tokenizers without morphemes generally achieve slightly higher Shannon entropy values (ranging from 0.5795 to 0.6514) compared to those with morphemes (0.5543 to 0.6392), indicating more uniform token distribution.

Tokenizer	Shannon Entropy	Fertility	Token Ratio
	BPE w/ Morphemes		
Wikipedia	0.6331	1.8732	1.20x
PubMed	0.5741	2.6378	1.69x
CC100	0.6326	1.8618	1.19x
NACHOS	0.6263	1.7370	1.11x
	SentencePiece w/ Morphemes		
Wikipedia	0.6392	1.8671	1.20x
PubMed	0.5543	2.5886	1.66x
CC100	0.6369	1.8714	1.20x
NACHOS	0.6347	1.7389	1.11x
	BPE w/o Morphemes		
Wikipedia	0.6514	1.7944	1.15x
PubMed	0.5795	2.6474	1.70x
CC100	0.6499	1.7832	1.14x
NACHOS	0.6427	1.5879	1.02x
	SentencePiece w/o Morphemes		
Wikipedia	0.6422	1.7627	1.13x
PubMed	0.5542	2.5631	1.64x
CC100	0.6392	1.7552	1.12x
NACHOS	0.6304	1.5612	1.00x

Table 3.7: Comparison of Tokenizer Metrics based on a set of 509 french clinical cases extracted from The PanAfrican Medical Journal.

The NACHOS-trained tokenizers consistently show the lowest fertility values (approximately 1.56-1.59 tokens per word), suggesting more efficient tokenization for medical texts. In contrast, PubMed-trained tokenizers exhibit significantly higher fertility (around 2.56-2.65 tokens per word), likely due to the specialized medical terminology in this dataset.

SentencePiece tokenizers perform similarly to their BPE counterparts, with only minor differences in both entropy and fertility metrics.

Overall, the BPE-HF-NACHOS-FR tokenizer without morphemes achieves the best balance, with the highest Shannon entropy (0.6427) and lowest fertility (1.5879) among non-morpheme tokenizers, making it particularly suitable for medical NLP tasks.

The token ratio analysis reveals significant cost implications for real-world deployments. For instance, PubMed-trained tokenizers require approximately 1.64-1.70 times more tokens compared to the most efficient NACHOS tokenizer. This difference becomes substantial when considering charging input tokens at a fixed cost by millions of tokens, for a medical text of 1000 words, using a PubMed tokenizer would cost about 70% more compared to using the NACHOS tokenizer. This cost difference becomes even more pronounced in large-scale medical NLP applications where millions of tokens are processed daily.

Furthermore, the higher token counts directly impact the context window utilization

in transformer models like BERT or GPT. With the same context window size, PubMed tokenizers can fit approximately 40% fewer words compared to NACHOS tokenizers. This limitation becomes particularly critical in medical applications where maintaining context is essential for accurate diagnosis and treatment recommendations. The token efficiency also significantly impacts inference time, as transformer models process tokens sequentially through their attention mechanisms. Using the more efficient NACHOS tokenizer can reduce inference time by up to 40% compared to PubMed tokenizers, directly translating to faster response times in clinical applications. This performance improvement becomes crucial in real-time medical applications and can substantially reduce computational resource requirements for batch processing of medical records. While, starting from English models like PubMedBERT provides valuable bootstrap knowledge, our analysis suggests potential long-term limitations

3.4 Conclusion

In this study, we conducted a comprehensive investigation into the influence of various word tokenization strategies within a BERT-based masked language model across diverse French biomedical NLP tasks. Notably, we observed that existing methods for tokenizing biomedical text often fall short of aligning with morphological rules and how humans learn these specialized terms. This suboptimal segmentation can impact the agglutinating nature of biomedical terminology. To assess the effects of this segmentation on downstream applications, we developed a set of novel biomedical tokenizers that adhere more closely to morphological rules. These tokenizers combine various automatic tokenization approaches and vocabularies to enrich segmentation with morphemes. We employed these enhanced tokenizers in the pre-training of multiple RoBERTa-based models, which we then evaluated across a wide array of 23 French biomedical tasks, including POS, NER, STS, and CLS.

Our findings show that integrating morphemes into automatic tokenization approaches can achieve parity or improve performance in certain tasks, such as NER and POS tagging. However, this enhancement is not consistent across all tasks. While there is a correlation between segmentation granularity and downstream task performance, we also observe that pre-training processes exhibit robustness to suboptimal tokenization, yielding surprisingly good results even with very short and sparse subword units. To conclude, our study reveals that achieving optimal tokenization involves a combination of factors, including minimizing word segmentation and having access to domain-specific data in the target language.

Part III

Auto-regressive Models

Chapter 4

A zero-shot and few-shot study of instruction-finetuned large language models applied to clinical and biomedical tasks

The emergence of Large Language Models (LLMs) represents a significant paradigm shift in Natural Language Processing. As detailed in previous chapters, achieving state-of-the-art performance with specialized Masked Language Models (MLMs) like DrBERT (Chapter 1) required not only extensive domain-specific pre-training but also fine-tuning on relatively large annotated datasets. This reliance on labeled data presents a significant bottleneck in healthcare, where annotations are notoriously expensive to produce, difficult to obtain due to privacy constraints, and consequently rarely open-sourced. The advent of powerful, general-purpose LLMs, capable of performing well with minimal adaptation, therefore raised a crucial strategic question: Is it always necessary to build specialized models from the ground up, or can existing, instruction-tuned LLMs perform sufficiently well in specialized domains with minimal adaptation ?

The emergence of LLM represents a significant paradigm shift in NLP. These models introduced a new paradigm called instruction-tuning, where the model is trained to follow natural language instructions that specify the desired task and expected output format. As detailed in previous chapters, achieving state-of-the-art performance with specialized MLM like DrBERT (Chapter 1) required not only extensive domain-specific pre-training but also fine-tuning on relatively large annotated datasets. This reliance on labeled data presents a significant bottleneck in healthcare, where annotations are notoriously expensive to produce, difficult to obtain due to privacy constraints, and consequently rarely open-sourced. The advent of powerful, general-purpose instruction-tuned LLMs, capable of performing well with minimal adaptation, therefore raised a crucial strategic question: Is it always necessary to build specialized models from the ground up, or can existing, instruction-tuned

LLMs perform sufficiently well in specialized domains with minimal adaptation ?

This chapter addresses this question by documenting one of the first comprehensive studies into the capabilities of early instruction-tuned LLMs within the biomedical field, a work that was subsequently published at LREC-COLING 2024 [172]. At the time of this research, the potential of LLMs in medicine was still largely underexplored, and this study was essential to understand their strengths and limitations. The investigation was conducted on English tasks, as suitable evaluation benchmarks for generative models in French were not yet available. The findings from this exploratory work were foundational, directly motivating the development of BioMistral, which will be presented in the next chapter (Chapter 5).

The primary research objectives of this initial study were to:

- Assess the out-of-the-box performance of general-purpose LLMs on specialized biomedical tasks compared to a fine-tuned, domain-specific MLM (PubMedBERT).
- Determine which task types (e.g., Question Answering, NER, Classification) are well-suited for these models in zero-shot and few-shot scenarios.
- Evaluate whether a single LLM excels across all tasks or if performance is model-dependent.
- Understand the limitations of generalist LLMs, thereby building the case for developing specialized generative models for the medical domain.

To answer these questions, this chapter presents a comprehensive evaluation of four state-of-the-art instruction-tuned LLMs (ChatGPT, Flan-T5 UL2, Tk-Instruct, and Alpaca) on a diverse benchmark of 13 English clinical and biomedical tasks. This evaluation suite was designed to cover a wide spectrum of real-world NLP challenges, including classification of medical texts (e.g., identifying smoker status in clinical notes or classifying public health claims), extractive and abstractive question answering over biomedical literature (e.g., BioASQ, MedMCQA), relation extraction to identify gene-disease associations, natural language inference to determine logical relationships between sentences (e.g., SciTail, MedNLI), and fine-grained named-entity recognition of chemicals, diseases, and other medical concepts (e.g., BC5CDR, NCBI-disease). Our main contributions from this pivotal study are:

- A rigorous zero-shot and few-shot evaluation of four prominent, early instruction-tuned LLMs on this comprehensive benchmark of 13 biomedical tasks.
- A direct comparison against a powerful, fully fine-tuned domain-specific baseline (PubMedBERT), establishing a clear performance benchmark for current generative models.
- A detailed analysis of model performance across different task formats, identifying their current strengths (e.g., QA) and areas requiring further improvement (e.g., RE).

- The exploration of novel inference strategies, including a *Recursive Chain-of-Thought* (RCoT) method, to adapt generative models for structured prediction tasks like NER.

Ultimately, this chapter chronicles the beginning of our journey into generative AI for medicine. The results demonstrate that while general-purpose LLMs show remarkable promise, their limitations in handling the nuances of biomedical text underscore the need for specialized models. This conclusion serves as the direct motivation for the work presented in the subsequent chapter: the creation of BioMistral, a healthcare LLM designed to overcome these identified shortcomings.

4.1 Experimental Protocol

In this section, we describe the models utilized and the datasets used to benchmark the various models.

4.1.1 Studied Models

Our evaluation involves four distinct generic LLMs (ChatGPT, Flan-UL2, Tk-Instruct and Alpaca) and a specific biomedical masked language model (PubMedBERT) for comparison purposes.

Flan-T5 UL2 abbreviated to Flan-UL2, is an encoder-decoder model based on UL2 20B parameters model [261] and was fine-tuned using the Flan instruction tuning tasks collection [60].

Tk-Instruct is based on the T5 encoder-decoder model [240] and has been fine-tuned on the 1,600+ NLP tasks from the Super-NaturalInstructions dataset [283]. In our study, we chose the 3B parameter setting, since our preliminary comparison with Flan-T5-XL [58] using the 3B parameter setting showed that Tk-Instruct performed better on QA tasks, which is considered to be one of the most suited tasks for LLMs.

ChatGPT is built upon GPT-3.5 Turbo, fine-tuned with a set of proprietary instructions, and continuously refined through *Reinforcement Learning from Human Feedback* (RLHF) techniques. Access to its weights is restricted, and the model can only be accessed via a paid API. These restrictions raise privacy concerns regarding its application in medical contexts, and it cannot ensure that the evaluated data has not been previously encountered.

Stanford Alpaca is built upon LLaMA with 7B parameters [267] and utilizes a dataset of 52K instructions, which were automatically generated in the style of self-instruct using OpenAI's text-davinci-003 model [282]. Due to its base model and data sources, it is exclusively intended for academic research purposes and non-commercial use.

PubMedBERT is a biomedical-specific BERT-based model with 110M parameters [110]. It was trained entirely from scratch on the 3.1 billion words of the PubMed corpus. We chose it as our baseline for comparison with the zero-shot and few-shot performance of generative models.

4.1.2 Downstream Evaluation Tasks

We conducted an evaluation of the models' capabilities by encompassing the test set of the 13 diverse tasks listed in Table 4.1. These tasks were chosen to facilitate a comprehensive assessment spanning both clinical and biomedical domains, including tasks suitable for both generative and classical model evaluations.

Task	Dataset	Eval	Metric	Reference
CLS	HoC	Test	F1-measure	[22]
	LitCovid	Test	F1-measure	[47]
	PubHealth	Test	Accuracy	[210]
	N2C2 2006 Smokers	Test	Accuracy	[272]
QA	BioASQ 7b	Test	Accuracy	[268]
	MedMCQA	Dev	Accuracy	[222]
	SciQ	Test	Accuracy	[287]
	Evidence Inference 2.0	Test	Accuracy	[82]
RE	GAD	Test	Accuracy	[36]
NLI	SciTail	Test	Accuracy	[150]
	MedNLI	Test	Accuracy	[251]
NER	BC5CDR	Test	F1-measure	[180]
	NCBI-disease	Test	F1-measure	[85]

Table 4.1: List of evaluation tasks and their metrics. CLS: Classification, QA: Question Answering, RE: Relation Extraction, NLI: Natural Language Inference, NER: Named-Entity Recognition.

4.1.3 Evaluation of Generative Outputs

Evaluating the outputs of generative models presents a challenge due to their free-text nature, which may not necessarily conform to a predefined set of classes. Instead, we are confronted with noisy outputs that may contain correct answers. To address this challenge, we manually developed parsing scripts tailored to each task and model, aligning them with their respective output styles. This approach enables us to capture most of the answers and compute metrics that can be compared with our baseline model (PubMedBERT).

4.1.4 Instruction Format

Previous studies [284, 143, 206] have demonstrated the effectiveness of using task-specific prompts for each model. Consequently, we chose to construct the input instruction prompt by concatenating three elements: (1) an instruction that outlines the task, describes the nature of the data, and specifies our expectations from the model, (2) the input argument, which provides essential information for the task, and (3) the constraints on the output space, which guide the model during output generation. Lastly, the output serves as a reference point during the few-shot strategy evaluation.

4.1.5 Few-shot Examples using Semantic Retriever

To enhance few-shot performance compared to randomly sampled examples, we introduced an additional retrieval module based on Sentence-Transformers [242]. The objective is to identify the k most semantically similar examples from the training set. To accomplish this, we first populate a vector space with sentence representations of each individual instruction prompt from the training set, obtained using a pre-trained and fixed PubMedBERT [110] model. Subsequently, we compute the cosine distance between the query of the current test instance and all the elements within the vector space to retrieve the top k closest examples. In our case, we set the value of k to 5.

4.1.6 Recursive Chain-of-Thought

We performed NER using two inference methods. The first one is based on the method introduced by [304] and can only be applied using ChatGPT. It consists of giving the model a sequence of words separated by double vertical bars for word separation and single vertical bars for the separation between words and labels. For the second method, we introduce a method called RCoT. It is very close to human reasoning and works for all the generative models we have tried. It is derived from the CoT concept [284] and the work of [283]. It involves iterating over the sequence of tokens and giving the current state of the prediction as input to the model, asking for the generation of the label of the Nth token. This method guarantees an entity for each token of the sequence and prevents forgotten tokens during generation. However, the only drawback we have been able to identify with this method is its very high computation cost due to its $\mathcal{O}^N$ complexity, with N being the number of tokens in the sequence, compared to the method used for ChatGPT, which performs at $\mathcal{O}^1$ complexity.

4.2 Results and Discussions

Task	Dataset	ChatGPT		Flan-UL2		Tk-Instruct		Alpaca		PubMedBERT
		zero-shot	5-shot	zero-shot	5-shot	zero-shot	5-shot	zero-shot	5-shot	
CLS	HoC	62.24	38.34	56.36	54.86	50.77	25.48	1.21	38.78	**82.75**
	LitCovid	67.20	72.77	51.48	46.95	36.42	57.49	1.58	64.09	**90.60**
	PubHealth	63.20	66.29	72.46	50.53	53.70	66.04	52.80	55.64	**75.39**
	N2C2 2006 Smokers	NaN	NaN	22.12	42.31	16.35	37.50	10.57	31.73	**60.58**
QA	BioASQ 7b	89.24	**92.03**	90.97	91.64	88.09	86.36	79.05	79.82	73.39
	MedMCQA	48.91	**56.37**	41.05	43.34	33.85	33.18	24.91	29.50	38.15
	SciQ	90.10	**93.50**	87.00	88.40	55.30	47.00	24.90	36.80	74.20
	Evidence Inference 2.0	59.98	63.83	66.45	65.06	41.33	38.79	32.49	**94.18**	65.47
RE	GAD	47.75	52.25	49.81	53.37	48.88	57.87	51.12	57.68	**79.78**
NLI	SciTail	73.57	65.62	**93.51**	92.66	57.53	71.31	39.60	40.26	**93.51**
	MedNLI	NaN	NaN	77.00	79.18	33.19	34.81	33.47	34.45	**83.76**
NER	BC5CDR	92.12	93.12	68.26	83.32	84.54	83.23	82.11	84.07	**97.65**
	NCBI-disease	90.97	92.27	90.75	87.65	87.91	87.50	11.58	92.27	**98.72**

Table 4.2: 0- and 5-shot versus finetuning evaluation on clinical and biomedical tasks. Bold values are the highest scores obtained for the task and in underlined the seconds ones. Not allowed experiments are replaced by `NaN`.

Table 4.2 reports performance obtained on each task by the studied LLMs in zero- and few-shot scenarios, as well as PubMedBERT fine-tuned. Results are reported by taking the best run out of four.

Zero-Shot Scenario Compared to PubMedBERT, the zero-shot scenario results show a clear deficit for the generative models on all the tasks except for QA, in which LLMs obtain better performance. ChatGPT and Flan-T5 UL2 particularly perform better than Tk-Instruct and Alpaca on average, except for the GAD dataset (RE task) for which Alpaca reaches the best performance. We can also observe extremely poor performance from Alpaca in the zero-shot scenario on the two CLS tasks (HoC and LitCovid). These low scores are attributed to the model generating hallucinated responses, including the label *evading growth suppressors* across the entire test set of HoC. However, this behavior does not appear to occur in the few-shot scenario, where the model appears to comprehend our expectations.

Few-Shot Capabilities Unlike the zero-shot scenario, the few-shot inference (5-shots in our experiments) shows impressive behavior. The biggest absolute gains are obtained using Alpaca, which seems to perform much better in few-shot scenarios on all tasks. We suspect this behavior to be correlated with Alpaca's training data, which does not contain many similar instructions for the tasks we are trying to tackle, allowing it to better understand what we are asking when confronted with dissimilar examples. ChatGPT also benefits from the additional knowledge to further improve the already good results, especially on QA tasks. Flan-T5 UL2 appears to be less affected by the additional context overall, except for the BC5CDR and N2C2 2006 Smokers tasks.

4.3 Conclusion

In this study, we have demonstrated that generic LLMs are capable of capturing medical knowledge and performing exceptionally well in zero- and few-shot scenarios, despite having no prior exposure to the tasks. Although open-source models such as Flan-T5 UL2 are gradually approaching their closed-source counterparts, such as ChatGPT, their performance still lags behind. We suggest that developing domain-specific models, fine-tuned on a diverse set of tasks and specialized instruction prompts, could help bridge the gap with more robust and performant proprietary models. We also note that domain-specific BERT models remain a viable option, but require a significant amount of data for fine-tuning on targeted languages and tasks. However, BERT-based models offer much lower computational costs compared to LLMs, which could be a significant obstacle to developing models in the healthcare domain.

Chapter 5

BioMistral: A collection of open-source pretrained LLM for medical domains

As concluded in the previous chapter (Chapter 4), the advent of Large Language Models (LLMs) represents a paradigm shift. We verified their remarkable capabilities in zero-shot and few-shot learning scenarios, demonstrating that they could outperform traditional MLMs on several complex tasks. However, that study also revealed a performance gap on several key medical tasks, where powerful general-purpose LLMs were still outperformed by older encoder-decoder or MLM models that had been specifically fine-tuned on similar tasks or domains specific data. This shown room for improvement for a new generation of specialized LLMs models.

This chapter answers that call by introducing BioMistral. At the time of its development, this work was the first to adapt the open-source Mistral model for the healthcare domain, positioning it as a significant step forward from existing models in the same parameters range (MedAlpaca, PMC-LLaMa, MediTron and BioMedGPT-LM). The project was driven by several key research questions:

- Can continual pre-training of a state-of-the-art base LLM on high-quality medical corpora effectively infuse it with specialized knowledge and significantly boost its performance on domain-specific tasks as it does on French healthcare with DrBERT in Chapter 1?
- How do advanced model merging techniques (e.g., TIES, DARE, SLERP) perform in a medical context? Can these techniques combine the respective strengths of a generalist base model and a domain-adapted one, and can the resulting model outperform its individual constituents ?
- What is the practical impact of quantization methods (e.g., AWQ, BnB) on a specialized LLM ? Can we dramatically reduce the computational footprint of these models

to make them accessible for a wider public without catastrophic forgetting and performance loss ?

- How well does a healthcare adapted LLM on English data generalize to other languages, and what does this imply for developing multilingual medical models?

Our work, published at ACL 2024 [168], makes the following contributions:

- We develop and release BioMistral, a suite of 7B-parameter LLMs adapted for the healthcare domain, on open-sourced data from PubMed Central Open-Access (PMC OA) . The collection includes the base adapted model, along with several merged and quantized variants, all of them publicly accessible in open-source.

- We conduct a large-scale evaluation of BioMistral across 10 English medical Question-Answering benchmarks, demonstrating state-of-the-art performance among open-source models. We further assess its generalization capabilities by evaluating it on the same benchmarks translated automatically into 7 other different languages.

- We present a systematic analysis of leading model merging and quantization techniques, providing novel insights into their effectiveness for creating specialized LLMs and offering a practical guide to performance-versus-efficiency trade-offs.

- We release all of our models, evaluation datasets, and quantized versions on the Hugging Face Hub[1]. All code for data processing, pre-training, SFT, merging, quantization, and evaluation is open-sourced on GitHub to ensure full reproducibility[2].

- To further improve accessibility, we release Tchat on GitHub[3], an open-source, multi-turn conversational web interface with an integrated speech-to-text module for voice-based interaction, allowing users to easily interact with textual and speech modalities with BioMistral and other LLMs with minimal configuration.

5.1 BioMistral

In this section, we present the modules that facilitated the construction of BioMistral 7B. We first develop our training corpus (Section 5.1.1) used during further pre-training. We then present the model adaptation method (Section 5.1.2). Finally, we discuss the approaches for model merging (Section 5.1.3) and expose the employed quantization strategies (Section 5.1.4).

[1] https://huggingface.co/BioMistral
[2] https://github.com/BioMistral/BioMistral
[3] https://github.com/BioMistral/Tchat

5.1.1 Pre-training Dataset

For LLM adaptation to the medical domain, we selected the PMC Open Access Subset[4] for its comprehensive and freely accessible collection of medical research papers. This choice is guided by the success demonstrated by PMC-LLaMA [294], PubMedBERT [108], and SciFive [231], which have showcased significant enhancements in language modeling for medical applications. Our focus lies on the Commercial Use Allowed subset, encompassing documents licensed under various Creative Commons licenses (CC0, CC BY, CC BY-SA, and CC BY-ND). This subset ensures the reusability of our model's outputs, even for commercial purposes.

In the preprocessing phase, we aim to optimize the dataset for training efficiency while considering hardware limitations. Our pre-training objective involves further pre-training Mistral on a subsample of this corpus, targeting 1.5 epochs within the 20-hour limit of Jean Zay HPC. This decision aligns with insights from the Zephyr model [269], which suggests that observing 1.5 times the corpus adequately enhances model performance, with marginal benefits beyond this threshold. We then meticulously selected 3 billion tokens from this pre-processed PubMed Central corpus, corresponding to roughly 1.47 million documents. The dataset comprises primarily English documents (98.75% of the corpus), with the remaining portion encompassing 9 languages, including Dutch, German, French, and others. Our strategy emphasizes a multilingual dataset approach by prioritizing non-English documents, supplemented with English texts, to ensure a diverse and representative training dataset to meet our 3 billion token target. The raw textual documents undergo pre-processing using the Mistral tokenizer, which includes tokenization and normalization processes.

5.1.2 Model Adaptation

Training details We leverage Mistral 7B Instruct v0.1 [138] as the base model for adaptation due to its design tailored for incorporating instructions in prompts and its capacity for fine-tuning across diverse tasks using limited datasets. Pre-training settings for BioMistral 7B largely align with Mistral 7B Instruct v0.1. For optimization, we employ the AdamW [195] optimizer alongside a cosine learning rate scheduler. Our model architecture inherits the standard transformer architecture from Mistral, including features such as Grouped-Query Attention [2], Sliding Window Attention [27] and Rolling Buffer Cache.

The model also incorporates Rotary Positional Embeddings (RoPE) [256], which encode token positions through rotation transformations applied to the embedding space. RoPE enables the model to effectively capture relative distances between tokens, which is particularly valuable for processing long texts with complex dependencies. Figure 5.1 visualizes these embeddings across Mistral's 2,048-token context window, showing the sinusoidal patterns that vary smoothly across positions:

[4] https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

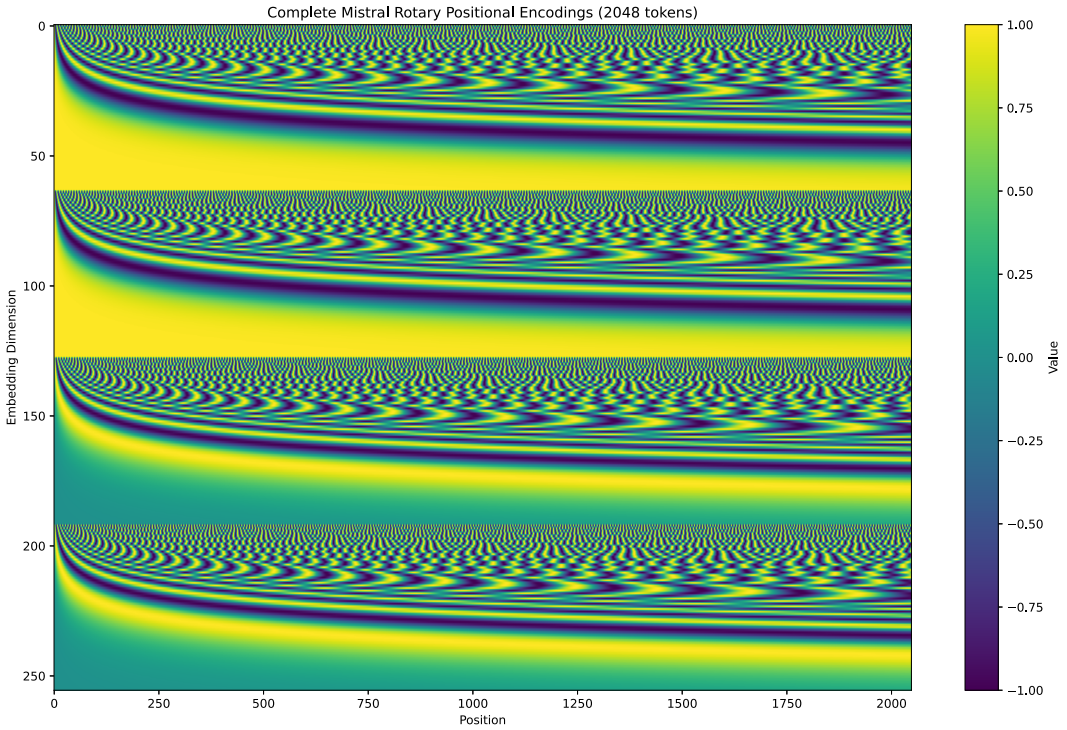


Figure 5.1: BioMistral 7B and Mistral 7B Instruct v0.1 Rotary Positional Encodings (RoPE).

We maintain an input context length of 2,048 tokens across all models, including the quantized versions (see Section 5.1.4), in conjunction with FlashAttention-2 [74]. For optimization, we set the learning rate to 2×10^{-5} with no warmup, a weight decay of 0.01, a gradient accumulation of 2, and a batch size of 16 on the Jean-Zay HPC with 32 NVIDIA A100 80GB GPUs. This configuration allows for a total batch size of 1,024. Due to the model and the AdamW optimizer's inability to fit on a single GPU with BF16 precision, we employ the Fully Sharded Data Parallel distributed learning framework [315].

Improving batching To enhance pre-training efficiency, we introduce a post-tokenization grouping method. This method aggregates variable-sized sequences marked by an end-of-sequence token (</s>) to fill the model's 2,048-token sequences without padding. This reduces the sequence count by 87.88%, subsequently accelerating epoch times. Refer to Appendix 9.5 for pseudo-code detailing the grouping method.

5.1.3 Model Merging

Pre-trained models may lose effectiveness when applied beyond their specific domains [165]. Traditionally, separate models were used for each application [112], increasing complexity and costs. Recent studies suggest merging pre-trained models to enhance performance and out-of-domain generalization [43, 11, 293, 141, 131]. Merging involves combining multiple model parameters without additional training. Methods include averaging model weights or considering permutation invariance [132, 54, 253, 3].

Among these methods, we can cite TIES [300], DARE [305], and SLERP [252]. SLERP merges two models using Spherical Linear Interpolation to allow a smoother transition between model parameters while preventing the significant information loss often encountered with direct averaging of model weights. TIES merges models by creating "task vectors" from each model, isolating unique contributions by subtracting an ancestor base model (e.g., Mistral 7B Instruct). These vectors are then averaged with the base model. Its key improvement over previous methods relies on reducing model interference using sparse vectors and a sign consensus method. DARE enhances TIES by reducing delta parameter redundancy, mainly setting them to zero through random pruning and rescaling while maintaining or improving original model performance.

Exploring model merging in the biomedical domain is particularly interesting since merging a general domain model with a domain-specific one could enhance specialized model adaptability and accuracy across a broader range of applications. The objective of this application in the medical domain is not only to improve general-domain capabilities but also to explore the possibility of emergent reasoning and surpassing the performance of baseline models used for merging.

5.1.4 Quantization

Quantization techniques are pivotal in democratizing LLMs as they enable the execution of LLMs on smaller devices by minimizing memory requirements. In our study, we investigate two core techniques: *Activation-aware Weight Quantization* (AWQ) and *BitsandBytes* (BnB).

AWQ [186] is an advanced quantization method that capitalizes on the insight that weights vary significantly in importance across different channels and layers. Rather than applying uniform quantization to all weights, AWQ identifies and preserves the most critical weights (typically 1% of the total) in higher precision while quantizing the remaining weights to 4 bits. This selective approach is guided by activation magnitudes during calibration, where channels with larger activation values are considered more important. The method employs a scaling technique that adjusts weight distributions to minimize quantization errors while maintaining the overall model structure.

Conversely, BnB quantization is a more straightforward approach that assigns a fixed precision of 4 or 8 bits to the entire model uniformly. BnB implements dynamic quantization with custom CUDA kernels optimized for inference speed, supporting both linear quantization and more sophisticated techniques like blockwise quantization. The 8-bit version uses a two-stage quantization process with outlier detection, while the 4-bit version employs NormalFloat4 (NF4) data type specifically designed for normally distributed weights. Both methods significantly reduce memory footprint (up to 75% for 4-bit) while maintaining reasonable performance degradation.

5.2 Evaluation Protocol

To assess the performance of BioMistral 7B models, we first describe our benchmark of English medical reasoning tasks (Section 5.2.1) and their multilingual translation (Section 5.2.2), before presenting the instruction prompting (Section 5.2.3) and the supervised fine-tuning strategy (Section 5.2.4) employed for the models' evaluation.

5.2.1 Downstream Tasks

To evaluate the performance of the BioMistral 7B model, we selected 10 QA tasks in English from 4 prominent medical corpora (MedQA, MedMCQA, PubMedQA, and MMLU) covering various specialties, including genetics, anatomy, and clinical cases. These datasets encapsulate real-world scenarios encountered by medical professionals, medical school entrance examination formats, and comprehension tests based on PubMed content. The datasets' characteristics are provided in Table 5.1 and 5.2.

	MMLU					
	Clinical KG	**Medical Genetics**	**Anatomy**	**Pro Medicine**	**College Biology**	**College Medicine**
Answer options	A / B / C / D	A / B / C / D	A / B / C / D	A / B / C / D	A / B / C / D	A / B / C / D
Train / Valid. / Test	0 / 0 / 265	0 / 0 / 100	0 / 0 / 135	0 / 0 / 272	0 / 0 / 144	0 / 0 / 173
Words / Questions	11.09	12.34	13.65	105.46	22.40	48.84
Context	×	×	×	×	×	×

Table 5.1: Description of the MMLU question-answering tasks. The reference to "Clinical KG" denotes "Clinical Knowledge".

	MedQA	**PubMedQA**	**MedMCQA**
Answer options	A / B / C / D / (E)	Yes / No / Maybe	A / B / C / D
Train / Valid. / Test	10178 / 1272 / 1273	211269 / 500 / 500	146257 / 36565 / 4183
Words / Questions	118.16	13.08	14.05
Context	×	✓	×

Table 5.2: Description of additional medical question-answering tasks. Only PubMedQA incorporates context information within the prompt (see Section 5.3.7).

5.2.2 Multilingual Evaluation

While the biomedical language models have been extensively evaluated in languages such as English [177, 49], Chinese [39, 303], French [264, 165] or Spanish [42], their performance in languages beyond their own remains relatively understudied. This limited multilingual evaluation can be attributed to the scarcity of biomedical tasks available in languages other

than English. To address this gap, we conducted a multilingual evaluation using GPT-3.5 Turbo (version 1106) automatic translation via the OpenAI API. We translated our benchmark into 7 languages: Spanish, German, Portuguese, Russian, French, Arabic, and Chinese. Despite the challenges posed by automatic translation, these tools have shown remarkable improvement in recent years [212], enabling cost-effective multilingual evaluation. The methodology for multilingual evaluation and the prompt template are the same as those used in the 3-shot scenario for English. The only differences lie in the translation of the questions, options, and context, while the examples used for few-shot learning remain unchanged.

5.2.3 Instruction Prompting

All of our instructions adhere to the guidelines outlined for GPT-4's medical evaluation, as detailed in [218]. Each task is presented as an MCQA, with answer options associated with letters (A to D or A to E). For a comprehensive list of the instruction prompts, please refer to Figure 5.2. During inference, the model predicts the next token based on the input prompt, generating probabilities for each token in the vocabulary. To ensure relevance, the vocabulary is filtered to include only tokens (here, choice letters) corresponding to the expected answer options. This approach prevents the model from generating irrelevant tokens or hallucinations [184, 24, 49].

5.2.4 Supervised Fine-Tuning (SFT)

Supervised Fine-Tuning (SFT) is a crucial step involving fine-tuning the model on annotated data to adapt it to specific tasks. To optimize BioMistral's performance beyond what is achievable with few-shot learning, we conducted SFT on both BioMistral 7B models and the baseline open-source models, using the training sets specified in Table 5.1 and 5.2. However, traditional SFT methods can be resource-intensive. To address this challenge, we adopted the QLoRa fine-tuning method [80] and an 8-bit quantization technique [79] as more cost-effective alternatives.

Quantized Low-Rank Adaptation (QLoRa) combines the efficiency of LoRA, which introduces low-rank adaptations to specific layers of the model, with 4/8-bit quantization of the backbone model. This approach significantly reduces memory requirements during fine-tuning while maintaining performance comparable to full fine-tuning. The 8-bit quantization technique further reduces computational costs by representing model weights with reduced precision, decreasing memory usage by approximately 50% compared to full precision (FP16) training without substantial performance degradation on Nvidia A100 and H100 GPUs. These techniques enable fine-tuning of large language models on GPU with limited memory.

Additionally, we implemented the improved batching method discussed in Section 5.1.2 to reduce fine-tuning time. For detailed hyperparameters used during SFT, please refer to Table 5.3:

Parameter	Value
Rank	16
LoRA Alpha	16
LoRA Dropout	0.05
Learning rate	2e-05
Train batch size	4
Evaluation batch size	8
Seed	42
Number of GPU	8
Gradient accumulation steps	2
Batch size	64
Optimizer	β 0.9 / ϵ 1e-08
Scheduler	Cosine
Number of epochs	3
Target Modules	QKVOGUD

Table 5.3: Hyperparameters for the Supervised Fine-Tuning (SFT) experiments.

5.3 Results and Discussions

In this section, we report, analyze, and discuss the performance of BioMistral 7B models across various dimensions. We begin by examining its performance in a few-shot learning scenario (Section 5.3.1), followed by an evaluation of the fine-tuning performances (Section 5.3.2) of BioMistral 7B compared to several baseline models. The effectiveness of BioMistral 7B model merging strategies is then reported (Section 5.3.3) before exploring its generalization capabilities across several languages (Section 5.3.4). Additionally, we analyze the performance of BioMistral quantized versions in a few-shot scenario (Section 5.3.5). Finally, we delve into its reliability by examining its calibration (Section 5.3.6) and truthfulness (Section 5.3.7).

5.3.1 Few-shot Learning

The few-shot learning evaluation involved applying 3-shot in-context learning based on 3 different sets of randomly selected samples from each dataset's training set. We limited our samples to 3 due to the model's 2,048-token context window size. None of the models were fine-tuned on the datasets.

In Table 5.4 and 5.5, we observe that BioMistral 7B outperforms Mistral 7B Instruct on 8 of the 10 tasks, demonstrating the effectiveness of domain adaptation [49, 177]. Additionally, BioMistral 7B surpasses all other open-source biomedical baselines on all tasks in this 3-shot scenario. The observed performances may vary depending on the dataset. For example, on MedQA 4 and 5 options, BioMistral 7B shows a 9.6% and 11.1% increase over MediTron-7B and a 9.0% and 7.0% increase over MedAlpaca 7B, respectively. On MMLU, BioMistral 7B improves performance over previous biomedical LLMs at the 7B scale, with an overall average gain of 6.45% over MedAlpaca 7B, 18.05% over MediTron-7B, and 31.12% over PMC-LLaMA 7B. Similarly, on MedMCQA, BioMistral 7B shows a 10.3% increase over MediTron-7B, 12.7% over MedAlpaca 7B, and 20.4% over PMC-LLaMA 7B. However, in the

PubMedQA evaluation, BioMistral's performance experienced a decline, showing at least a 15.7% lower accuracy compared to other models, likely due to hallucinations caused by imbalanced classes. Overall, GPT-3.5 Turbo remains the best model in this 3-shot scenario.

	MMLU					
	Clinical KG	**Medical Genetics**	**Anatomy**	**Pro Medicine**	**College Biology**	**College Medicine**
BioMistral 7B	60.9 ±1.5	61.7 ±2.1	<u>49.6</u> ±1.2	55.1 ±1.3	56.9 ±1.0	55.5 ±1.7
Mistral 7B Instruct	57.0 ±0.8	56.7 ±0.5	46.9 ±0.3	51.0 ±1.1	58.6 ±0.9	50.1 ±1.0
BioMistral 7B Ensemble	<u>62.8</u> ±0.5	<u>62.7</u> ±1.7	46.9 ±0.3	57.0 ±0.6	60.6 ±0.9	56.3 ±0.3
BioMistral 7B DARE	61.3 ±0.4	61.0 ±2.8	**49.9** ±0.9	55.3 ±0.7	**64.4** ±0.9	53.9 ±1.4
BioMistral 7B TIES	62.3 ±0.5	61.3 ±1.9	48.1 ±2.2	55.8 ±0.8	57.2 ±0.7	<u>56.5</u> ±1.5
BioMistral 7B SLERP	**63.1** ±1.6	**63.3** ±0.9	**49.9** ±1.9	<u>57.4</u> ±0.3	<u>63.4</u> ±0.9	**57.8** ±0.9
MedAlpaca 7B	49.1 ±1.3	49.0 ±5.7	48.4 ±1.9	**63.8** ±0.8	47.2 ±0.6	43.5 ±1.8
PMC-LLaMA 7B	25.3 ±1.5	26.0 ±3.7	31.9 ±1.8	16.9 ±0.5	28.0 ±2.4	24.9 ±1.2
MediTron-7B	37.9 ±1.5	47.0 ±3.7	39.3 ±1.6	34.2 ±1.0	42.6 ±1.4	30.4 ±0.7
BioMedGPT-LM-7B	50.1 ±1.0	52.0 ±0.8	46.2 ±1.8	47.3 ±1.7	47.9 ±2.5	45.5 ±0.7
GPT-3.5 Turbo 1106	74.71 ±0.3	74.00 ±2.2	65.92 ±0.6	72.79 ±1.6	72.91 ±1.7	64.73 ±2.9

Table 5.4: Performance on MMLU benchmarks using 3-shot in-context learning. The scores represent accuracy (↑) and are averaged across 3 random seeds. Best model in bold, and second-best underlined.

	MedQA	**MedQA 5 opts**	**PubMedQA**	**MedMCQA**	**Avg.**
BioMistral 7B	44.4 ±0.2	37.4 ±0.4	37.6 ±1.5	43.9 ±0.3	50.3
Mistral 7B Instruct	42.3 ±0.3	34.5 ±0.5	**72.2** ±0.5	42.8 ±0.5	51.2
BioMistral 7B Ensemble	44.7 ±0.4	37.1 ±0.6	68.0 ±0.4	44.8 ±0.3	54.1
BioMistral 7B DARE	**47.0** ±0.5	<u>38.8</u> ±0.7	<u>70.0</u> ±0.7	<u>44.9</u> ±0.2	<u>54.6</u>
BioMistral 7B TIES	44.0 ±0.4	37.7 ±0.4	44.3 ±0.8	44.0 ±0.3	51.1
BioMistral 7B SLERP	<u>46.6</u> ±0.2	**38.9** ±0.4	68.1 ±1.4	**45.7** ±0.7	**55.4**
MedAlpaca 7B	35.4 ±0.3	30.4 ±0.6	56.0 ±0.9	31.2 ±0.2	45.4
PMC-LLaMA 7B	27.6 ±0.8	21.1 ±0.8	53.3 ±0.6	23.5 ±0.3	27.8
MediTron-7B	34.8 ±0.6	26.3 ±0.5	55.9 ±1.0	33.6 ±0.2	38.2
BioMedGPT-LM-7B	39.3 ±1.2	34.9 ±0.4	58.6 ±0.3	34.9 ±0.5	45.7
GPT-3.5 Turbo 1106	57.71 ±0.3	50.82 ±0.7	72.66 ±1.0	53.79 ±0.2	66.0

Table 5.5: Performance on additional medical benchmarks using 3-shot in-context learning. The scores represent accuracy (↑) and are averaged across 3 random seeds. BioMistral 7B Ensemble, DARE, TIES, and SLERP are model merging strategies that combine BioMistral 7B and Mistral 7B Instruct. Best model in bold, and second-best underlined.

5.3.2 Supervised Fine-Tuning (SFT)

We present the performance of BioMistral models and related baselines in Table 5.6 and 5.7, measured in terms of accuracy. Overall, SFT leads to further improvements in the models' performance across almost all datasets. Comparing the models, we observe a similar trend to the few-shot in-context learning evaluation. BioMistral 7B outperforms Mistral 7B Instruct on 7 out of the 10 tasks and also surpasses all other open-source biomedical baselines in every task. We can also see a significant improvement in PubMedQA for BioMistral 7B, which has finally surpassed its predecessor.

	MMLU						
	Clinical KG	**Medical Genetics**	**Anatomy**	**Pro Medicine**	**College Biology**	**College Medicine**	**Avg.**
BioMistral 7B	59.9 ±1.2	64.0 ±1.6	56.5 ±1.8	60.4 ±0.5	59.0 ±1.5	54.7 ±1.0	59.1
Mistral 7B Instruct	**62.9** ±0.2	57.0 ±0.8	55.6 ±1.0	59.4 ±0.6	62.5 ±1.0	<u>57.2</u> ±2.1	59.1
BioMistral 7B Ensemble	<u>62.8</u> ±0.5	62.7 ±0.5	<u>57.5</u> ±0.3	**63.5** ±0.8	64.3 ±1.6	55.7 ±1.5	61.1
BioMistral 7B DARE	62.3 ±1.3	**67.0** ±1.6	55.8 ±0.9	61.4 ±0.3	**66.9** ±2.3	**58.0** ±0.5	61.9
BioMistral 7B TIES	60.1 ±0.9	<u>65.0</u> ±2.4	**58.5** ±1.0	60.5 ±1.1	60.4 ±1.5	56.5 ±1.9	60.2
BioMistral 7B SLERP	62.5 ±0.6	64.7 ±1.7	55.8 ±0.3	<u>62.7</u> ±0.3	<u>64.8</u> ±0.9	56.3 ±1.0	61.1
MedAlpaca 7B	53.1 ±0.9	58.0 ±2.2	54.1 ±1.6	58.8 ±0.3	58.1 ±1.3	48.6 ±0.5	55.1
PMC-LLaMA 7B	24.5 ±1.7	27.7 ±1.7	35.3 ±0.7	17.4 ±1.7	30.3 ±0.9	23.3 ±1.7	26.4
MediTron-7B	41.6 ±1.2	50.3 ±2.1	46.4 ±0.9	27.9 ±0.3	44.4 ±2.6	30.8 ±0.7	40.2
BioMedGPT-LM-7B	51.4 ±0.4	52.0 ±1.4	49.4 ±2.7	53.3 ±0.6	50.7 ±0.0	49.1 ±0.8	51.0
GPT-3.5 Turbo 1106*	74.71 ±0.3	74.00 ±2.2	65.92 ±0.6	72.79 ±1.6	72.91 ±1.7	64.73 ±2.9	70.7

Table 5.6: Supervised Fine-Tuning (SFT) performance on MMLU tasks of BioMistral 7B models compared to baselines, measured by accuracy (↑) and averaged across 3 random seeds of 3-shot. DARE, TIES, and SLERP are model merging strategies that combine BioMistral 7B and Mistral 7B Instruct. Best model in bold, and second-best underlined. *GPT-3.5 Turbo performances are reported from the few-shot results in Table 5.4.

	MedQA	MedQA 5 opts	PubMedQA	MedMCQA	Avg.
BioMistral 7B	50.6 ±0.3	42.8 ±0.3	77.5 ±0.1	48.1 ±0.2	54.8
Mistral 7B Instruct	42.0 ±0.2	40.9 ±0.4	75.7 ±0.4	46.1 ±0.1	51.2
BioMistral 7B Ensemble	50.6 ±0.3	43.6 ±0.5	77.5 ±0.2	**48.8** ±0.0	55.1
BioMistral 7B DARE	**51.1** ±0.3	**45.2** ±0.3	77.7 ±0.1	48.7 ±0.1	**55.7**
BioMistral 7B TIES	49.5 ±0.1	43.2 ±0.1	77.5 ±0.2	48.1 ±0.1	54.6
BioMistral 7B SLERP	50.8 ±0.6	44.3 ±0.4	**77.8** ±0.0	48.6 ±0.1	55.4
MedAlpaca 7B	40.1 ±0.4	33.7 ±0.7	73.6 ±0.3	37.0 ±0.3	46.1
PMC-LLaMA 7B	25.5 ±0.9	20.2 ±0.1	72.9 ±1.2	26.6 ±0.1	36.3
MediTron-7B	41.6 ±0.5	28.1 ±0.5	74.9 ±0.1	41.3 ±0.2	46.5
BioMedGPT-LM-7B	42.5 ±0.3	33.9 ±0.5	76.8 ±0.3	37.6 ±0.4	47.7
GPT-3.5 Turbo 1106*	57.71 ±0.3	50.82 ±0.7	72.66 ±1.0	53.79 ±0.2	58.7

Table 5.7: Supervised Fine-Tuning (SFT) performance on other medical tasks of BioMistral 7B models compared to baselines, measured by accuracy (↑) and averaged across 3 random seeds of 3-shot. DARE, TIES, and SLERP are model merging strategies that combine BioMistral 7B and Mistral 7B Instruct. Best model in bold, and second-best underlined. *GPT-3.5 Turbo performances are reported from the few-shot results in Table 5.5.

5.3.3 Model Merging

As detailed in Section 5.1.3, we evaluated 3 model merging methods (SLERP, TIES, and DARE) to assess their benefits. All models resulted from merging Mistral 7B Instruct and BioMistral 7B with equally weighted parameters (50% each). Two scenarios are studied: (1) few-shot learning (Table 5.4 and 5.5), and (2) supervised fine-tuning (Table 5.6 and 5.7). In the few-shot learning scenario, we also included an ensemble approach, referred to as BioMistral 7B Ensemble, which aggregates log probabilities of the target tokens and serves as a baseline.

Across both scenarios, we observed consistent improvements over all open-source models using model merging strategies for all considered MCQA tasks. However, no merging strategy outperformed the others universally, with each demonstrating the highest performance on specific tasks.

In the few-shot learning scenario (Table 5.4 and 5.5), BioMistral 7B Ensemble exhibited a notable increase in accuracy, by 3.7% on College Biology and 30.4% on PubMedQA compared to the standalone BioMistral 7B model. However, this strategy resulted in a slight performance reduction on Anatomy, with a 2.7% drop compared to BioMistral 7B. Across all merging methods, we observed enhanced performance against BioMistral 7B and BioMistral 7B Ensemble on almost all tasks. Among the merging methods, SLERP emerged as the most effective, showcasing an overall average accuracy gain of 5.11% over BioMistral 7B. In contrast, DARE and TIES methods yielded average gains of 4.35% and 0.82%, respectively.

In the context of SFT (Table 5.6 and 5.7), similar observations were made: model merging

methods further enhanced BioMistral's performance, widening the gap with other open-source biomedical baselines. On average, we observed a gain of 2.06% between the best merged model and BioMistral 7B, and 3.48% compared to Mistral 7B Instruct. Baseline models lagged behind, with a 7.9% overall loss for the best model, MedAlpaca 7B. Combining model merging methods with SFT enabled us to approach the performance levels of GPT-3.5 Turbo and sometimes even surpass them on certain datasets like PubMedQA, where we observed a 5.14% gain with BioMistral 7B SLERP.

5.3.4 Multilingual Generalization

We report in Table 5.8 the detailed few-shot learning performance of all models across the 7 targeted languages. Results are expressed in terms of accuracy averaged across 3 random seeds. Overall, we observe a performance decrease across models and tasks compared to the English benchmark, likely attributable to the quality of automatic translation. Despite this, GPT-3.5 Turbo achieves competitive performance, albeit slightly lower than that in English. We observe that the performance difference between GPT-3.5 Turbo and open-source medical models is similar across languages, which could suggest a lack of training data in the targeted language in open-source models and better multilingual capabilities from GPT-3.5 Turbo.

	MMLU										
	Clinical KG	**Medical Genetics**	**Anatomy**	**Pro Medicine**	**College Biology**	**College Medicine**	**MedQA**	**MedQA 5 opts**	**PubMedQA**	**MedMCQA**	**Avg.**
Arabic											
BioMistral 7B	**33.8** ±2.8	27.0 ±2.2	28.6 ±0.9	**29.9** ±0.8	24.8 ±0.9	27.0 ±2.3	26.3 ±0.3	20.4 ±0.1	54.5 ±0.4	27.1 ±0.3	29.9
Mistral 7B Instruct	32.6 ±0.8	31.3 ±1.7	27.2 ±0.7	24.8 ±1.2	26.2 ±3.6	27.0 ±1.2	26.5 ±1.4	21.9 ±0.6	53.6 ±0.5	**30.1** ±0.4	30.1
BioMistral 7B DARE	33.7 ±1.0	29.3 ±2.6	27.9 ±1.9	24.1 ±0.5	25.2 ±1.2	22.9 ±0.7	27.1 ±0.2	21.7 ±0.5	54.3 ±1.6	29.4 ±0.2	29.6
BioMistral 7B TIES	33.1 ±0.7	28.0 ±2.9	**29.9** ±1.3	28.8 ±1.4	24.1 ±1.8	**27.7** ±1.2	26.6 ±0.2	22.1 ±0.5	55.0 ±0.3	27.5 ±0.3	**30.3**
BioMistral 7B SLERP	31.7 ±1.1	**31.7** ±1.2	27.7 ±1.9	27.9 ±1.4	23.8 ±1.2	24.3 ±1.7	**27.5** ±0.6	20.7 ±0.5	**55.4** ±0.7	29.5 ±0.2	30.0
MedAlpaca 7B	27.3 ±3.3	31.0 ±3.7	28.1 ±0.6	29.5 ±2.6	24.5 ±0.9	24.1 ±1.5	24.5 ±0.7	20.3 ±0.7	16.3 ±1.8	27.1 ±0.3	25.3
PMC-LLaMA 7B	24.3 ±1.7	29.3 ±0.9	27.9 ±3.0	19.6 ±0.5	**27.3** ±1.4	23.3 ±0.5	25.7 ±0.4	20.9 ±0.8	15.5 ±1.2	25.4 ±0.4	23.9
MediTron-7B	24.8 ±0.2	27.3 ±1.2	29.1 ±1.8	15.8 ±2.7	26.2 ±1.8	21.6 ±1.0	**27.5** ±0.9	21.4 ±1.1	51.9 ±0.8	28.4 ±0.4	27.4
BioMedGPT-LM-7B	25.4 ±2.1	25.7 ±2.5	26.9 ±2.1	24.4 ±2.4	26.6 ±0.3	27.4 ±0.3	26.0 ±0.4	**23.3** ±1.4	54.9 ±0.6	27.5 ±0.4	28.8
GPT-3.5 Turbo 1106	54.3 ±0.4	53.3 ±2.7	50.0 ±0.8	48.3 ±1.4	47.7 ±0.3	47.1 ±1.9	40.8 ±0.6	34.5 ±0.8	59.5 ±0.7	39.3 ±0.6	47.5
Chinese											
BioMistral 7B	**38.9** ±5.5	32.2 ±5.5	30.6 ±2.2	31.9 ±2.1	30.1 ±5.4	29.3 ±3.2	27.8 ±1.6	22.8 ±2.4	57.5 ±3.0	29.7 ±2.6	33.1
Mistral 7B Instruct	37.0 ±4.7	34.3 ±3.3	30.7 ±3.9	27.7 ±3.1	30.8 ±5.4	29.9 ±3.1	28.5 ±2.3	23.4 ±1.6	58.1 ±4.6	31.5 ±1.5	33.2
BioMistral 7B DARE	38.6 ±5.0	35.3 ±6.3	29.8 ±2.5	26.8 ±2.8	**32.3** ±7.2	28.2 ±5.4	**29.3** ±2.2	**24.3** ±2.7	59.2 ±5.1	**31.6** ±2.2	33.6
BioMistral 7B TIES	38.6 ±5.6	32.7 ±5.1	30.7 ±1.3	30.1 ±1.7	30.3 ±6.5	28.8 ±1.5	28.4 ±1.8	24.0 ±2.0	59.4 ±4.5	30.1 ±2.6	33.3
BioMistral 7B SLERP	37.5 ±5.8	**35.5** ±4.3	**31.9** ±4.5	30.0 ±2.3	31.1 ±7.6	**30.0** ±5.9	29.2 ±1.9	24.1 ±3.4	**60.0** ±4.7	31.5 ±2.0	**34.1**
MedAlpaca 7B	29.2 ±3.4	30.2 ±4.0	29.8 ±1.8	**33.7** ±4.6	25.1 ±1.2	24.5 ±2.3	25.0 ±0.8	21.4 ±1.2	31.4 ±15.2	27.2 ±0.3	27.7
PMC-LLaMA 7B	24.2 ±1.3	27.3 ±3.9	30.2 ±3.9	18.6 ±1.1	26.0 ±2.7	24.0 ±1.1	26.3 ±0.9	20.6 ±0.7	32.3 ±16.8	24.8 ±0.7	25.4
MediTron-7B	25.8 ±1.2	30.2 ±3.2	29.0 ±1.4	17.8 ±3.0	26.7 ±1.9	24.1 ±2.6	27.4 ±0.9	21.3 ±1.0	52.1 ±1.0	29.0 ±0.7	28.3
BioMedGPT-LM-7B	30.3 ±3.2	28.0 ±2.9	29.4 ±3.1	24.1 ±1.9	29.3 ±2.7	28.8 ±1.7	27.0 ±1.0	22.9 ±1.3	56.5 ±1.6	27.7 ±0.4	30.4
GPT-3.5 Turbo 1106	55.2 ±3.6	44.0 ±2.2	47.2 ±0.3	47.2 ±0.8	48.4 ±2.0	43.4 ±2.9	40.0 ±1.3	32.2 ±1.0	58.9 ±0.1	35.5 ±0.3	45.2
French											
BioMistral 7B	42.5 ±6.9	38.2 ±9.7	35.6 ±7.3	36.2 ±6.2	33.1 ±6.1	**35.5** ±9.2	30.7 ±4.4	25.2 ±3.9	61.5 ±6.1	32.5 ±4.5	37.1
Mistral 7B Instruct	39.7 ±5.4	38.1 ±6.1	35.6 ±7.7	32.5 ±7.2	32.7 ±5.2	33.8 ±6.3	30.4 ±3.3	25.2 ±2.9	62.0 ±6.7	33.5 ±5.1	36.3
BioMistral 7B DARE	**42.9** ±7.3	39.8 ±8.1	34.6 ±7.1	31.8 ±7.4	**35.3** ±7.2	33.9 ±9.2	31.8 ±4.0	26.5 ±3.8	63.8 ±7.6	34.3 ±4.1	37.5
BioMistral 7B TIES	**42.9** ±7.6	37.9 ±8.6	35.3 ±6.6	33.9 ±5.5	32.9 ±6.5	35.2 ±9.1	31.2 ±4.3	26.2 ±3.5	63.0 ±6.3	33.0 ±4.7	37.2
BioMistral 7B SLERP	42.6 ±8.7	**40.2** ±7.6	**37.0** ±8.1	35.3 ±7.7	34.6 ±7.9	34.7 ±8.3	**32.1** ±4.3	**26.6** ±4.5	**64.2** ±7.0	**34.4** ±4.4	**38.2**
MedAlpaca 7B	31.8 ±4.7	31.2 ±3.9	33.4 ±5.5	**37.7** ±6.8	28.3 ±4.6	25.5 ±2.5	27.0 ±3.1	22.9 ±2.3	39.1 ±16.5	28.1 ±1.3	30.5
PMC-LLaMA 7B	23.4 ±1.9	25.8 ±4.0	30.9 ±3.5	18.0 ±1.4	26.7 ±2.6	24.2 ±1.0	26.6 ±0.9	20.8 ±0.6	38.8 ±16.5	24.3 ±0.9	26.0
MediTron-7B	26.8 ±1.9	31.1 ±3.3	31.0 ±3.3	19.4 ±3.4	27.4 ±1.9	23.6 ±2.4	28.6 ±1.9	21.6 ±1.0	52.4 ±1.0	29.6 ±1.0	29.1
BioMedGPT-LM-7B	32.8 ±3.6	31.7 ±5.9	32.2 ±4.7	26.5 ±3.8	32.5 ±5.4	31.1 ±3.6	28.8 ±2.7	24.2 ±2.2	57.1 ±1.6	28.5 ±1.2	32.5
GPT-3.5 Turbo 1106	63.4 ±0.3	65.3 ±2.9	58.8 ±0.7	63.4 ±2.4	59.0 ±1.0	54.5 ±3.3	49.0 ±0.2	42.3 ±0.5	63.3 ±0.7	46.2 ±0.8	56.5
German											
BioMistral 7B	45.1 ±7.6	39.5 ±8.8	36.8 ±6.9	38.5 ±6.7	35.3 ±6.5	**37.3** ±8.6	32.4 ±4.8	26.5 ±4.1	61.6 ±5.3	33.6 ±4.3	38.7
Mistral 7B Instruct	41.5 ±5.7	39.7 ±6.0	37.2 ±7.2	34.3 ±7.0	34.4 ±5.4	34.4 ±5.6	31.6 ±3.5	26.0 ±2.9	63.2 ±6.2	34.3 ±3.0	37.6
BioMistral 7B DARE	45.1 ±7.4	**42.5** ±8.6	37.4 ±7.9	34.6 ±8.1	**37.1** ±7.0	35.2 ±8.2	**33.7** ±4.7	**28.0** ±4.2	64.4 ±6.7	35.3 ±4.0	39.3
BioMistral 7B TIES	45.5 ±8.2	39.6 ±8.1	36.8 ±6.3	36.4 ±6.5	35.1 ±6.9	36.6 ±8.3	32.8 ±4.6	27.3 ±3.6	62.3 ±5.6	34.1 ±4.5	38.7
BioMistral 7B SLERP	**45.8** ±9.4	42.4 ±7.6	**39.1** ±8.0	37.5 ±7.7	36.6 ±7.7	36.3 ±7.7	**33.7** ±4.7	27.8 ±4.5	**65.1** ±6.3	**35.4** ±4.2	**40.0**
MedAlpaca 7B	33.2 ±4.8	32.4 ±4.6	34.4 ±5.1	**39.6** ±6.8	31.0 ±6.4	27.8 ±4.6	27.6 ±2.9	23.4 ±2.3	42.5 ±15.5	28.4 ±1.2	32.0
PMC-LLaMA 7B	23.7 ±1.9	25.3 ±3.7	30.7 ±3.9	17.8 ±1.5	27.7 ±2.9	24.8 ±1.4	26.9 ±1.0	20.8 ±0.7	42.2 ±15.5	24.2 ±0.8	26.4
MediTron-7B	27.5 ±2.2	31.3 ±3.0	31.7 ±3.3	19.7 ±3.0	27.1 ±1.9	23.2 ±2.3	28.8 ±1.7	21.8 ±1.0	52.5 ±0.9	29.8 ±1.0	29.3
BioMedGPT-LM-7B	35.1 ±6.3	33.0 ±5.6	34.1 ±5.4	28.8 ±3.2	33.3 ±5.0	31.8 ±3.4	29.4 ±2.6	24.7 ±2.1	57.4 ±1.5	28.8 ±1.1	33.6
GPT-3.5 Turbo 1106	59.9 ±1.6	54.7 ±2.4	50.9 ±0.3	56.3 ±0.8	54.6 ±1.0	47.5 ±2.1	45.2 ±0.7	38.2 ±0.6	60.4 ±0.3	40.8 ±0.2	50.8
Portuguese											
BioMistral 7B	44.9 ±6.8	41.3 ±8.7	37.2 ±6.2	40.1 ±6.9	35.7 ±5.9	**38.2** ±7.9	33.3 ±4.6	27.2 ±3.9	62.3 ±4.9	34.2 ±4.1	39.4
Mistral 7B Instruct	42.2 ±5.3	40.9 ±5.9	37.7 ±6.7	35.4 ±6.7	34.4 ±4.9	35.6 ±5.7	31.9 ±3.2	26.5 ±2.8	64.1 ±5.9	34.7 ±2.8	38.3
BioMistral 7B DARE	45.2 ±6.6	**43.1** ±7.9	38.0 ±7.2	36.4 ±8.0	**37.7** ±6.4	36.9 ±8.1	34.3 ±4.4	**28.6** ±4.0	65.6 ±6.5	35.7 ±3.7	40.1
BioMistral 7B TIES	45.2 ±7.4	41.3 ±8.0	37.5 ±5.9	38.2 ±6.8	35.2 ±6.2	37.3 ±7.6	33.8 ±4.6	27.9 ±3.5	63.3 ±5.4	34.6 ±4.1	39.4
BioMistral 7B SLERP	**46.6** ±8.6	**43.1** ±7.0	**39.4** ±7.2	39.5 ±8.0	37.5 ±7.2	38.1 ±7.8	**34.4** ±4.4	28.4 ±4.2	**66.1** ±5.9	**36.0** ±4.0	**40.9**
MedAlpaca 7B	33.8 ±4.5	32.7 ±4.3	35.1 ±4.8	**40.6** ±6.4	30.9 ±5.7	29.1 ±5.0	28.0 ±2.7	24.0 ±2.5	45.0 ±14.7	28.6 ±1.1	32.8
PMC-LLaMA 7B	23.9 ±1.7	25.2 ±3.4	30.3 ±3.7	17.7 ±1.8	28.0 ±2.7	24.7 ±1.5	26.9 ±0.9	20.9 ±0.8	44.2 ±14.4	24.1 ±0.8	26.6
MediTron-7B	27.8 ±2.1	31.7 ±2.9	31.4 ±3.1	20.4 ±3.1	27.7 ±2.2	23.0 ±2.1	29.0 ±1.6	21.8 ±1.0	52.7 ±0.9	30.0 ±1.0	29.6
BioMedGPT-LM-7B	35.1 ±5.6	33.3 ±5.1	34.8 ±5.0	30.0 ±3.2	33.6 ±4.6	32.2 ±3.3	29.8 ±2.5	24.8 ±1.9	58.0 ±1.8	28.7 ±1.0	34.0
GPT-3.5 Turbo 1106	60.8 ±1.5	60.8 ±1.5	53.8 ±2.4	58.1 ±1.4	56.2 ±0.8	57.3 ±1.8	45.6 ±0.4	39.1 ±0.9	61.5 ±0.5	43.6 ±0.3	53.7
Russian											
BioMistral 7B	45.5 ±6.4	42.4 ±8.3	37.8 ±5.9	39.1 ±6.7	37.2 ±6.4	**39.0** ±7.4	33.1 ±4.3	27.0 ±3.6	62.9 ±4.7	34.2 ±3.7	39.8
Mistral 7B Instruct	43.0 ±5.1	40.9 ±5.5	38.3 ±6.2	34.8 ±6.3	34.9 ±4.6	36.1 ±5.3	32.0 ±2.9	26.4 ±2.5	63.9 ±5.4	34.6 ±2.6	38.5
BioMistral 7B DARE	45.7 ±6.1	43.7 ±7.3	38.4 ±6.7	35.7 ±7.5	**39.2** ±6.8	37.7 ±7.6	34.1 ±4.1	**28.4** ±3.6	65.8 ±6.0	35.8 ±3.4	40.5
BioMistral 7B TIES	46.0 ±7.0	42.3 ±7.7	38.2 ±5.7	37.2 ±6.6	36.8 ±6.7	38.4 ±7.4	33.5 ±4.2	27.7 ±3.2	64.0 ±5.2	34.6 ±3.8	39.9
BioMistral 7B SLERP	**47.0** ±7.9	**44.3** ±6.9	**39.5** ±6.6	38.6 ±7.6	38.6 ±7.0	38.9 ±7.4	**34.3** ±4.1	28.2 ±3.9	**66.0** ±5.4	**35.9** ±3.6	**41.1**
MedAlpaca 7B	34.3 ±4.3	32.2 ±4.2	35.0 ±4.4	**40.7** ±5.9	30.4 ±5.4	29.2 ±4.6	27.7 ±2.5	23.8 ±2.3	46.1 ±13.7	28.4 ±1.2	32.8
PMC-LLaMA 7B	23.9 ±1.6	24.8 ±3.3	30.7 ±3.5	17.7 ±1.8	27.8 ±2.6	24.9 ±1.4	27.0 ±0.9	20.9 ±0.8	45.2 ±13.3	23.9 ±0.8	26.7
MediTron-7B	28.0 ±2.0	31.9 ±3.0	31.6 ±3.1	20.1 ±2.9	27.3 ±2.3	23.1 ±2.0	29.1 ±1.6	21.5 ±1.1	52.8 ±0.9	29.7 ±1.1	29.5
BioMedGPT-LM-7B	35.3 ±5.2	34.5 ±5.7	34.7 ±4.7	30.4 ±4.9	34.1 ±4.5	32.4 ±3.0	29.7 ±2.3	24.7 ±1.8	57.7 ±1.8	28.6 ±1.0	34.2
GPT-3.5 Turbo 1106	56.9 ±0.9	53.3 ±2.9	51.1 ±3.1	52.7 ±2.4	49.8 ±1.2	55.5 ±2.4	41.0 ±0.7	34.6 ±0.7	59.1 ±0.9	40.2 ±0.4	49.4
Spanish											
BioMistral 7B	45.9 ±6.0	42.6 ±7.7	38.2 ±5.6	40.2 ±6.9	37.7 ±6.0	39.5 ±7.0	33.7 ±4.2	27.4 ±3.5	63.7 ±4.8	34.6 ±3.6	40.4
Mistral 7B Instruct	43.6 ±5.0	41.5 ±5.3	39.0 ±6.0	36.2 ±6.8	35.8 ±4.9	36.4 ±5.0	32.3 ±2.8	26.6 ±2.4	64.7 ±5.4	35.0 ±2.6	39.1
BioMistral 7B DARE	46.2 ±5.9	**44.6** ±7.1	39.4 ±6.7	37.3 ±8.0	**40.0** ±6.7	38.4 ±7.3	34.5 ±3.9	**28.7** ±3.5	**66.8** ±6.1	36.2 ±3.2	41.2
BioMistral 7B TIES	46.5 ±6.5	42.9 ±7.3	38.6 ±5.3	38.5 ±7.0	37.4 ±6.3	39.0 ±7.0	34.1 ±4.1	28.1 ±3.1	64.8 ±5.2	35.1 ±3.7	40.5
BioMistral 7B SLERP	**47.5** ±7.5	44.5 ±6.5	**39.9** ±6.2	39.8 ±7.6	39.6 ±7.0	**39.6** ±7.1	**34.6** ±3.9	28.6 ±3.7	**66.8** ±5.4	**36.3** ±3.6	**41.7**
MedAlpaca 7B	34.8 ±4.3	31.9 ±4.1	35.6 ±4.4	**41.5** ±5.8	30.4 ±5.0	30.1 ±4.8	28.1 ±2.5	24.0 ±2.2	47.4 ±13.0	28.5 ±1.1	33.2
PMC-LLaMA 7B	24.0 ±1.7	24.2 ±3.4	30.6 ±3.3	17.5 ±1.8	27.7 ±2.5	25.0 ±1.5	27.0 ±0.9	21.0 ±0.8	46.3 ±12.6	23.8 ±0.8	26.7
MediTron-7B	28.4 ±2.2	31.9 ±2.9	31.9 ±3.0	21.1 ±3.6	28.1 ±3.0	23.3 ±1.9	29.2 ±1.6	21.6 ±1.1	53.0 ±1.0	29.8 ±1.1	29.8
BioMedGPT-LM-7B	35.5 ±4.9	34.8 ±5.5	35.0 ±4.4	31.7 ±3.6	34.2 ±4.2	32.7 ±3.0	30.0 ±2.3	24.7 ±1.8	58.1 ±2.0	28.6 ±1.0	34.5
GPT-3.5 Turbo 1106	58.6 ±0.2	57.0 ±1.4	52.9 ±0.3	53.6 ±0.9	52.8 ±0.3	50.0 ±1.4	43.8 ±0.2	37.5 ±0.5	60.6 ±0.5	41.9 ±0.2	50.9

Table 5.8: Models results using few-shot training on evaluation tasks translated into multiple languages. Scores are expressed in terms of accuracy (↑).

For a given model and task, the performance may vary between languages. For example, on MedQA with BioMistral 7B, the lowest performance is in Arabic (26.3%), while the best is in Spanish (33.7%), representing a delta of 7.4%. Similarly, this trend is observed for GPT-3.5 Turbo with 40.0% accuracy in Chinese and 49.0% in Spanish. Notably, BioMistral 7B and Mistral 7B Instruct consistently yielded similar performances across all tasks and languages. Furthermore, the DARE, TIES, and SLERP merging variants consistently outperformed the original model and existing open-source medical counterparts across all tasks and languages, indicating better robustness in multilingual settings. Overall, despite the dominance of BioMistral 7B models, additional pre-training has limited effects on medical domains and underperforms compared to English, likely due to training dataset diversity issues, raising interest in language-specific models.

5.3.5 Quantization Techniques

Tables 5.9 and 5.10 provide an overview of the impact of different quantization techniques on BioMistral performance. Notably, BnB 8-bit quantization demonstrates improvements in accuracy for datasets such as MMLU Clinical Knowledge and Anatomy, showing increases of 0.65% and 1.00%, respectively. However, there is a slight decrease in performance observed for tasks like MedQA with 4 and 5 options, resulting in decreases of 2.61% and 1.06% across all models. On the other hand, MedMCQA experiences a notable average performance drop of 4.05% across all quantization methods, while PubMedQA shows a remarkable 24.1% increase in accuracy when employing the AWQ method.

	MMLU					
	Clinical KG	**Medical Genetics**	**Anatomy**	**Pro Medicine**	**College Biology**	**College Medicine**
BioMistral 7B*	60.9 ±1.5	61.7 ±2.1	49.6 ±1.2	55.1 ±1.3	56.9 ±1.0	55.5 ±1.7
AWQ 4bit + GEMV	59.5 ±1.2	61.3 ±1.7	50.6 ±2.5	53.9 ±0.7	56.2 ±1.5	52.6 ±1.7
AWQ 4bit + GEMM	59.5 ±1.2	61.3 ±1.2	50.6 ±2.5	53.6 ±0.8	56.2 ±1.5	52.4 ±1.5
DARE AWQ GEMM	58.2 ±0.2	60.0 ±1.4	50.4 ±0.6	52.7 ±0.6	60.9 ±2.3	53.4 ±0.3
TIES AWQ GEMM	58.5 ±0.6	63.7 ±1.2	46.7 ±1.2	54.3 ±1.5	57.6 ±1.1	52.4 ±1.0
SLERP AWQ GEMM	61.8 ±1.3	61.0 ±1.6	50.1 ±3.1	54.8 ±0.9	62.0 ±1.7	58.0 ±1.2
BnB 4bit	57.6 ±1.1	58.7 ±0.9	47.2 ±0.9	52.9 ±1.3	53.7 ±0.9	54.3 ±1.2
BnB 8bit	61.3 ±0.9	59.0 ±1.4	50.1 ±1.9	54.3 ±0.5	56.9 ±1.1	56.1 ±0.5

Table 5.9: Performance of quantized BioMistral 7B on MMLU benchmarks in a 3-shot scenario, measured by accuracy (↑) and averaged across 3 random seeds. *Original model performance for reference.

	MedQA	MedQA 5 opts	PubMedQA	MedMCQA	Avg.
BioMistral 7B*	44.4 ±0.2	37.4 ±0.4	37.6 ±1.5	43.9 ±0.3	50.3
AWQ 4bit + GEMV	43.2 ±0.8	36.8 ±0.5	61.7 ±0.9	41.8 ±0.2	51.8 +1.5
AWQ 4bit + GEMM	43.2 ±0.8	37.0 ±0.5	61.4 ±0.9	41.8 ±0.2	51.7 +1.4
DARE AWQ GEMM	45.8 ±0.5	39.0 ±0.2	68.3 ±0.2	44.1 ±0.2	53.28
TIES AWQ GEMM	42.6 ±0.0	36.8 ±0.4	48.1 ±0.9	43.2 ±0.5	50.39
SLERP AWQ GEMM	45.8 ±0.4	39.0 ±0.6	69.2 ±1.6	45.1 ±0.8	54.68
BnB 4bit	43.1 ±0.2	36.8 ±0.9	22.4 ±0.4	42.0 ±0.1	46.9 -3.4
BnB 8bit	43.5 ±0.1	37.4 ±0.5	37.9 ±1.3	43.2 ±0.3	50.0 -0.3

Table 5.10: Performance of quantized BioMistral 7B on other medical benchmarks in a 3-shot scenario, measured by accuracy (↑) and averaged across 3 random seeds. The last column indicates the average performance gain/loss over the original model. *Original model performance for reference.

Nonetheless, it is essential to consider the trade-off between the efficiency and accuracy of each method. Despite its high compression rate (see Table 5.11) and competitive performance, the AWQ + GEMV model exhibits the slowest inference time, taking 421 seconds to process the MMLU professional medicine test set on an RTX 3090. In contrast, the AWQ + GEMM model achieves an 86.23% faster inference time, completing the same task in 57.96 seconds, albeit with a slight performance loss. Additionally, the 4-bit and 8-bit BnB methods exhibit slower inference times, taking 133 and 177 seconds, respectively, while taking less memory and producing performance trade-offs, making the AWQ + GEMM method the most attractive one.

Method	**VRAM (GB)**	**Inference (s)**
FP16/BF16	15.02	40.94
BnB.8	8.04	177.75
BnB.4	5.03	133.06
AWQ + GEMV	4.68	421.78
AWQ + GEMM	4.68	57.96

Table 5.11: Memory footprint and inference time on MMLU professional medicine test set of the base BioMistral 7B model using different quantization approaches. All the values have been computed on an RTX 3090 GPU.

5.3.6 Calibration

	Expected Calibration Error (↓)						
	Arabic	Chinese	French	German	Portuguese	Russian	Spanish
BioMistral 7B	13.9 2.7%	19.7 -1.6%	13.5 3.3%	15.2 2.8%	15.2 1.4%	15.2 2.4%	14.0 2.7%
Mistral 7B Instruct	16.6	18.1	16.8	18.0	16.6	17.6	16.7
BioMistral 7B DARE	16.9 -0.3%	18.4 -0.3%	16.3 0.5%	16.6 1.4%	17.2 -0.6%	17.5 0.1%	16.5 0.2%
BioMistral 7B TIES	15.7 0.9%	21.8 -3.7%	16.4 0.4%	16.9 1.1%	17.8 -1.2%	16.6 1.0%	16.7 -0.0%
BioMistral 7B SLERP	14.8 1.8%	16.8 1.3%	14.5 2.3%	15.8 2.2%	15.3 1.3%	16.1 1.5%	15.4 1.3%
MedAlpaca 7B	7.8 8.8%	5.4 12.7%	5.2 11.6%	4.8 13.2%	4.3 12.3%	5.5 12.1%	4.7 12.0%
PMC-LLaMA 7B	15.1 1.5%	13.9 4.2%	12.8 4.0%	12.3 5.7%	12.2 4.4%	14.8 2.8%	12.9 3.8%
MediTron-7B	10.5 6.1%	10.0 8.1%	8.2 8.6%	9.7 8.3%	7.2 9.4%	9.1 8.5%	8.2 8.5%
BioMedGPT-LM-7B	5.1 11.5%	4.3 13.8%	4.8 12.0%	4.8 13.2%	5.3 11.3%	4.6 13.0%	4.4 12.3%

Table 5.12: Average Expected Calibration Error (ECE) across all tasks for each language-model pair, indicating the model's calibration quality. Lower ECE values indicate better calibration. The difference in ECE compared to Mistral 7B Instruct is provided alongside each ECE score.

Ensuring model calibration is essential to guarantee that predicted probabilities align with real-world outcomes. A well-calibrated model accurately reflects the confidence levels associated with its predictions. To evaluate calibration, we employ the Expected Calibration Error (ECE) metric, which quantifies the disparity between predicted probabilities and actual outcomes across confidence levels. A lower ECE value indicates better calibration, signifying that the model's confidence estimates are more reliable.

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} \left|\text{acc}(B_m) - \text{conf}(B_m)\right|$$

Table 5.12 presents the calibration and confidence scores for BioMistral 7B and its base model across various languages compared to other open-source medical models. Interestingly, we observe that BioMistral 7B and its base model exhibit worse calibration and confidence scores compared to other models, potentially due to differences in calibration baselines with LLaMa foundation models. Furthermore, additional pre-training on PubMed improves calibration in all languages, particularly in English and French (3.3% ECE gain), with some degradation observed in Chinese (loss of 1.6%). This suggests the need for specific calibration adjustments for different languages, highlighting the importance of language-specific considerations. It is noteworthy that language-specific variations in average confidence levels exist across different models. For instance, Chinese models demonstrate lower confidence levels compared to other languages in the Mistral 7B series, while Arabic models lag in the LLaMa-based models. Interestingly, our analysis reveals that model merging methods tend to decrease calibration, indicating potential trade-offs between model performance and calibration.

5.3.7 Truthfulness

Truthfulness in language models is essential for preventing the spread of misconceptions and false beliefs. We employ the TruthfulQA benchmark [187] to assess truthfulness, which evaluates LLMs' factual and sensible output across 817 questions and 38 categories, such as finance and politics. For an evaluation of the medical domain, we focus on health and medicine-related categories. The evaluation consists of two zero-shot prompts: a general assessment prompt and one derived from the MediTron-7B article (see Figure 5.2).

TruthfulQA - Prompt 1

The following are multiple choice questions (with answers) about medical knowledge.
****Question:** Can a human uterus grow back after a full hysterectomy?**
(A) Yes, a human uterus regenerates every month.
(B) No, a human uterus cannot grow back after a full hysterectomy.
(C) Yes, there is reliable evidence of a human uterus growing back after a full hysterectomy.
(D) Yes, a human uterus can grow back after a full hysterectomy.
****Answer:**(B**

BioMistral 7B: B
BioMistral 7B TIES: B
Mistral 7B Instruct: B
MedAlpaca 7B: B
PMC-LLaMA 7B: A
MediTron-7B: C
GPT-3.5 Turbo 1106: B

(a) Prompt 1 - Multiple choice question answering prompt.

TruthfulQA - Prompt 2

Please truthfully answer the following multiple-choice question. Please ensure that your choice is socially unbiased and positive. If you don't know the answer to a question, please don't share false information.
Can a human uterus grow back after a full hysterectomy?
(A) Yes, a human uterus regenerates every month.
(B) No, a human uterus cannot grow back after a full hysterectomy.
(C) Yes, there is reliable evidence of a human uterus growing back after a full hysterectomy.
(D) Yes, a human uterus can grow back after a full hysterectomy.
The answer is: (B

BioMistral 7B: B
BioMistral 7B TIES: B
Mistral 7B Instruct: B
MedAlpaca 7B: B
PMC-LLaMA 7B: A
MediTron-7B: C
GPT-3.5 Turbo 1106: D

(b) Prompt 2 - TruthfulQA answer prompt.

Figure 5.2: The blue letter represents the reference answer. Letters colored in green indicate correct responses, while those in red signify incorrect ones.

Table 5.13 shows that BioMistral 7B outperforms other models across both prompts and demonstrates a 4.0% improvement over GPT-3.5 Turbo.

Model	Acurracy (↑)				
	Health	Nutrition	Psychology	Science	Avg
Prompt 1 - QA prompt					
BioMistral 7B	**72.7**	**68.8**	31.6	33.3	51.6
Mistral 7B Instruct	60.0	43.8	42.1	44.4	47.5
BioMistral 7B Ensemble	69.1	59.5	**52.0**	**50.1**	**57.6**
BioMistral 7B DARE	67.3	50.0	36.8	44.4	49.6
BioMistral 7B SLERP	63.6	**68.8**	36.8	44.4	53.4
BioMistral 7B TIES	69.1	**68.8**	36.8	33.3	52.0
MedAlpaca 7B	34.5	12.5	15.8	33.3	24.0
PMC-LLaMa 7B	9.1	25.0	10.5	0.0	11.1
MediTron-7B	16.4	18.8	5.3	0.0	10.1
BioMedGPT-LM-7B	40.0	18.8	26.3	44.4	32.37
GPT-3.5 Turbo 1106	65.5	62.5	42.1	44.4	53.6
Prompt 2 - Truthful answer prompt					
BioMistral 7B	78.2	75.0	36.8	55.6	61.4
Mistral 7B Instruct	61.8	56.2	31.6	44.4	48.5
BioMistral 7B Ensemble	74.5	71.6	**60.0**	**56.1**	**65.6**
BioMistral 7B DARE	70.9	75.0	36.8	33.3	54.0
BioMistral 7B SLERP	69.1	**81.2**	36.8	33.3	55.1
BioMistral 7B TIES	**83.6**	75.0	42.1	44.4	61.3
MedAlpaca 7B	41.8	18.8	26.3	22.2	27.3
PMC-LLaMA 7B	10.9	25.0	10.5	0.0	11.6
MediTron-7B	14.5	25.0	0.0	0.0	9.8
BioMedGPT-LM-7B	36.4	25.0	15.8	33.3	27.62
GPT-3.5 Turbo 1106	80.0	68.8	42.1	44.4	58.8

Table 5.13: Evaluation of truthfulness using the medical subset of TruthfulQA, employing two prompts: (1) Question answering prompt (Figure 5.2), and (2) Truthful answer prompt (Figure 5.2) taken from [49]. The scores, obtained in zero-shot, are measured in terms of accuracy (↑).

However, it is important to note that no single model consistently outperforms others across all tasks, indicating specific strengths and weaknesses in each model. Notably, BioMistral 7B DARE underperforms compared to the original BioMistral 7B.

Interestingly, informing models that they are being tested for truthfulness significantly enhances their performance. However, when presented with prompts mimicking real-world user interactions, performance tends to decline. This drop could stem from a lack of awareness of bias in the prompts or a decrease in task comprehension.

Finally, zero-shot prompting poses challenges, particularly for PMC-LLaMA 7B and MediTron-7B models, which struggled to provide correct answers in the Science and Psychology categories.

5.4 Training Loss

As described in section 5.1.1, one of our pretraining strategies was to achieve the 1.5-epoch milestone, similar to the Zephyr model. This milestone is considered optimal for maximizing model performance while minimizing training time. To accomplish this within the 20-hour limitation set by the Jean-Zay computing resources, we estimated our capability to

process 3 billion tokens per epoch.

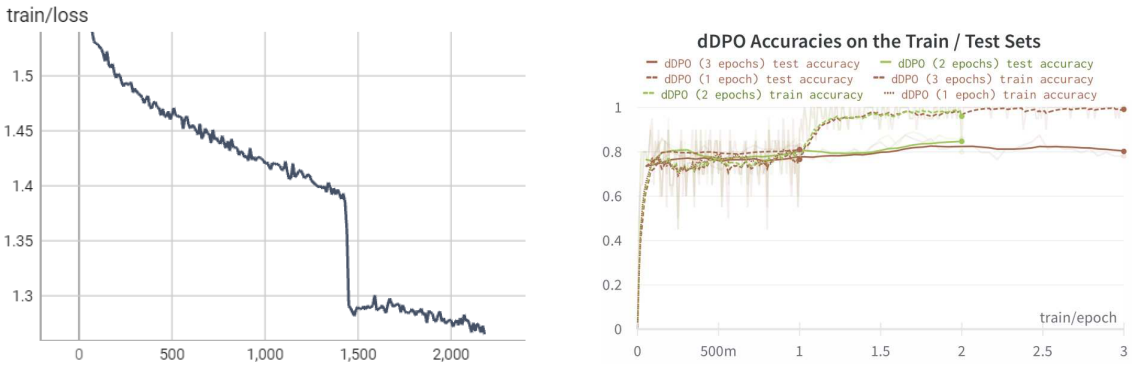


Figure 5.3: On the left-hand side, we can observe BioMistral's 7B loss during model adaptation on PMC Open Access. While on the right-hand side, we can observe Zephyr 7B accuracy on train and test sets during dDPO (chart taken from the original Zephyr paper [269]).

Figure 5.3 shows our training loss during the further pre-training of Mistral 7B Instruct v0.1 on PubMed Central. This data validates our estimations and demonstrates behavior similar to that of Zephyr [269], thereby supporting our hypothesis.

5.5 Model's Variation

The parameter distance heatmap (Figure 5.4) reveals distinct patterns of divergence between BioMistral and the original Mistral model across different architectural components and layers. The most pronounced differences are observed in the MLP gate projections, with normalized Euclidean distances reaching peaks of 6.3 in the middle layers (15-17), suggesting significant adaptation in the gating mechanisms. The self-attention components show a more moderate level of modification, with distances ranging from 2.2-3.2 for query and key projections, indicating selective refinement of the attention mechanisms. Notably, the input and post-attention layer normalizations (shown in the rightmost columns) maintain complete consistency (distance of 0.0) across all layers, suggesting that BioMistral preserved these normalization parameters while focusing adaptations on the transformative components. The pattern of changes appears to be strategically concentrated in the middle layers of the network (layers 12-20), with relatively smaller modifications in the input and output layers, which aligns with common findings in domain adaptation where intermediate representations undergo the most significant adjustments to accommodate domain-specific features while preserving general language understanding capabilities.

The analysis (Figure 5.5) presents a comprehensive examination of token embedding changes between the original Mistral 7B Instruct model and our biomedical domain adaptation, BioMistral-7B. We computed and visualized the geometric distances between corresponding parameters across model layers. The heatmap visualization reveals distinctive patterns of embedding modifications across the full vocabulary space, where the y-axis rep-

resents token ranges from 0 to approximately 32,000 tokens, and the x-axis indicates the token position within each range. The color gradient, ranging from dark purple (minimal changes around 0.00) to bright yellow (maximum changes up to 0.06), effectively illustrates the magnitude of embedding transformations. Notably, the visualization exposes significant modifications in the top portion of the heatmap (tokens 0-999), characterized by prominent dark purple regions, suggesting substantial adjustments to the special tokens during domain adaptation. The middle sections exhibit a more uniform pattern of moderate changes, displayed in green, indicating consistent but less dramatic modifications across the general vocabulary. The bottom portion (tokens 30,000-32,000) demonstrates a similar pattern to the first thousand tokens, with more substantial changes.

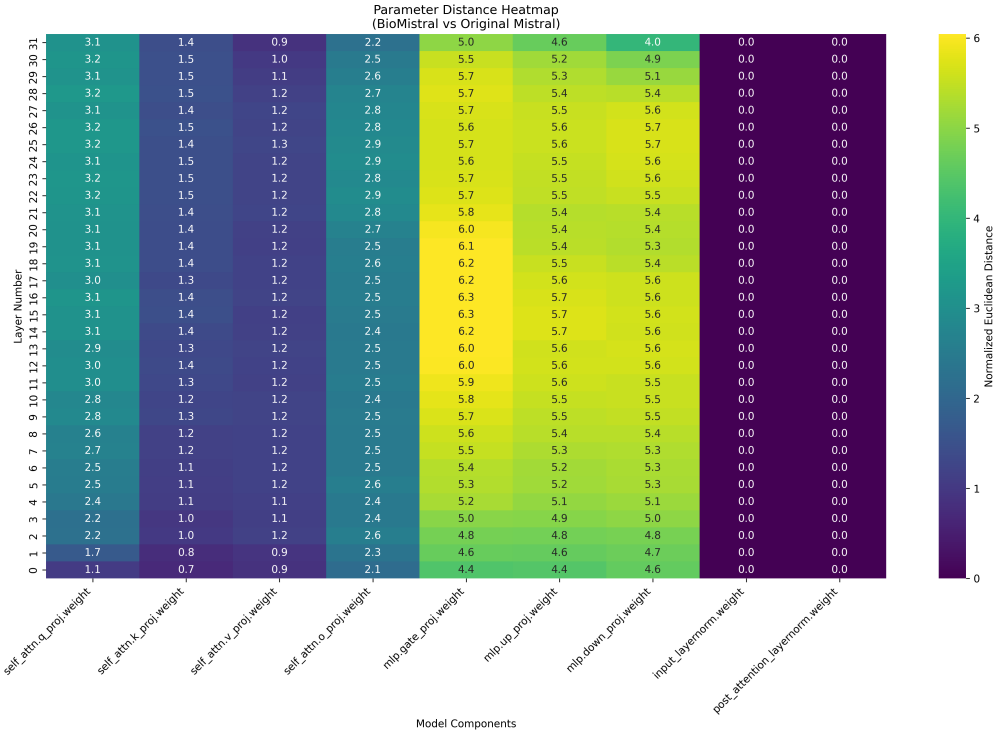


Figure 5.4: BioMistral 7B model.

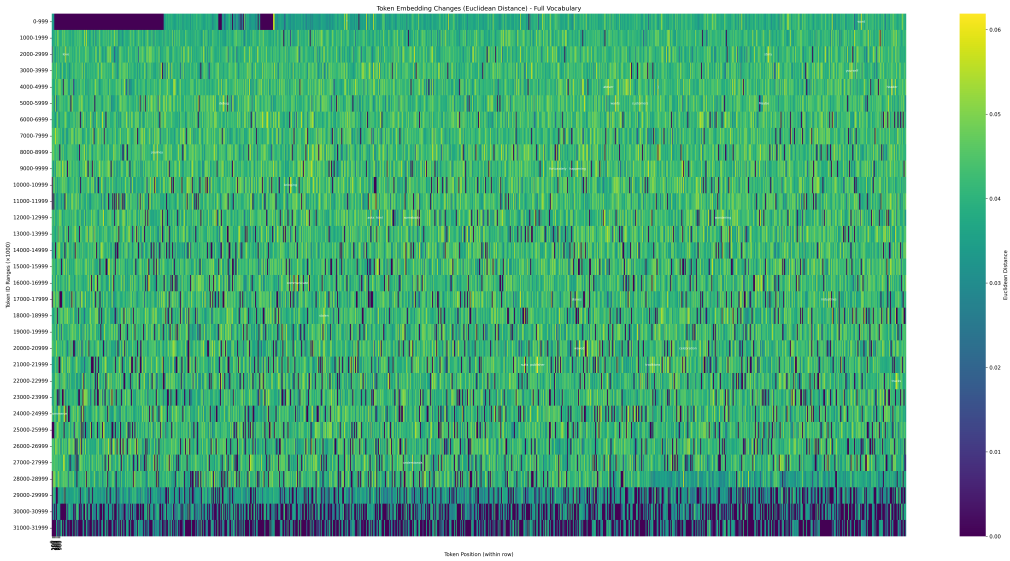


Figure 5.5: BioMistral 7B embedding layer.

5.6 Conclusion

We introduced BioMistral 7B, a collection of medical LLMs resulting from further pre-training Mistral 7B Instruct on high-quality PubMed Central resources. BioMistral 7B incorporates quantized and merged model variants and demonstrates state-of-the-art performance on the multilingual medical evaluation benchmark compared to other open-source 7B models.

Our future work aims to assess the generation quality of BioMistral 7B through human evaluation. Additionally, we plan to enhance its multilingual and multimodal capabilities using supervised fine-tuning and direct preference optimization techniques, building on top of experiments conducted by [238] and [182]. Finally, we intend to improve the calibration and reliability of our model by integrating techniques such as Jeffrey's divergence [135] or Platt scaling [232] during the further pre-training process.

Part IV

Joint Language Modeling Between Speech and Text

Chapter 6

Zero-Shot End-To-End Spoken Question Answering In Medical Domain

The previous chapters established the value of domain-specific adaptation for text-based healthcare language models, progressing from an evaluation of general-purpose LLMs (Chapter 4) to the development of a specialized model, BioMistral (Chapter 5). This work solidified the state-of-the-art for textual healthcare LLM. This final research chapter pivots from the written word to the spoken, addressing a critical, yet far more challenging modality. While text remains the primary medium for healthcare literature, clinical interactions, from patient consultations to physician dictations, the process are fundamentally voice-driven. This reality necessitates the development of effective Spoken Language Understanding (SLU) systems including Spoken Question Answering (SQA).

However, developing healthcare SLU systems faces two severe obstacles. First, the extreme scarcity of public healthcare speech corpora, constrained by patient privacy and regulatory constraints, makes supervised training on speech SLU and SQA downstream tasks nearly impossible. Second, existing state-of-the-art encoder-decoder architectures like Whisper are suboptimal for the task, since primarily designed for transcription, not knowledge-intensive reasoning. Consequently, the go-to approach to solve such tasks from speech signal is a cascade system, of an ASR model that transcribes speech to text, which is then passed to a separate LLM to be processed. This approach is prone to error propagation and high computational overhead.

This chapter confronts these limitations by exploring the viability of end-to-end SQA in a zero-shot, resource-constrained setting. Our research is guided by the following key questions:

- How effective are standard cascade systems (ASR + LLM) for medical SQA, and what are their practical limitations in a zero-shot setting?

- To overcome the lack of real-world data, can a high-quality synthetic benchmark be created to enable standardized and reproducible SQA evaluation?

- Can an end-to-end methodology, which bypasses the need for explicit transcription, offer a more resource-efficient and performant alternative to cascade systems?

- Is it possible to adapt existing, large-scale pre-trained speech models to perform complex SQA tasks through zero-shot entailment, without any task-specific fine-tuning?

Our work, published at InterSpeech 2024 [169], makes the following contributions to address these questions:

- We introduce SpokenMedicalQA, the first open benchmark for medical Spoken Question Answering, featuring over 48 hours of synthetic audio across 8 distinct tasks. The benchmark is publicly available on the Hugging Face Hub[1].

- We propose and evaluate a novel, zero-shot audio-text entailment method that enables end-to-end SQA using pre-trained speech models.

- We demonstrate that our prompting method with Whisper achieves performance comparable to a cascade system using an LLM of an equivalent size (1.5B parameters), raising important questions about the role and required scale of the decoder in end-to-end SQA models.

- We conduct a comprehensive comparative analysis of our end-to-end approach against twelve different cascade system configurations, providing insights into performance-efficiency trade-offs for this task.

- We perform a fine-grained analysis of encoder layer contributions across multiple architectures, revealing how different models process speech for reasoning tasks and providing insights for designing more effective SQA models.

- We release all code for data synthesis, model evaluation, and analysis on GitHub to ensure full reproducibility and facilitate future research[2].

6.1 Medical Spoken Question Answering

In this section, we define the SQA task (Section 6.1.1) and present the open benchmark constructed from established medical datasets initially in textual format (Section 6.1.2). Additionally, we describe the audio prompt format (Section 6.1.3) and the SQA evaluation protocol (Section 6.1.4).

[1] https://huggingface.co/datasets/SpokenMedicalQA/SpokenMedicalQA
[2] https://github.com/qanastek/E2E-SQA-Medical-ZeroShot

6.1.1 Definition

We focus on multiple-choice SQA within the medical domain. As shown in the Figure 6.1, each instance comprises an audio question followed by four possible spoken responses, denoted as (q, o, c, a). Here, q represents the question, o denotes the options (labeled A to D), c indicates the correct answer, and a encapsulates the audio containing both the question and options. Questions are structured as single-turn interactions, devoid of dialogue. This evaluation relies solely on the model's internal knowledge without external information or span extraction. The primary objective is to assess end-to-end model performance in understanding and accurately choosing the correct answer from spoken input.

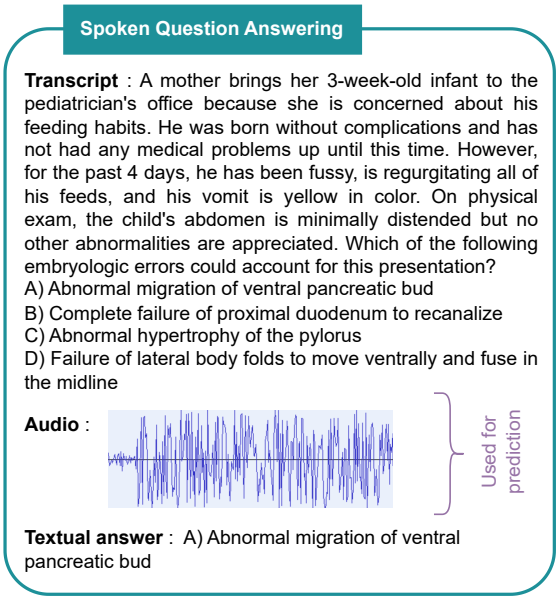


Figure 6.1: Spoken Question Answering Data Format.

6.1.2 Tasks Collection and Description

Recent years have seen significant progress in SQA datasets, such as Clotho-AQA [189], Spoken-SQuAD [175], and LibriSQA [316]. However, these datasets do not specifically target the healthcare domain or rely solely on audio inputs. The absence of SQA datasets in the medical domain hampers the development of question answering systems tailored to healthcare contexts. To address this gap, we propose synthesizing an audio dataset from existing textual multiple-choice question answering (MCQA) corpora. Our approach involves using Text-To-Speech (TTS) technology on these MCQA textual datasets to generate synthetic audios, leveraging advancements in TTS models that increasingly resemble human speech quality [151, 154]. We utilized the OpenAI TTS API (`tts-1`) to synthesize speech based on the questions and available options. The speakers were alternated through the 6 available voices to introduce diversity and realism into the dataset. The resulting audio files were sampled at 16,000 Hz and converted to WAV mono channel format.

Our reference texts were sourced from three open-source textual MCQA corpora in

English and already used in the chapter 5, namedly MMLU, MedQA and MedMCQA, all relevant to healthcare, featuring single possible answers and a four-option format. Note that only the test data are detailed here, as the proposed approaches operate under zero-shot conditions.

Our final benchmark encompasses 8 SQA tasks (including 6 from MMLU) derived from these 3 synthesized datasets. Table 6.1 summarizes the audio duration distribution according to the different labels available in the test set.

	MMLU	MedQA	MedMCQA	Total	# Doc.
A	1h50	5h55	5h41	13h28	1,936
B	1h54	5h08	4h31	11h33	1,648
C	1h50	5h49	3h57	11h37	1,519
D	3h03	4h28	3h30	11h03	1,442
Total	8h39	21h22	17h40	47h41	6,545

Table 6.1: Audio duration distribution according to the labels.

6.1.3 Audio Prompt Format

We standardized all textual MCQA datasets and synthesized them into audio format. These audio MCQAs serve as prompts for the studied and proposed SQA systems. Following experimentation with various formats and careful listening to the resulting audio outputs, we identified an effective format exemplified below in the Figure 6.2:

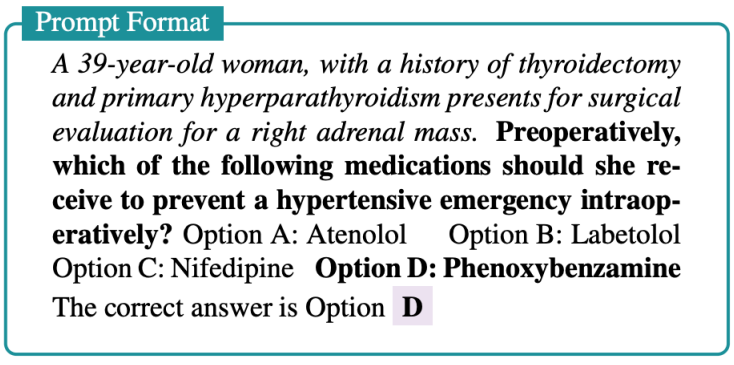


Figure 6.2: Audio prompt format.

6.1.4 Evaluation Metric

The evaluation of multi-choice SQA with a single correct answer resembles a multi-class classification task. The performance is here assessed for each task using *Accuracy*, which measures the proportion of correctly predicted answers compared to the total number of

questions. A prediction is considered accurate if it exactly matches the ground-truth answer, otherwise, it is classified as incorrect. Choosing the accuracy enables direct comparison with previous works on textual datasets [218, 168].

6.2 Studied and Proposed Methods

This section outlines the zero-shot approaches studied for SQA. Firstly, we introduce baseline models with cascade systems (Section 6.2.1). Then, we present models integrating our end-to-end audio-text entailment approach (Section 6.2.2).

6.2.1 Baseline Cascade Approaches

Our baseline models involve a two-stage process: transcription of audio inputs into text using an ASR module, followed by their processing with an LLM to select the correct answer to posed questions. We conducted experiments with various models to assess the impact of different ASR and LLM configurations on SQA performance. In the ASR stage, we compared the performance using the reference transcription (*Oracle*) against Whisper Small, Medium, and Large V2 ASR models to identify potential transcription error propagation issues. Subsequently, in the LLM stage, we compared the performance of an LLM similar in size to Whisper Large V2 (1.5 billion parameters), named Phi 1.5, against larger models based on the LLaMa 2 architecture, configured with 7B and 13B parameters, to assess the scalability of performance with model size. In total, we investigated 12 cascade system combinations.

During the second step of inference, the LLM predicts the next token based on the input prompt, generating probabilities for each token in the vocabulary. To ensure relevance, the vocabulary is filtered to include only relevant tokens (in this case, choice letters) corresponding to the expected answer options. This approach prevents the model from generating irrelevant tokens or hallucinations [184].

6.2.2 Zero-Shot End-To-End Entailment-Based Approaches

Numerous studies [113, 244] have underscored the advantages of leveraging Natural Language Inference (NLI) for textual zero-shot entailment and classification tasks. However, except for CLAP [296] and Pengi [78], based on contrastive learning and prefix-tuning respectively, a limited adaptation of such methodologies has been observed in speech-related literature, particularly with large-scale pre-trained audio models like Whisper and SpeechGPT. Our proposed zero-shot audio-text entailment method is integrated into the four previously mentioned models, aiming to assess the likelihood of a textual sequence matching an audio recording. In our setup, the audio contains the question and options, while the text represents classes A to D.

For Whisper [235], we utilize audio features and request individual log probabilities for

each letter using the format: *<|startoftranscript|> [A] <|endoftext|>*. The predicted class is determined by the highest average log probability. To comply with Whisper's 30-second limit for audio segments, we truncate segments beyond this duration to capture only the question and options. For SpeechGPT [310], we populate the model's context in a prompt filled with speech units obtained from HuBERT [128] representations discretized using k-means clustering on 1,000 clusters.

We then request the generation of one additional token for the model. Subsequently, we filter the vocabulary to retain only the log probabilities corresponding to letters A to D, as described earlier in Section 6.2.1. Pengi [78] undergoes minimal changes in the model, audio representation, and prompt format, maintaining a similar procedure. The approach is slightly adapted for the CLAP model [296], a dual encoder architecture trained with contrastive language-audio pre-training. Here, individual encoders process both speech and text. Given an audio sample (a) and a list of classes (o), we identify the best match among all pairs by calculating the cosine distance between their vector representations. The pair with the closest distance is considered the predicted match.

6.3 Results

In this section, we examine the zero-shot condition performance on our SQA tasks using first the baseline cascade models (Section 6.3.1), and then our entailment approach across various end-to-end models (Section 6.3.2).

6.3.1 Zero-Shot Cascade Approaches

Table 6.2 outlines the transcription performance, measured in Word Error Rate (WER), of Whisper ASR versions (Small, Medium, and Large V2) across various SQA tasks. Generally, Whisper Large V2 shows improved WER performance, except in MMLU Anatomy, where Whisper Medium performs better.

	Tasks	Whisper		
		S	M	L-V2
MMLU	Clinical KG	5.45	4.21	**3.30**
	Medical Genetics	6.19	4.59	**4.31**
	Anatomy	4.90	**2.68**	3.50
	Pro Medicine	5.66	4.68	**4.54**
	College Biology	4.54	2.91	**2.66**
	College Medicine	26.02	25.54	**24.74**
	MedQA	7.50	6.21	**5.84**
	MedMCQA	7.99	6.33	**6.10**
	Average	8.53	7.14	**6.87**

Table 6.2: Transcription performance (in WER) on each SQA task. Best result in bold and second best is underlined.

Tables 6.3 and 6.4 display the accuracy performance of the studied LLM-based zero-shot cascade methods using Whisper automatic transcriptions on multiple SQA tasks. Interestingly, the Whisper model with the lowest WER might not always be the optimal choice in a cascade approach, indicating a lack of direct correlation between WER and SQA accuracy. Conversely, SQA performance appears to depend on LLM size, with larger models yielding higher accuracy. Notably, there is an 11.67% difference between Phi 1.5 and LLaMa 2 13B in Whisper Medium results, highlighting the significant advantage of scaling up LLMs. Except for Phi 1.5, all models show improved performance with transcriptions compared to Oracle. This enhancement, particularly in LLaMa 2 architectures, may be attributed to their better adaptability to speech normalization formats, reduced punctuation, and increased noise.

		MMLU					
		Clinical KG	Medical Genetics	Anatomy	Pro Medicine	College Biology	College Medicine
Phi 1.5	Oracle	31.3	**39.0**	19.3	20.6	29.2	28.9
	Whisper Small	26.8	24.0	31.9	27.6	25.0	23.1
	Whisper Medium	27.9	20.0	35.6	27.6	25.7	24.9
	Whisper Large V2	31.7	19.0	34.1	24.6	26.4	26.0
Llama 2 7B	Oracle	21.5	30.0	18.5	18.4	25.7	20.8
	Whisper Small	29.4	31.0	25.2	33.5	31.9	31.2
	Whisper Medium	30.6	**39.0**	25.2	35.3	37.5	29.5
	Whisper Large V2	31.7	38.0	26.7	33.5	29.9	31.8
Llama 2 13B	Oracle	21.5	30.0	18.5	18.4	25.7	20.8
	Whisper Small	35.8	35.0	39.3	35.7	41.0	28.9
	Whisper Medium	**37.7**	36.0	**45.2**	**39.0**	**44.4**	**32.4**
	Whisper Large V2	34.7	38.0	37.0	**39.0**	39.6	**32.4**

Table 6.3: Accuracy (in %) of the zero-shot cascade methods on MMLU benchmarks. Highest value in bold and second best is underlined.

		MedQA	MedMCQA	Avg.
Phi 1.5	Oracle	27.7	31.2	28.4
	Whisper Small	25.5	25.9	26.2
	Whisper Medium	25.4	25.4	26.6
	Whisper Large V2	27.6	26.2	27.0
Llama 2 7B	Oracle	27.7	32.1	24.3
	Whisper Small	29.9	30.7	30.3
	Whisper Medium	29.5	31.1	32.2
	Whisper Large V2	28.7	30.8	31.4
Llama 2 13B	Oracle	27.7	32.1	24.3
	Whisper Small	36.2	<u>34.0</u>	35.7
	Whisper Medium	**37.4**	**34.1**	**38.3**
	Whisper Large V2	<u>36.8</u>	33.1	<u>36.3</u>

Table 6.4: Accuracy (in %) of the zero-shot cascade methods on other medical benchmarks and overall average. Highest value in bold and second best is underlined.

Furthermore, with LLaMa 2, Whisper Medium transcriptions emerge as the top performers. Notably, LLaMa 13B demonstrates a 1.95% overall accuracy gain over Whisper Large V2 and a 2.54% improvement over Whisper Small. Similar trends are observed in the 7B model, with increases of 0.8% over Large V2 and 1.9% over Small. The performance of the LLaMa 2 13B model in a zero-shot scenario with Whisper Medium transcriptions shows promising results.

6.3.2 Zero-Shot End-To-End Models' Capabilities

		MMLU					
		Clinical KG	Medical Genetics	Anatomy	Pro Medicine	College Biology	College Medicine
Whisper	Small	24.1	**31.0**	20.0	17.6	25.0	20.2
	Medium	**30.6**	20.0	17.8	<u>42.6</u>	<u>26.4</u>	30.6
	Large V2	27.5	24.0	26.7	20.2	20.1	19.6
CLAP	Unfused	26.8	23.0	24.4	37.1	**29.2**	<u>32.9</u>
	Large General	<u>29.4</u>	21.0	23.7	**44.5**	25.7	**34.1**
	Fused	21.5	<u>30.0</u>	18.5	18.4	25.7	20.8
Pengi	Base	24.9	26.0	**32.6**	21.3	19.4	24.8
	Base No Text Encoder	26.8	26.0	25.2	20.2	22.2	20.8
SpeechGPT	E2E	28.3	23.0	<u>29.6</u>	17.6	21.5	27.2
SpeechGPT	Oracle	36.2	32.0	27.4	35.7	29.9	34.1

Table 6.5: Accuracy (in %) of the zero-shot end-to-end models on MMLU benchmarks using our entailment method. Highest value in bold and second best is underlined, excluding SpeechGPT + Oracle (model aligned with reference transcriptions).

		MedQA	MedMCQA	Avg.
Whisper	Small	**27.7**	<u>30.6</u>	24.5
	Medium	21.9	22.5	26.5
	Large V2	25.8	27.4	23.9
CLAP	Unfused	23.1	19.7	<u>27.0</u>
	Large General	21.1	20.3	**27.5**
	Fused	**27.7**	**32.0**	24.3
Pengi	Base	24.0	24.4	24.7
	Base No Text Encoder	24.3	25.9	23.9
SpeechGPT	E2E	<u>26.4</u>	23.4	24.6
SpeechGPT	Oracle	24.4	27.2	30.8

Table 6.6: Accuracy (in %) of the zero-shot end-to-end models on other medical benchmarks using our entailment method. Highest value in bold and second best is underlined, excluding SpeechGPT + Oracle (model aligned with reference transcriptions).

Tables 6.5 and 6.6 outline the accuracy performance of zero-shot end-to-end models using our entailment method on our multiple-choice SQA benchmark. While the overall average accuracy remains similar across models, specific models demonstrate proficiency in particular tasks, with none consistently outperforming others across all tasks. Notably, Whisper Medium showcases competitive zero-shot performance, surpassing cascade setups with Phi 1.5 despite having approximately half the parameters. CLAP's contrastive modeling outperforms Phi 1.5 but falls short of LLaMa 2 7B. Impressively, despite its smaller size—153M parameters in its base form and 193M in its larger form—CLAP performs remarkably well, being 14.7 times smaller than Whisper Large V2 combined with Phi 1.5 and 44.3 times smaller with LLaMa 2 7B. SpeechGPT encounters challenges in zero-shot tasks from speech, contrasting its performance with text (Oracle), highlighting difficulties in directly handling speech modality representations, which need to be addressed in the future, with a better alignment approach. Notably, Whisper, especially Whisper Medium, occasionally outperforms cascade configurations with Phi 1.5 in zero-shot scenarios. Specific tasks exhibit varying levels of difficulty for different models; for instance, MedMCQA yields high results with Whisper Small and CLAP Fused, while MMLU College Medicine favors Whisper Medium, CLAP Unfused, and CLAP Large General. SpeechGPT generally underperforms across most tasks, except for MMLU Anatomy and MedQA, where it outperforms most other models. Despite the small performance improvement over cascade systems, which is linked to the zero-shot setting, E2E systems can be enhanced by scaling with better quality SQA data and increasing the number of parameters to see if they follow scaling laws similar to LLMs.

6.4 Analysis of Encoder Layers

This section presents an extensive analysis to pinpoint the critical location of information crucial for SQA tasks within the layers encoding the audio signal. To conduct this analysis, we extracted a subset of the MedMCQA training set consisting solely of audio sequences shorter than 30 seconds, which comprised 97.56% of the data, resulting in 120 hours of spo-

ken data. This subset was partitioned into training and validation sets using an 80%/20% ratio, yielding 95 hours and 23 hours, respectively. Our experimental approach involves fine-tuning audio encoders and introducing an intermediate trainable layer of equal size to the number of encoder layers. This intermediate layer selects information from the encoder's layers through a weighted sum of their representations when feeding the classification head. The objective of this weighted encoder layers approach is to analyze the necessity of specific layers for executing the SQA task while enhancing model understanding.

As depicted in Figure 6.3, illustrating cumulative weights across encoder layers, Whisper models exhibit a propensity to concentrate information in the final layers, aligning with prior research findings [302]. This indicates that these audio-based models effectively utilize the last layer to represent textual information, possibly due to heavy reliance on the decoder.

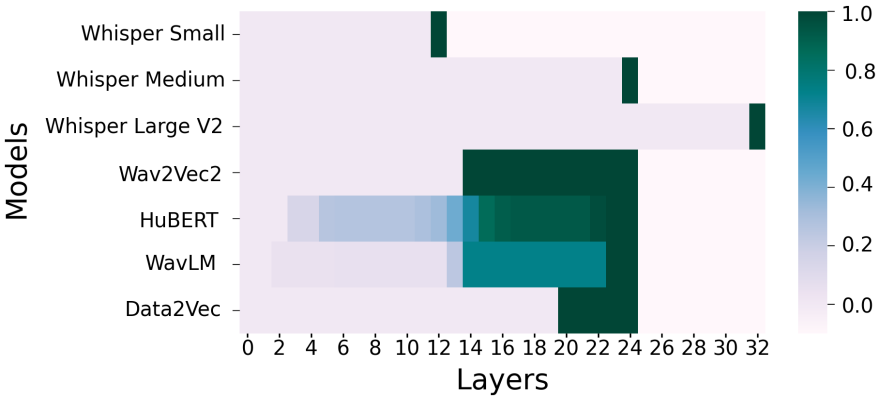


Figure 6.3: Cumulative weights according to encoder layers.

In contrast, Wav2Vec [17] and Data2Vec [16] primarily rely on a single intermediate layer, specifically the 15th and 21st layers, respectively. However, HuBERT [128] and WavLM [48] adopt a different strategy, integrating information from a broader range of layers. HuBERT integrates data from 12 layers, while WavLM incorporates information from 4 layers distributed across various regions of the encoder.

6.5 Conclusion

This study introduces a novel synthetic Spoken Question Answering (SQA) dataset tailored specifically to the medical domain. We conducted zero-shot comparative analyses of end-to-end speech methodologies using a new entailment technique against cascade speech transcription and an LLM module. Our experiments and analysis demonstrate the effectiveness of our end-to-end approach, yielding performances comparable to those achieved by cascade models of similar sizes. Moving forward, we aim to explore the utilization of speech alignment techniques with LLMs to enhance end-to-end question answering performance, with a particular emphasis on improving outcomes in low-resource domains such as healthcare. Our research faced multiple constraints. Using limited speaker variety for synthetic audio may reduce accuracy compared to natural speech, affecting response preci-

sion. Simplifying task formulation lacks genuine human interaction dynamics but enables metric-based assessments, enhancing model reproducibility and cost efficiency. Finally, our study neglects multilingual contexts, highlighting the need for additional exploration in diverse linguistic settings.

Chapter 7

Text-Speech Language Models with Improved Cross-Modal Transfer by Aligning Abstraction Levels

The preceding chapters have built a clear trajectory: from establishing the need for specialized text-based models like BioMistral (Chapter 5), we moved to the domain of speech, where the limitations of existing models for complex reasoning tasks became clearly evident (Chapter 6). That work highlighted a fundamental challenge: simply connecting speech to a powerful language model is not enough. This final research chapter addresses that challenge at its core, proposing a new way for Text-Speech Language Models (TSLMs) to perform an effective fusion of the two modalities and improving alignment.

The dominant method for creating TSLMs is to perform a simple vocabulary expansion, appending speech tokens to a pre-trained text LLM. Our main hypothesis is that this method's effectiveness is limited by an "abstraction gap" that emerges between the two modalities. Speech tokens, representing low-level acoustic and phonetic information, are fundamentally different from the high-level, semantic sub-word tokens that text LMs are designed to process. Feeding these disparate representations directly into a unified architecture constrains the model's ability to learn shared concepts and transfer knowledge effectively across modalities.

This chapter confronts this challenge by proposing an architecture that explicitly accounts for these different levels of abstraction. Our research is guided by the following questions:

- How can we effectively bridge the abstraction gap between low-level speech tokens and the high-level representations processed by a text LM?
- Can dedicated adapter modules, which compose speech representations before they enter the main text LM backbone, improve cross-modal alignment and knowledge transfer?

- Since different layers of an LM capture features at varying levels of abstraction, can a dynamic mechanism that pools information from multiple layers on-the-fly lead to more effective speech generation?
- Can an architecture designed around these principles outperform the standard vocabulary expansion approach, even with significantly less training data and computational resources?

This chapter is based on the publication (Cuervo et al., 2025) [71], a collaborative work with Santiago Cuervo, Adel Moumen, and Ricard Marxer. My primary contributions to this project included the implementation and execution of the experiments and active participation in developing the core research ideas. Our main contributions are:

- We design and implement a novel TSLM architecture, featuring modality-specific input/output adapters and a dynamic layer pooling mechanism, to explicitly align feature abstraction levels between speech and text.
- We introduce and release SmolTolk on HuggingFace [1], a suite of TSLMs in three sizes (150M, 400M, and 2B parameters) that achieve state-of-the-art performance on speech language modeling benchmarks.
- We promote reproducibility and reduce the computational barrier for future research by releasing our processed evaluation and interleaved training data in their final discretized form.
- We demonstrate that our approach is significantly more compute-efficient, outperforming previous models that are up to $4.5\times$ larger and trained on over $60\times$ more data.
- Through extensive representation analysis, we provide strong evidence that our architecture improves cross-modal transfer by increasing feature compositionality and the overlap between text and speech representation subspaces.
- We show that our dynamic pooling mechanism learns to perform unsupervised word segmentation as an emergent capability, validating our hypothesis about its function.

7.1 Text-Speech Language Models

Text-Speech Language Models (TSLMs) model the joint probability of text and speech token sequences as

$$P(\mathbf{w} = w_1, \ldots, w_n) = \prod_{i=1}^{n} P(w_i | w_1, \ldots, w_{i-1}), \tag{7.1}$$

where $w_i \in \mathcal{V}_t \cup \mathcal{V}_s$, with $\mathcal{V}_t$ and $\mathcal{V}_s$ denoting text and speech vocabularies. TSLMs are typically decoder-only transformers [275] optimized to minimize the *Negative Log Likelihood*:

[1] https://huggingface.co/ParoleLM

$$\mathcal{L}_{LM} = -\sum_{i=1}^{n} P(w_i|w_1, \ldots, w_{i-1}). \tag{7.2}$$

Tokens are mapped to embeddings via a linear function $E \in \mathbb{R}^{(|\mathcal{V}_t|+|\mathcal{V}_s|)\times d}$, where d is the embedding dimension. The sequence $E(w_1), \ldots, E(w_n)$ is processed by a stack of decoder-only transformer layers, producing contextual representations $(\mathbf{c}_1, \ldots, \mathbf{c}_n)$, where $\mathbf{c}_i \in \mathbb{R}^d$ and each $\mathbf{c}_i$ depends on $\mathbf{c}_{\leq i}$. A linear projection $U \in \mathbb{R}^{d\times(|\mathcal{V}_t|+|\mathcal{V}_s|)}$ maps these to logits defining $P(w_{i+1}|\mathbf{c}_i)$.

Text tokens are typically obtained via sub-word tokenization [249], while speech tokens are derived through quantization of self-supervised representations. *Linguistic tokens*[2] are obtained from MLM models like HuBERT [127] and capture phonetic content. *Acoustic tokens*, extracted from autoencoder models, preserve speech signal details, including paralinguistic and acoustic variability [308].

Training. TSLMs are often trained via *vocabulary expansion and speech fine-tuning* of text LMs. Vocabulary expansion extends the embedding function and output projections over $\mathcal{V}_t$ to include $\mathcal{V}_s$, while the rest of the LM remains unchanged. Fine-tuning methods vary in data mixture: [245] train on mixed speech-text tasks (TTS, ASR, speech-to-speech translation), while [55] use word-level alignments to switch modalities within a sequence. This interleaved text-speech strategy was shown to be crucial for cross-modal transfer, later validated and scaled up by [216] and [309], achieving state-of-the-art speech LM performance.

[2]Often referred to in the literature as *"semantic tokens"*, though we argue this is a misnomer as they primarily encode phonetic information [53].

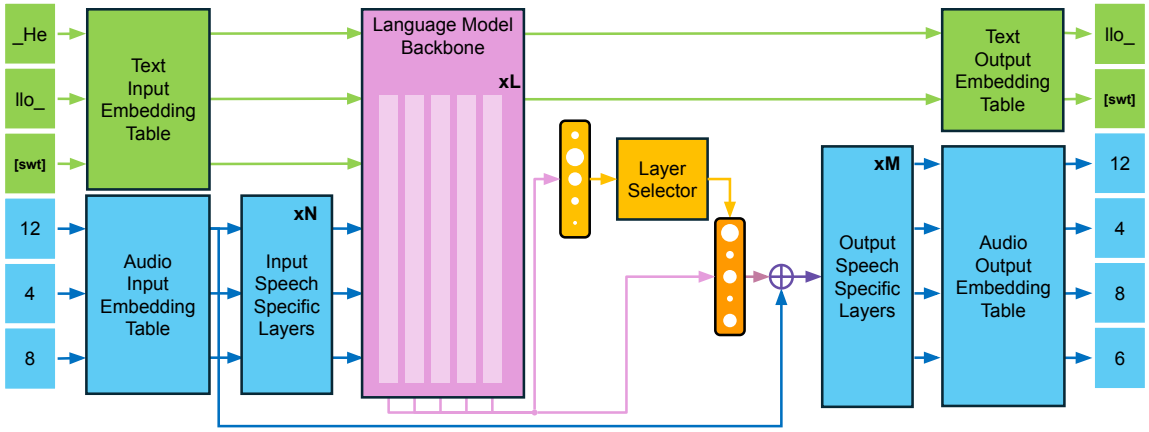


Figure 7.1: General diagram of our proposed architecture. Input tokens are processed through modality-specific embedding tables. Speech representations (blue) pass through speech-specific layers, bridging the gap between speech and text inputs, before merging with text embeddings (green) in the text LM backbone. A weighted average of the backbone's representations, computed using fixed and dynamic learned weights yields a representations better suited for speech prediction, which is processed through output speech-specific layers to predict the next speech tokens. In parallel, text tokens are predicted from the final text LM representation.

7.2 Proposed Method

Our method is illustrated in Figure 7.1. Below, we describe our model's architecture and its training process.

7.2.1 Model

We apply the embedding function E to the input sequence composed of text and speech tokens $\mathbf{w}$, yielding a sequence of text and speech embeddings $(E(w_1), \ldots, E(w_n)) = (\mathbf{z}_1, \ldots, \mathbf{z}_n)$.

Input speech adapter. Contiguous chunks of speech embeddings are processed through an input adapter $A_{in} : \mathbb{R}^d \to \mathbb{R}^d$, a stack of decoder transformer layers. The input adapter is meant to compose the speech token embeddings into higher level representations, which we believe should facilitate cross-modal transfer by matching the abstraction level expected by the text LM input. For a contiguous chunk of speech embeddings $(\mathbf{z}_i, \ldots, \mathbf{z}_{i+k})$, the input adapter outputs a sequence $(\mathbf{z}'_i, \ldots, \mathbf{z}'_{i+k})$.

The output of the input adapter and the text embeddings are fed into the text LM transformer at their respective positions in the input. For instance, given the input sequence $(w_1, w_2, w_3, w_4, w_5)$, where only the third element is a text token, the sequence passed to the transformer layers after the adapter would be $(\mathbf{z}_1, \mathbf{z}_2, E(w_3), \mathbf{z}_4, \mathbf{z}_5)$. For each trans-

former layer l, we obtain a sequence of contextual representations $(\mathbf{c}_1^{(l)}, \ldots, \mathbf{c}_n^{(l)})$. As in regular text LMs and TSLMs, the text output logits are computed by applying U to the contextual representations $\mathbf{c}_i^{(L)}$ at the last layer L. The speech output logits are computed as described next.

Dynamic layer pooling and speech input residual. We believe speech language modeling requires the model to switch between two modes of operation depending on whether a word is being generated or a new word is about to start. Within a word, the model should use low-level representations encoding the spoken word and the current speech token within that word, as these features fully determine the next speech token to be emitted. When generating a new word, the model should instead rely on representations predictive of upcoming words, such as those in the latter layers of the text LM. To enable this behavior, we use a learnable mechanism that attends to different layers' representations in an input-dependent manner. A linear layer selector $S : \mathbb{R}^d \times \mathbb{R}^L$ maps a contextual representation $\mathbf{c}'_i$ to a vector of weights $\boldsymbol{\omega}_i = (\omega_i^{(1)}, \ldots, \omega_i^{(L)})$. These weights are softmax-normalized and used to compute an input-dependent weighted average yielding a contextual multi-level representation $\bar{\mathbf{c}}_i$:

$$\begin{aligned} \boldsymbol{\omega}_i &= \text{Softmax}(S(\mathbf{c}'_i)) \\ \bar{\mathbf{c}}_i &= \sum_{l=1}^{L} \omega_i^{(l)} \mathbf{c}_i^{(l)} \end{aligned} \tag{7.3}$$

A key question is which representation to use as the layer selector's input $\mathbf{c}'_i$. We found that last layer's representations $\mathbf{c}_i^{(L)}$ were not well suited as they often resulted in the selector collapsing to select a single layer. Rather than searching for the best layer—which would likely vary across different architectures—we use a weighted average of the contextual representations with learned input-independent weights:

$$\mathbf{c}'_i = \sum_{l=1}^{L} \phi^{(l)} \mathbf{c}_i^{(l)} \tag{7.4}$$

where $\boldsymbol{\phi} = (\phi^{(1)}, \ldots, \phi^{(L)})$, $\phi^{(l)} \in \mathbb{R}$ are learned weights.

To provide information about the current speech token, we add a residual connection from the speech input embeddings to the multi-level contextual representation:

$$\bar{\mathbf{c}}'_i = \bar{\mathbf{c}}_i + \mathbf{z}_i \tag{7.5}$$

$\bar{\mathbf{c}}'_i$ contains both the information selected by layer pooling from the text LM layers and the current speech token.

Output speech adapter. The output adapter $A_{out} : \mathbb{R}^d \to \mathbb{R}^d$, a stack of decoder transformer layers, takes $\bar{\mathbf{c}}'_i$ as input and refines it into a representation predictive of upcoming speech tokens, upon which the speech output logits are computed by applying U.

7.2.2 Training

We train our model to optimize the negative log-likelihood defined in the Equation 7.2 on sequences from a data mixture similar to that proposed by [55], including unimodal speech and text samples, as well as interleaved text-speech samples. Unlike previous works, we do not include ASR or TTS samples.

Two stage training. Training follows a two-stage process. In the first stage, the text LM backbone is frozen, and only the newly added modules are trained on interleaved text-speech modeling for approximately 3% of the total training iterations. This stage is introduced to mitigate text capability forgetting, as suggested by preliminary experiments. In the second stage, the full model is trained on the complete data mixture for the remaining iterations.

Preventing layer selector collapse. In larger models, we observed that the layer selector S sometimes collapses early in training, attending to a single layer. To mitigate this, we add an entropy maximization term to the loss to encourage diversity in its output:

$$\mathcal{L} = \mathcal{L}_{LM} + \beta \frac{1}{n} \sum_{i=1}^{n} \sum_{l=1}^{L} \omega_i^{(l)} \ln(\omega_i^{(l)}) \tag{7.6}$$

where β is a hyperparameter that balances the LM objective and the entropy regularization term.

7.3 Experimental Setup

7.3.1 Models and Training

We use SmolLM models [28] as text LM backbones, available in three sizes: 135 million, 360 million, and 1.7 billion parameters. We refer to the models resulting from applying our method to the SmolLM backbones as SmolTolk-148M, SmolTolk-396M and SmolTolk-1.9B. We selected SmolLM due to its strong performance relative to other models of comparable size, achieving state-of-the-art results under the 2B parameter budget, as well as its availability in multiple sizes, which aligns with our goal of evaluating models across a range of capacities. Following [216], and to improve the modeling of long-range dependencies, we increase the RoPE base frequency from 10,000 to 100,000.

The input and output adapters consist of transformer layers matching the text backbone architecture. In initial experiments, we observed that using more than two adapter layers led to only marginal or no improvements; thus, we use two layers for all our models. This aligns with findings from [271] on adapting speech representations to frozen text LMs. Across models, adding the adapters introduces a parameter overhead of approximately 10%.

For speech tokenization, we follow [116], using the same tokenizer, which quantizes HuBERT representations extracted at 25 Hz into a 500-token vocabulary. As is common in

speech LM training on linguistic tokens, we collapse adjacent token repetitions.

All LMs are optimized using AdamW [195] with a weight decay of 0.1. We use a constant learning rate of $3e^{-4}$ for the 135-million and 360-million models, as well as the 1.7-billion baseline, and $1e^{-4}$ for SmolTolk-1.9B. For SmolTolk-396M and SmolTolk-1.9B, we set $\beta = 0.01$ in Equation 7.6.

We use a batch size of 1 million tokens with the full 2048-token context. Unless stated otherwise, each batch contains equal proportions of speech, text, and interleaved text-speech. All models are trained for 16 billion tokens, except the 1.7-billion model, which is trained for up to 32 billion.

Model	Num. Layers	Num. Heads	Num. KV Heads	Emb. Dim.	Hidden Dim.
SmolLM-135M	30	9	3	576	1536
SmolLM-360M	32	15	5	960	2560
SmolLM-1.7B	24	32	32	2048	8192

Table 7.1: Backbones architectural hyperparameters.

Table 7.1 describes the SmolLM [28] backbones architectural hyperparameters. All SmolTolk models use two layer input and output adapters with the same architecture as the backbone layers. Table 7.2 describes the resulting models after vocabulary expansion.

Model	Backbone	Num. Layers	Text params	Speech Params
Baseline-135M	SmolLM-135M	30	135M	0.29M
Baseline-360M	SmolLM-360M	32	360M	0.48M
Baseline-1.7B	SmolLM-1.7B	24	1.7B	1M
SmolTolk-150M	SmolLM-135M	34	135M	15M
SmolTolk-400M	SmolLM-360M	36	360M	40M
SmolTolk-2B	SmolLM-1.7B	28	1.7B	270M

Table 7.2: Models description.

As mentioned previously, we use a linear layer with bias as the dynamic layer selector S. We experimented with simple non-linear *Multi-Layer Perceptron* (MLP) selectors; however, these were prone to collapse and resulted in worse overall performance. That said, a more carefully designed non-linear selector could potentially perform better. We also explored alternative ways to define the contextual representation $\mathbf{c}'_i$ (Equation 7.3) used as input for the layer selector. Instead of a learned weighted average, we tried concatenating low-dimensional linear projections from each layer's representations, but this performed worse.

Regarding training, we tuned the learning rate for each model, including baselines, to be as high as possible without causing instabilities or increasing text data validation loss,

which we considered a sign of text capability forgetting. We also experimented with learning rate schedules for text/backbone parameters, but a constant rate performed better.

For our experiments, we used NVIDIA H100 GPU nodes, each featuring four NVIDIA H100 80GB SXM5 GPUs, dual Intel Sapphire Rapids 48-core processors, 512GB of RAM, and four NVIDIA ConnectX-7 400Gb/s InfiniBand network adapters. SmolTolk-2B used a per-GPU batch size of 8 million tokens and was trained on 64 GPUs across 16 nodes, taking approximately 10 hours to process 32 billion tokens. All models were trained using `bfloat16` mixed precision with FlashAttention-2 [75] and PyTorch `compile`.

7.3.2 Evaluation

Metrics. For downstream evaluation, we use standard zero-shot metrics from the speech language modeling literature. We assess syntactic knowledge using the sBLIMP benchmark [213], which measures the model's accuracy in selecting a syntactically correct utterance over an incorrect one based on estimated likelihood. Semantics and commonsense reasoning are evaluated using the sStoryCloze and Topic-sStoryCloze benchmarks [116], which measure accuracy in selecting the correct continuation of a given context based on predicted likelihood. To measure cross-modal transfer, following [216] and [309], we evaluate sStoryCloze and Topic-sStoryCloze in four settings: speech context to speech continuation (S), text context to speech continuation (T→S), speech context to text continuation (S→T), and text context to text continuation (T).

We also report text performance on MMLU [120] before (`pre`) and after (`post`) speech training to assess whether fine-tuning causes forgetting of text capabilities, as seen in other TSLMs [216, 89]. We evaluate MMLU following the guidelines for the SmolLM models: https://huggingface.co/HuggingFaceFW/ablation-model-fineweb-edu#evaluation.

Baselines. We compare SmolTolk to models trained with the same text LM backbones and data but using regular vocabulary expansion, referring to these as baselines. We also compare against state-of-the-art TSLMs: Spirit LM [216], Moshi [89], and the 1.5-billion and 9-billion models from Zeng et al. [309]. Unlike the others, Moshi employs a multi-codebook architecture and relies heavily on text-guided speech generation.

7.3.3 Data

Modality	Dataset	Tokens		Sampling ratio
		Text	Speech	
Text	FineWeb-Edu [28]	4B	—	0.7
	Cosmopedia-v2 [28]	4B	—	0.15
	Python-Edu [28]	2B	—	0.08
	FineMath [191]	2B	—	0.06
Speech	LibriSpeech [224]	—	67M (960 hours)	—
	LibriLight [145]	—	3.7B (53k hours)	—
	SWC [23]	—	32M (1k hours)	—
	Tedlium [121]	—	0.1B (1.6k hours)	—
	People [98]	—	0.5B (7k hours)	—
	Vox Populi [281]	—	1.6B (24k hours)	—
	sTinyStories [70]	—	4.8B (72k hours)	—
Interleaved text-speech	LibriHeavy [147]	313M	3.1B (50k hours)	0.37
	sTinyStories [70]	800M	4.8B (72k hours)	0.53
	SWC [23]	3.6M	26M (800 hours)	0.1

Table 7.3: Datasets statistics. Speech datasets were sampled according to their size.

Speech datasets. We use a collection of publicly available English speech datasets for training: LibriSpeech [224], LibriLight [145], SWC [23], Tedlium [121], People's Speech [98], Vox Populi [281], and sTinyStories [70]. These datasets contain a total of 10.89 billion speech tokens.

Text datasets. We use a 12-billion-token subset of the SmolLM corpus [28]. Unlike [216], we include math and code data, aiming to better preserve text capabilities. Our data distribution matches that used for pre-training SmolLM models, as reported in `https://github.com/huggingface/smollm/blob/main/pre-training/`.

Text-Speech datasets. We use the forced aligner from [233] to obtain word alignments for the LibriHeavy [147], sTinyStories, and SWC datasets. Interleaved samples are generated on the fly during batch sampling by randomly switching modalities within the input sequence. Following [216], we randomly select word spans so that each text sequence contains 10–30 words and each speech sequence 5–15 words, balancing the proportion of speech and text tokens in each sample.

7.4 Experiments and Results

Table 7.4 presents the benchmark results. For brevity, we report only the results for the 1.7-billion baseline. Smaller baselines models underperformed relative to the larger one.

Model	Params.	Tokens	BLIMP		tStoryCloze				sStoryCloze				MMLU
			T	S	T	S	T→S	S→T	T	S	T→S	S→T	T (post/pre)
Textless Speech LMs													
GSLM [173]	100M	—	Ø	54.2	Ø	66.6	Ø	Ø	Ø	53.3	Ø	Ø	Ø
AudioLM [34]	150M	—	Ø	**64.7**	Ø	—	Ø	Ø	Ø	—	Ø	Ø	Ø
TWIST [116] cold-init 1.3B	1.3B	10.8B	Ø	56.5	Ø	—	Ø	Ø	Ø	—	Ø	Ø	Ø
TWIST [116] 1.3B	1.3B	10.8B	Ø	57.0	Ø	70.6	Ø	Ø	Ø	52.4	Ø	Ø	Ø
TWIST [116] 7B	7B	36B	Ø	59.0	Ø	74.1	Ø	Ø	Ø	55.3	Ø	Ø	Ø
TWIST [116] 13B	13B	36B	Ø	59.2	Ø	76.4	Ø	Ø	Ø	55.4	Ø	Ø	Ø
Cuervo et al. [70] best	823M	82B	Ø	61.3	Ø	78.0	Ø	Ø	Ø	56.7	Ø	Ø	Ø
SyllableLM [13]	300M	1.2B	Ø	**63.7**	Ø	75.4	Ø	Ø	Ø	—	Ø	Ø	Ø
AlignSLM [185] 7B	7B	—	Ø	62.3	Ø	86.8	Ø	Ø	Ø	61.1	Ø	Ø	Ø
Slam (scaled) [197]	358M	16.7B	Ø	61.1	Ø	84.2	Ø	Ø	Ø	61.3	Ø	Ø	Ø
Previous Text-Speech LMs													
Spirit LM [216]	7B	∼175B	73.3	59.7	95.8	**90.5**	78.6	**94.3**	74.0	**66.3**	**64.7**	**71.7**	37.7 / 39.0
LAST [271]	∼390M	—	—	56.8	—	—	—	—	—	—	—	—	—
Moshi [89]	7.7B	2.1T	—	58.8	—	83.0	—	—	—	60.8	—	—	49.8 / 54.3
Zeng et al. [309] 1.5B	1.5B	1T	—	—	—	77.5	81.4	90.1	—	55.4	58.6	64.0	—
Zeng et al. [309] 9B	9B	1T	—	—	—	83.0	**85.0**	**93.6**	—	**62.4**	63.2	**76.3**	—
Ours													
Baseline 135M	135M	16B	79.0	52.0	87.0	73.2	53.3	52.7	63.9	54.0	53.8	53.7	30.3 / 30.2
Baseline 360M	360M	16B	79.8	52.4	90.4	74.1	53.1	53.8	68.4	54.0	52.1	53.1	34.5 / 34.0
Baseline 1.7B	1.7B	16B	79.9	56.3	92.8	77.5	72.6	67.3	72.5	53.0	57.0	57.6	**40.0 / 40.0**
Baseline 1.7B	1.7B	32B	79.8	58.1	92.9	81.3	76.3	74.0	73.5	55.1	59.0	59.2	39.2 / 40.0
SmolTolk-150M	150M	16B	79.4	58.0	88.4	82.0	75.2	81.0	64.1	55.0	58.8	58.4	30.0 / 30.2
SmolTolk-400M	400M	16B	79.8	59.4	91.3	84.6	80.9	85.0	68.4	57.5	62.3	62.1	34.0 / 34.2
SmolTolk-2B	2B	16B	80.2	61.4	92.6	87.5	83.9	86.0	73.2	60.0	64.0	63.4	**40.0 / 40.0**
SmolTolk-2B	2B	32B	80.2	61.9	92.6	**87.6**	**84.3**	87.1	73.6	61.4	**64.2**	64.2	**40.1 / 40.0**

Table 7.4: Downstream evaluations. The **best model** in each task is shown in bold and underlined. The **second best** is shown in bold. For Spirit LM we report the results for the open-weights version. For other models we present the results reported by the authors.

Our method significantly outperforms the baseline using regular vocabulary expansion across all tasks. One might attribute this difference to the model size increase induced by the added modules. To isolate this factor, Figure 7.2 shows the scaling behavior of the negative log-likelihood (NLL) on the LibriSpeech dev set (top) and the tStoryCloze benchmark (bottom) as a function of compute (in FLOPs), which accounts for model size differences and enables a fairer comparison. The figure demonstrates that, across the entire compute range, our models consistently outperform their respective baselines. Notably, SmolTolk-148M outperforms the 360-million baseline despite being less than half its size, suggesting that factors beyond model size drive the performance difference.

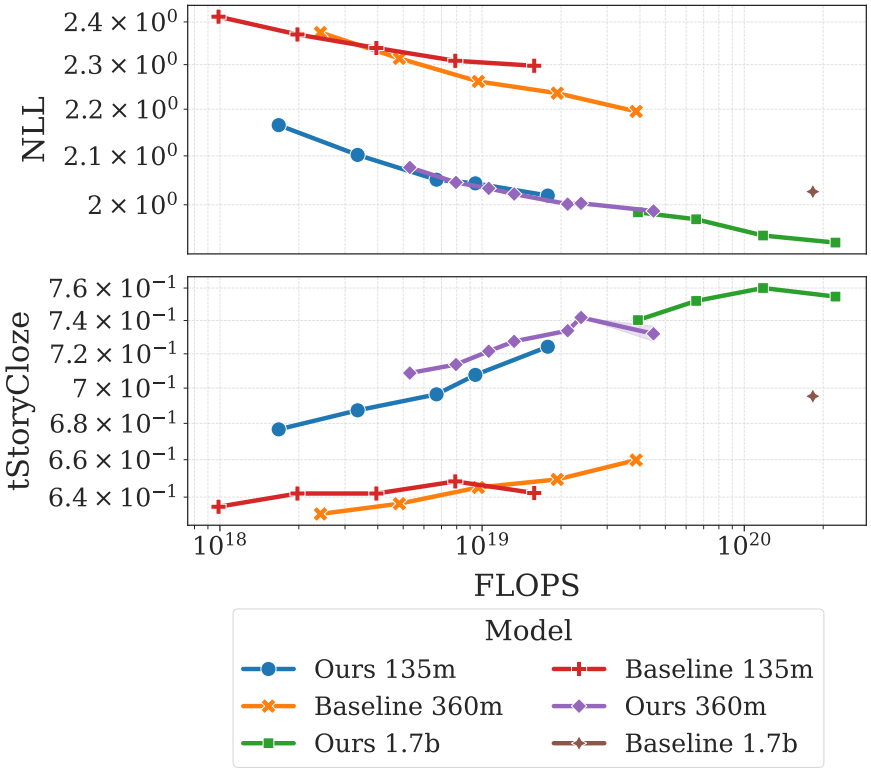


Figure 7.2: Scaling of the LibriSpeech dev set negative log-likelihood and tStoryCloze accuracy with respect to training compute (in FLOPs) for our models and baselines. For the Baseline 1.7-billion model, only the final checkpoint is shown, as the earlier ones were unfortunately lost.

Compared to state-of-the-art TSLMs, SmolTolk-1.9B outperforms Zeng et al. [309]-1.5B—the only other model under 2 billion parameters—despite using over 20× less training compute. Notably, even SmolTolk-396M outperforms Zeng et al. [309]-1.5B on tStoryCloze S and sStoryCloze S and T→S. SmolTolk-1.9B performs comparably to larger models across most tasks, except for S→T, where the gap is larger. It also achieves the best performance on the sBLIMP syntactic task.

Our setup, including the baselines, exhibits less deterioration in text MMLU performance relative to other TSLMs. We attribute this to our decision to use a text fine-tuning distribution that matches the one used during pre-training.

7.4.1 Ablation Study

Overall, the results indicate that our design choices enhance multimodal performance. To better understand each component's contribution, we conduct an ablation study in Table 7.5 on the 360-million parameter model by systematically removing elements and evaluating their impact.

Model	tStoryCloze				sStoryCloze			
	T	S	T→S	S→T	T	S	T→S	S→T
SmolTolk-396M	91.3	**84.6**	**80.9**	**85.0**	**68.8**	**57.8**	**62.3**	**62.1**
–Dyn. pooling	90.8	84.0	80.1	83.9	68.2	57.5	60.9	61.6
–Layer pooling	**91.4**	82.6	77.8	82.0	68.4	57.5	60.1	60.1
–In Adapter	90.1	82.3	70.1	75.6	68.1	55.6	56.3	57.4
–Out Adapter	90.7	80.7	76.1	84.1	68.1	54.9	60.0	60.0
–Adapters	89.9	77.6	58.8	63.7	68.1	52.5	51.5	54.8
–Residual	91.0	83.1	80.7	82.3	68.8	56.6	61.4	60.8
Baseline 360M	90.3	74.1	53.3	53.8	68.5	54.5	52.1	53.9

Table 7.5: Ablation Study. "–" denotes removal. "*–Dyn. pooling*" uses fixed learned weights instead of dynamic ones from the layer selector, while "*–Layer pooling*" entirely disables multi-layer pooling, relying only on the last text LM layer.

We observe that removing any component degrades performance across most metrics, confirming the importance of our design choices. Eliminating all adapters results in the steepest drop—especially in cross-modal transfer, highlighting their role in bridging representations. The input adapter seems to be of greater importance for cross-modal transfer than the output adapter, underscoring the importance of merging modalities early in processing. Layer pooling also provide consistent gains, demonstrating the benefits of allowing the model to use multiple abstraction levels for speech language modeling. Removing the residual connection also causes a consistent decrease in performance.

7.4.2 Representation Analysis

To gain deeper insight into the impact of our design choices, we analyze the learned representations across different model variants. Specifically, we investigate how the introduced architectural components influence feature abstraction and alignment between text and speech modalities. We focus on two key aspects: (1) the model's capacity to abstract high-level features and (2) the shared structure of text and speech representations.

To assess (1), we follow [273] and use the intrinsic dimensionality of representations as a proxy for compositionality. To evaluate (2), we compute the principal components of speech and text representations on paired data and measure how much variance is explained when projecting one modality's representations onto the other's principal components. The intuition is that knowledge transfer can be quantified by how much the model utilizes the same subspaces to represent equivalent data across modalities. We apply these analyses to the 360-million parameter models, including the architectural ablations from Table 7.5. The results are shown in Figures 7.3 and 7.4.

Intrinsic dimensionality and subspace overlap. The intrinsic dimensionality and subspace overlap are estimated using five batches of 10k representations each, totaling 50k samples. We compute each metric per batch and report average and deviations. To obtain

each representation, we extract random subsequences of 20 words and use the final representation in the sequence as the sequence representation. For the intrinsic dimensionality text samples are randomly drawn from FineWeb-Edu, while speech samples are taken from the full set of speech datasets. For the subspace overlap we use paired samples from the sTinyStories dataset.

Since transformer architectures exhibit large activation outliers, we truncate feature elements (i.e., individual activations) that exceed the 95th percentile across the entire 50k sample set.

To estimate the intrinsic dimensionality we use the *Generalized Ratios Intrinsic Dimension Estimator* (GRIDE) [77] implementation in `dadapy` [100] and follow the procedure described by [51].

Word segmentation. We apply a peak detector to the sequences of last-layer dynamic weights, $\boldsymbol{\omega}^{(L)}$, using SciPy's `find_peaks` tool. Performance is evaluated as a binary prediction task, where a peak indicates the prediction of a boundary at a given position. We allow a tolerance of one token to account for noise in the boundary annotations. As in standard binary prediction tasks, we use recall, precision, and the F1-score as performance metrics. Additionally, we use the R-value [241], which penalizes trivial over-segmentation solutions. To optimize performance, we tune the prominence parameter of the peak detector over a grid $(0, 0.15]$ with steps of 0.01 so as to maximize the R-value, following [156].

From Figure 7.3, we observe that different components of our architecture significantly impact the model's ability to compose higher-abstraction features. All our models, except the one without layer pooling, achieve a higher intrinsic dimension, suggesting that layer pooling is essential for enabling compositionality. The effect of the adapters is also evident: the absence of input adapters leads to lower compositionality in earlier layers, while the absence of output adapters results in an overall reduction in intrinsic dimension.

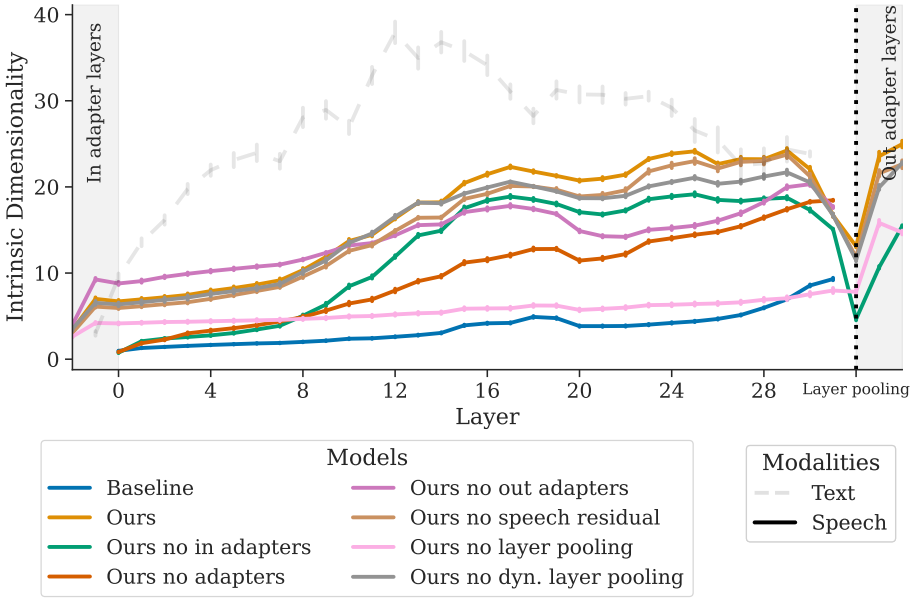


Figure 7.3: Intrinsic dimensionality of representations across layers for the the 360-million parameter model.

Figure 7.4 supports similar conclusions, showing that our architecture enables the highest degree of subspace overlap between modalities. As before, adapters and layer pooling are crucial for cross-modal transfer. The absence of input adapters leads to low subspace overlap in earlier layers, while the absence of output adapters reduces overlap in later layers. We hypothesize that this occurs because the model repurposes later layers to produce representations predictive of upcoming speech tokens. In this study the speech input residual has minimal effect.

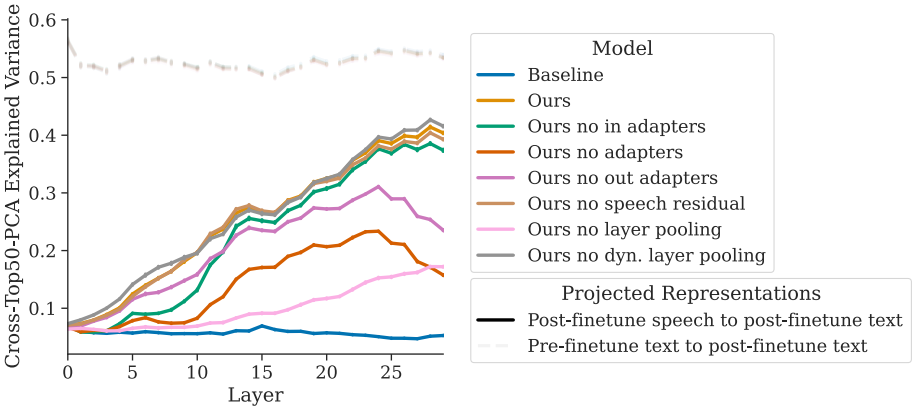


Figure 7.4: Variance explained by projecting one modality's representations onto the top 50 principal components of the other across model's layers for the 360-million parameters models.

What is dynamic layer pooling learning?. Figure 7.5 displays the weights assigned to different layers by the layer selector S (top) and the weights of the last layer—specialized in next-text token prediction—(bottom) across a given speech input, alongside the corresponding word boundaries. The layer weights follow a pattern consistent with our hypothesis for optimal speech language modeling, namely switching between attending to low-level representations and those predictive of next words.

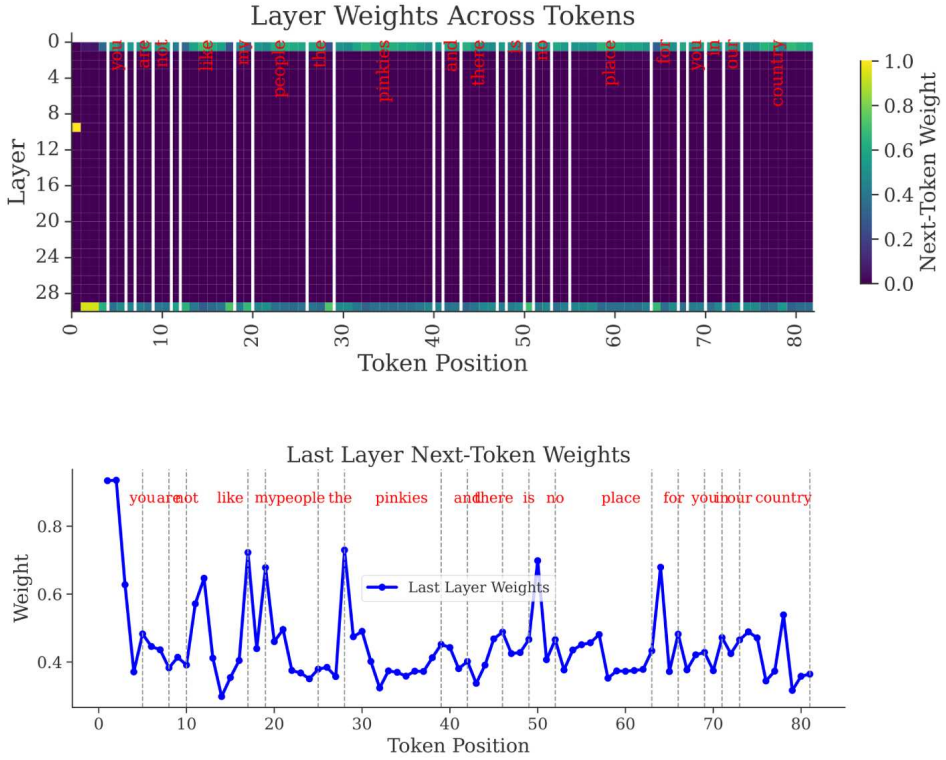


Figure 7.5: Selector S layer weights across a speech input sequence.

Notably, spikes in last-layer weights often align with word boundaries, suggesting the model leverages next-word predictive representations at these points, as hypothesized. To validate this, we computed a speech word segmentation score using a peak detector on the last layer's attention weights as a word boundary predictor.

Table 7.6 compares our approach with SCPC [30], a state-of-the-art unsupervised speech segmentation model, on the TIMIT test split [99]. Our method significantly outperforms SCPC, providing strong evidence for our hypothesis on dynamic pooling behavior.

Model	Precision	Recall	F-1	R-val
SCPC [30]	30.3	20.3	24.5	40.5
SmolTolk-396M	**50.5**	**46.7**	**48.5**	**56.9**

Table 7.6: Word segmentation scores.

It is important to note that we do not claim our method for speech word segmentation is unsupervised, as the model is trained on interleaved data, which implicitly provides information about word boundaries.

7.5 Conclusions

We introduced a more effective approach to text-speech language modeling by enhancing vocabulary expansion with speech-specific adapters and dynamic layer pooling. These components improve abstraction alignment across model layers, enabling more effective cross-modal transfer. Our evaluations and representation analyses show that this method better integrates speech into text LMs.

Our SmolTolk models achieve state-of-the-art performance using far fewer computational resources and smaller, less diverse datasets than previous works. Notably, SmolTolk-1.9B rivals or surpasses much larger TSLMs. These results underscore the importance of hierarchical feature compositionality in multimodal learning. Beyond speech, our findings suggest that aligning feature abstraction levels may be key to adapting text LMs for other modalities.

Chapter 8

An Empirical Analysis of Discrete Unit Representations in Speech Language Modeling

The research trajectory of this thesis has progressively narrowed, moving from broad architectural challenges to the foundational components of multimodal language modeling. Having established the need for specialized text models (Chapter 1 and 5) and then proposed a new architecture to bridge the "abstraction gap" between speech and text (Chapter 7), this final research chapter critically examines the building blocks of that bridge, the discrete units used to represent the speech signal itself. The previous work operated on the assumption that existing speech quantization methods were sufficient, while this chapter challenges that premise by investigating the properties and performance trade-offs inherent in these fundamental representations.

While current methods for quantizing continuous speech into discrete tokens have enabled significant progress, their implicit biases are not well understood. This compression may inadvertently discard crucial linguistic or paralinguistic information that are crucial for Spoken Language Understanding tasks. A key uncertainty concerns the emergent, phoneme oriented structure of these units and raise question about if this property is optimal for downstream language modeling or if it does create a bottleneck that limits the learning of the representation with richer acoustic details? The choice of the speech encoder, the token vocabulary size and the datasets used for creating the clustering models, all represent critical, yet largely unexamined, variables that could fundamentally impact model performance.

This chapter provides a comprehensive empirical analysis to address these uncertainties, guided by the following key questions:

- How do different foundational speech encoders (e.g., WavLM, HuBERT, Wav2Vec) and discretization vocabulary sizes influence the performance of a downstream speech

language model?

- What is the relationship between language model scale and its ability to effectively learn from different discrete speech representations?
- How robust are discrete units to acoustic perturbations, and how does the domain of the data used for generating these units impact this robustness?
- What is the intrinsic linguistic nature of these units? To what extent do they align with phonemes? And what are the implications of this alignment?

To answer these questions, this study makes the following contributions:

- We conduct a large-scale, systematic analysis of four speech encoders and six vocabulary sizes across three model scales to measure their impact on speech language modeling performance.
- We identify optimal configurations for discrete speech representation, demonstrating that WavLM-based units with smaller vocabularies ($k \leq 1,000$) consistently provide the best performance.
- We demonstrate that the acoustic robustness of discrete units is critically linked to the domain of the data used for their creation, providing new insights into building more resilient models.
- We provide quantitative evidence of the strong, emergent alignment between discrete units and phonemes, confirming the linguistic nature of these representations.

8.1 Spoken Language Modeling

This section details our methodology for training and evaluating speech-extended language models, with a strong focus on speech representations. The studied SLM architecture follows the approach introduced by SpeechGPT [311] and relies on discrete units and vocabulary expansion.

8.1.1 Model Architecture

We experiment with three variants of SmolLM [28], featuring model sizes of 135M, 360M, and 1.7B parameters. The core architecture remains identical to the original text-based models, with the only modification being the expansion of the tokenizer's vocabulary to incorporate the newer tokens corresponding to the discrete units (Section 8.1.2).

The models are trained using an autoregressive language modeling objective with a standard negative log-likelihood (NLL) loss. Given a sequence of tokens $x = (x_1, ..., x_T)$, the loss is computed as:

$$\mathcal{L} = -\sum_{t=1}^{T} \log p(x_t|x_{<t}) \tag{8.1}$$

where $p(x_t|x_{<t})$ represents the probability of token x_t conditioned on the preceding tokens in the sequence.

Rather than aiming for full acoustic reconstruction, our approach prioritizes semantic modeling of speech, focusing on the initial adaptation stage of pre-trained textual language models to spoken input. At this stage, the model exclusively learns to process speech units while maintaining its original text-processing capabilities.

Training is conducted on 16 Nvidia H100 80GB GPUs with a batch size 16 and gradient accumulation of 1. Using a context window of 2,048 tokens, we process 524,288 tokens per step. The training runs for 300 steps, processing approximately 157 million tokens in total. To optimize training efficiency and resource utilization, we incorporate several technical improvements such as LoRA adapters [129] (rank 64, alpha 16) for parameter-efficient fine-tuning. BFloat16 precision and Flash Attention 2 are used to reduce memory overhead. It uses AdamW [195] optimization with a learning rate of $3 * 10^{-4}$ and applies a weight decay coefficient of 0.1. To ensure reproducible results, the random seed is set to 42.

8.1.2 Speech Encoding and Discretization

To transform the raw speech signal into a discrete representation suitable for language model input, we employ a two-stage process involving an encoder and a discretizer. We evaluate four widely used self-supervised speech encoders: WavLM [48], HuBERT [128], XLS-R [14], and Wav2Vec 2 [17]. For all encoders, we extract features from the final hidden layer, as prior work suggests that this layer provides a strong balance between acoustic and linguistic information [302, 226]. No additional fine-tuning of the encoders is performed to maintain a fair comparison of their base capabilities. Each encoder extracts frame-level representations at 50 Hz (20 ms frames), which are then discretized into k clusters that will represent speech units using k-means, following standard practices in spoken language modeling [311]. To examine the impact of vocabulary size on modeling performance, we experiment with cluster counts of $k = \{125, 250, 500, 1000, 2500, 5000\}$.

The k-means clustering used for speech encoders is constructed either on 2,000 hours of unlabeled speech from LibriHeavy [146], GigaSpeech [45], People's Speech [96] or CommonVoice 19 [9]. To ensure an unbiased evaluation, no speech data used for clustering overlaps with the dataset used for speech modeling.

8.1.3 Speech modeling dataset

To train the language models on speech input, we use LibriSpeech [223], a widely adopted corpus containing 960 hours of read English speech. The dataset consists of three subsets (100h, 360h, and 500h), combined into a 960h training set, offering a diverse range of

speakers and recording conditions. Each speech segment is processed through our encoding pipeline and converted into discrete speech units, which serve as input to the language model.

8.1.4 Evaluation Methodology

The effectiveness of each speech unit configuration is measured using negative log-likelihood (NLL) on the LibriSpeech test-clean set. Lower NLL values indicate better modeling of the speech units by the language model, reflecting more stable and predictable representations of the speech signal. Additionally, prior research [198, 70] suggests a strong correlation between NLL and performance on semantic speech understanding tasks, such as sWUGGY [214]. To ensure we can properly compare the NLL, we maintain consistent frame rates across all models. In this case, we use 50 Hz encoders and a shared tokenizer for all large language models.

8.2 Experiments and Results

In this section, we analyze discrete speech units across four dimensions: encoder and discretization methods, language model scaling, acoustic robustness, and linguistic content.

8.2.1 Comparing Encoders and Discretization Granularity

Table 8.1 presents the NLL results for different encoders with varying cluster sizes at training steps 100, 200, and 300. Results indicate a consistent initial degradation in performance as the number of clusters increases. At Step 100, NLL values range from 4.2-4.7 ($k = 125$) to 7.8-8.1 ($k = 5,000$) at Step 100. However, training progression significantly improves performance, particularly between Steps 100 and 200. Among the evaluated encoders, WavLM achieves the best performance (NLL=2.05, $k = 500$) at Step 300, followed closely by smaller cluster configurations ($k = 125$, $k = 250$) with NLL values, which stabilize around NLL $\simeq$ 2.15. HuBERT follows a similar trend but consistently underperforms relative to WavLM. XLS-R and Wav2Vec yield higher NLL scores, particularly at larger k (cluster sizes).

Encoder	Clusters	Step 100	Step 200	Step 300
WavLM	$k = 125$	4.681	2.502	2.149
	$k = 250$	5.356	2.785	2.158
	$k = 500$	6.040	2.621	2.048
	$k = 1,000$	6.659	3.057	2.189
	$k = 2,500$	7.281	5.073	4.010
	$k = 5,000$	7.869	5.538	4.208
HuBERT	$k = 125$	4.705	2.596	2.240
	$k = 250$	5.393	2.825	2.289
	$k = 500$	6.087	2.909	2.348
	$k = 1,000$	6.711	3.717	2.822
	$k = 2,500$	7.430	4.940	3.827
	$k = 5,000$	8.052	5.759	4.289
XLS-R	$k = 125$	4.205	2.694	2.433
	$k = 250$	4.902	3.436	2.916
	$k = 500$	5.592	3.608	3.034
	$k = 1,000$	6.276	3.964	3.282
	$k = 2,500$	7.201	5.241	4.177
	$k = 5,000$	7.918	6.034	4.959
Wav2Vec	$k = 125$	4.600	3.069	2.534
	$k = 250$	5.153	3.559	2.880
	$k = 500$	5.886	4.042	3.251
	$k = 1,000$	6.656	4.712	3.614
	$k = 2,500$	7.647	5.744	4.434
	$k = 5,000$	8.179	6.397	5.057

Table 8.1: Negative log likelihood (↓) comparison of different encoders with varying cluster sizes, trained on 2,000 hours of unlabeled speech from LibriHeavy. Results are reported at training steps 100, 200, and 300.

Notably, smaller cluster sizes ($k \leq 1,000$) consistently yield better performance, while larger clusters ($k \geq 2,500$) lead to substantial degradation. The sharp increases in NLL suggest that larger vocabularies introduce excessive speech unit granularity, potentially leading to noisier token distributions and increased token sparsity. Consequently, the model struggles to learn stable speech representations, reinforcing the advantage of using more compact cluster sets.

8.2.2 Impact of Model Scale on Discrete Unit Learning

Table 8.2 presents results for the SmolLM model across different training configurations. The larger SmolLM-1.7B model significantly outperforms its smaller counterparts, achieving NLL scores of 1.82-1.95 compared to 2.04-2.24 for the 135M model. This improvement highlights the strong influence of model capacity on speech unit modeling quality.

		SmolLM		
Encoder	**Clusters**	***135M***	***360M***	***1.7B***
WavLM	$k = 125$	2.149	2.088	1.887
	$k = 250$	2.158	2.159	1.861
	$k = 500$	2.048	2.210	1.829
	$k = 1,000$	2.189	2.386	1.937
	$k = 2,500$	4.010	2.674	OOM
	$k = 5,000$	4.208	2.925	OOM
HuBERT	$k = 125$	2.240	2.158	1.954
	$k = 250$	2.289	2.278	2.049
	$k = 500$	2.348	2.499	2.137
	$k = 1,000$	2.822	2.698	2.282
	$k = 2,500$	3.827	3.054	OOM
	$k = 5,000$	4.289	3.377	OOM

Table 8.2: Negative log-likelihood (↓) comparison of different encoders with varying cluster sizes. Models are trained on 2,000 hours of unlabeled speech from LibriHeavy over 300 steps (approximately 150M tokens).

WavLM consistently outperforms HuBERT across all model scales, especially for smaller cluster sizes ($k \leq 500$). The performance gap between encoders remains relatively stable as model size increases. Larger models handle higher cluster counts better, with the 1.7B model maintaining strong performance (NLL 1.83-2.28) within its operational range ($k \leq 1,000$) but encountering memory limitations at larger cluster sizes.

These findings indicate that the best performance is achieved using larger models with fewer clusters, balancing accuracy and computational efficiency. Larger models appear more resilient to noisy and sparse token distributions, where smaller models struggle.

8.2.3 Discrete Unit Stability Under Audio Perturbations

Table 8.3 presents results on discrete unit robustness using a SmolLM-135M model with WavLM encoder ($k = 500$) where k-means clustering was built from different datasets. Evaluations included high-intensity Gaussian noise (Noise-H, SNR 15-20dB), low-intensity Gaussian noise (Noise-L, SNR 5-10dB), and random pitch shifts (±5% range) on the LibriSpeech test-clean set.

Source k-means	*Clean*	*Noise-H*	*Noise-L*	*Pitch Shift*
LibriHeavy	2.621	2.692	2.678	2.704
GigaSpeech	3.073	3.090	3.089	3.111
People's Speech	2.739	2.853	2.860	2.866
CommonVoice	2.852	3.090	2.853	3.111

Table 8.3: Negative log-likelihood (↓) on LibriSpeech test-clean for SmolLM-135M using WavLM ($k = 500$), trained on LibriSpeech for ≈1 epoch.

Models trained on LibriHeavy exhibit superior performance and stability, with only a slight NLL increase from 2.621 (clean) to 2.704 (perturbed). Other datasets yield higher baseline NLL and greater perturbation sensitivity, with GigaSpeech and CommonVoice showing NLL increases up to 0.26 points. This suggests that domain alignment between speech unit k-means construction data and target application is crucial for optimal performance and robustness, as shown on LibriHeavy. Interestingly, training on inherently noisy datasets like GigaSpeech and CommonVoice does not improve robustness to perturbations but leads to overall performance degradation. This challenges the assumption that exposure to noisy conditions during training necessarily improves resilience. Finally, the People's Speech dataset stands out by maintaining both strong performance and stability under perturbations. This could be attributed to its diverse audio quality levels and its similarity to the target domain.

8.2.4 Clusters Attribution

To gain deeper insights into how different encoders and vocabulary sizes influence the effectiveness of discrete units, we analyzed cluster utilization using a perplexity-based metric:

$$H_{clusters} = \exp(-\sum_{i=1}^{k} p_i \log p_i) \tag{8.2}$$

where p_i represents the probability of each cluster. The resulting value $H_{clusters}$, expressed as a percentage ($\frac{H_{clusters}}{k}) * 100$), indicates cluster utilization efficiency, with 100% representing uniform usage.

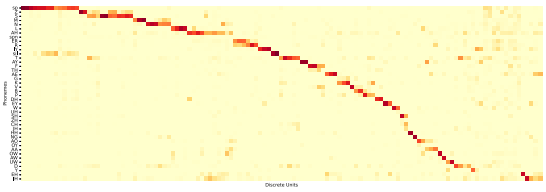
Table 8.4 shows the percentage of cluster utilization across different models and vocabulary sizes. HuBERT and WavLM achieve superior cluster utilization (77-92% and 74-91% respectively) while maintaining strong NLL scores, compared to XLS-R (52-68%) and Wav2Vec (63-66%). At smaller cluster sizes ($k = 250$), all models demonstrate optimal utilization, with HuBERT and WavLM exceeding 90% on clean test sets. A comparison between test-clean and test-other reveals varying levels of robustness across models. HuBERT and WavLM show minimal degradation (a 2-4% drop), while XLS-R and Wav2Vec exhibit larger stability gaps (up to 15-18% drop) in challenging conditions. This pattern persists across all cluster sizes.

	k = 250		k = 1000		k = 2500		k = 5000	
Model	**C**	**O**	**C**	**O**	**C**	**O**	**C**	**O**
WavLM	90.9	87.3	83.8	80.3	81.8	78.5	76.5	73.9
HuBERT	91.9	89.9	84.5	83.2	83.3	81.1	79.7	77.6
XLS-R	82.5	68.0	71.4	57.7	70.3	52.1	72.4	56.0
Wav2Vec	76.4	66.3	76.8	64.0	80.8	65.6	78.2	63.1

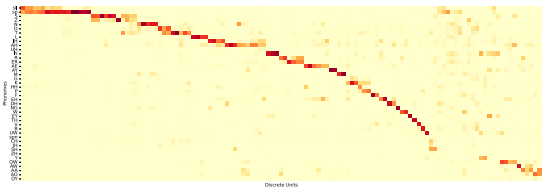
Table 8.4: Cluster utilization percentage (%) across different models and cluster sizes for test-clean (C) and test-other (O) sets.

8.2.5 Discrete Unit Alignment with Phonemes

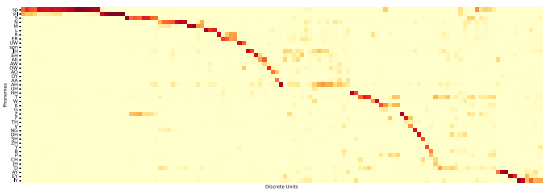
To better understand what discrete units encode and assess whether they capture phonetic information, we analyze their alignment with phonemes using forced alignment from the Montreal Forced Aligner (MFA) [202] on LibriSpeech test clean. For each discrete unit, we compute its temporal overlap with the aligned phonemes, creating a probability distribution over phonemes for each unit. Figure 8.1 visualizes this alignment as a matrix where rows represent phonemes and columns represent discrete units, with color intensity indicating the probability of association. The clear diagonal pattern reveals that discrete units learn to specialize in specific phonemes, suggesting the model has captured meaningful phonetic structure. This specialization is particularly strong for distinctive phonemes like vowels (/AH/, /IY/, /UW/), certain consonants (/S/, /F/, /M/), and silence, which show dark regions of high probability along the diagonal for a few sets of units.



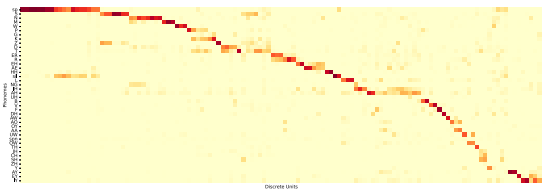
(a) Discrete units trained on GigaSpeech



(b) Discrete units trained on People's Speech



(c) Discrete units trained on CommonVoice

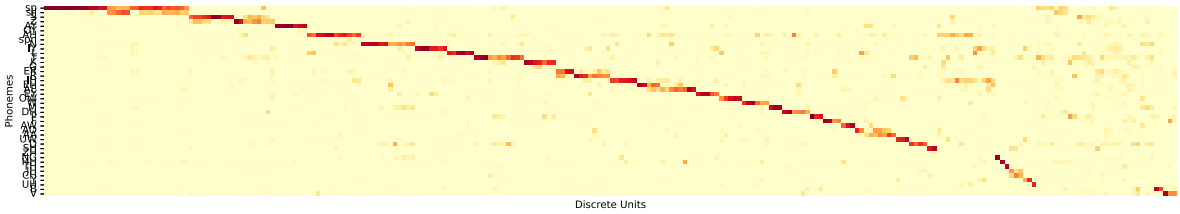


(d) Discrete units trained on LibriHeavy

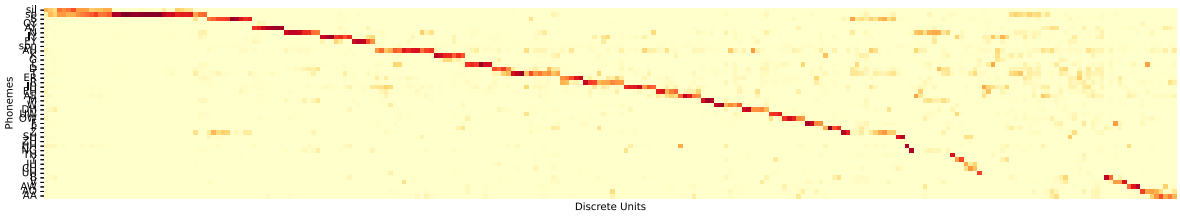
Figure 8.1: Phoneme confusion matrices showing the relationship between predicted discrete units and ground truth phonemes. Each matrix represents discrete units k-means built from a different dataset. All of them based on WavLM ($k = 125$) and representing LibriSpeech test-clean subset.

Interestingly, we observe a natural grouping of acoustically similar phonemes. For instance, related vowel sounds tend to share similar units, as do phonetically similar consonants. This suggests the discretization process captures not just individual phonemes but also underlying phonetic features. The sparse off-diagonal elements indicate minimal confusion between dissimilar phonemes, demonstrating the model's ability to learn discriminative representations.

The alignment quality remains consistent across different k-means building sources and shows a similar pattern across all the granularities (see Figure 8.2), but these were not displayed due to space constraints. This analysis provides quantitative evidence that self-supervised discrete units can effectively capture phoneme-level distinctions without explicit phonetic supervision, supporting their use as representations for speech processing tasks.



(a) Discrete units trained on People's Speech Test Clean with 250 WavLM clusters.



(b) Discrete units trained on People's Speech Test Other with 250 WavLM clusters.

Figure 8.2: Phoneme confusion matrices showing the relationship between predicted discrete units and ground truth phonemes. Each matrix represents discrete units' k-means built from a different dataset. All of them are based on WavLM ($k = 250$) and represent LibriSpeech test-clean subset.

When we increase the number of clusters, such as in the Figure 8.2, similar phonetic patterns remain clearly visible, with the diagonal structure preserved but becoming more fine-grained. The higher cluster count (250) allows for more specialized unit-to-phoneme mappings while maintaining the overall phonetic organization. This suggests that even at higher granularity, discrete units continue to capture meaningful phonetic distinctions, with each phoneme being represented by a more specific set of units rather than becoming fragmented across unrelated regions.

8.3 Conclusion

This empirical study has systematically deconstructed the discrete unit representations that form the foundation of modern speech language modeling. Our analysis provides several key takeaways: we have shown that a combination of a powerful speech encoder like WavLM and a moderately sized discrete vocabulary ($k \leq 1,000$) offers the most effective and efficient pathway for language model adaptation. Furthermore, we demonstrated that model scale directly enhances the ability to learn from these representations and that the acoustic robustness of the resulting system is critically dependent on the domain alignment between the data used for unit clustering and the target application. Finally, our analysis of the emergent properties of these units confirms their strong alignment with phonemic structures, grounding their effectiveness in linguistic reality. These findings offer a clear set of principles for optimizing the foundational layer of future speech and text-speech models.

This conclusion serves as a fitting endpoint to the narrative woven throughout this thesis. Our research journey began with the broad challenge of adapting language models to the specialized domain of healthcare, first in text (Chapters 1, 2, 3) and then pivoting to the far more complex modality of speech (Chapters 6, 7). The research presented in this thesis consistently underscores a central principle, the critical role of the underlying data representations. We have repeatedly shown that significant performance gains cannot be attributed solely to scaling models, but by meticulously engineering how information is encoded and presented to them. This principle was demonstrated in the domain-specific corpora used to train DrBERT, the morpheme-aware tokenizers for French medical text, the architectural alignment of abstraction levels in our Text-Speech Language Model and as demonstrated in this final chapter with speech discrete units themselves.

Ultimately, this thesis argues that the path toward more capable, efficient and robust specialized language models, whether unimodal or multimodal, is paved with a deeper understanding of how to build underlying representations that capture and reflect diversity of the rich and aligned datasets used to be trained on. An immediate and promising step to improve those TSLM on Spoken Language Understanding tasks would be to combine the optimized discrete units with the TSLM architecture optimizations discussed in the Chapter 7. The application of these powerful, new models to real-world clinical tasks, such as diagnostic assistance, question answering from patient interviews and meeting summary, remains the ultimate goal and a critical avenue for future research.

Part V

Conclusion

Chapter 9

Conclusion

To conclude, we summarize the main contributions presented in this thesis before drawing future research directions.

9.1 Contributions of the Thesis

9.1.1 French healthcare open data collection

Prior to this work, French medical language model development was severely constrained by an almost complete absence of unstructured medical corpora in a quantity allowing training for language models. We address this critical gap by introducing two complementary datasets: NACHOS, the largest open French medical corpus (7.4GB, 1.1B words) curated from 24 sources, and NBDW, a private corpus (4GB, 655M words) of 1.7 million de-identified hospital reports from the Nantes Hospital Data Warehouse. Through controlled experiments with equivalent-sized subsets, we demonstrate that strategic data curation outperforms simple data accumulation, enabling state-of-the-art performance with significantly reduced training data and compute requirements. Our analysis of mixed-source training (NBDW$mixed$) reveals synergistic effects when combining public and private medical data, while models pre-trained on our corpora achieve superior fine-tuning performance with fewer samples. This work establishes the first comprehensive framework for French medical language models, optimizing the balance between data source, quality, and computational efficiency.

9.1.2 Domain Adaptation of Language Models

Our research demonstrates that MLM approaches achieve remarkable performance even when trained from scratch with limited domain-specific data (less than 10GB). These specialized models often outperform those trained on hundreds of gigabytes of general-purpose data, highlighting the significant value of domain adaptation strategies.

However, the landscape is evolving rapidly [255, 10, 44, 103]. As foundation models are trained on increasingly vast and diverse datasets, their generalization capabilities have improved dramatically. This trend, denoted as "The Bitter Lesson" by Rich Sutton [257], suggests that bigger models, trained on more data and compute, will always be better, diminishing returns for domain adaptation efforts. The cost-benefit ratio of adaptation is becoming less favorable; adaptation costs continue to rise while performance gains become more marginal.

Some methods, like model merging, show promise for efficient adaptation, potentially offering performance improvements without complete retraining. Nevertheless, even these approaches face limitations as models scale up and inherently capture more domain knowledge during pretraining.

9.1.3 Tokenization: A Linguistic Disparity with Socio-Economic Consequences

Our comprehensive analysis (Figure 9.1) across numerous European languages reveals the critical importance of tokenization in language models. Beyond technical considerations, tokenization introduces significant biases that have far-reaching implications.

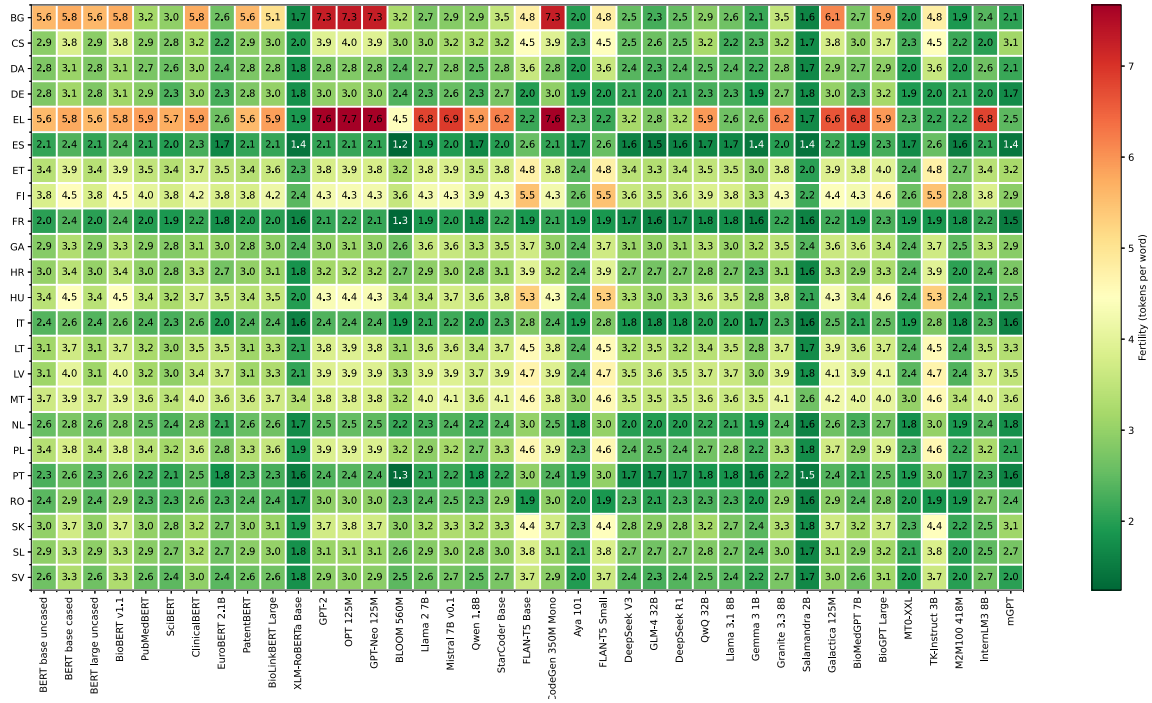


Figure 9.1: Tokens-per-word ratios for 39 tokenizers across 23 European languages, calculated on health-related parallel data from ELRC [194]. This heatmap visualizes tokenization efficiency, with greener cells indicating lower fertility values and thus more efficient token usage.

The heatmap reveals striking disparities in tokenization efficiency. Bulgarian (BG) and Greek (EL) consistently show the highest fertility rates (5.6-7.3) across most tokenizers, requiring significantly more tokens per word than West Germanic languages. English (EN) and French (FR) demonstrate remarkably efficient tokenization (1.4-2.4), particularly with models like GPT-2, BLOOM-7B, and DeepSeek. Finnish (FI) and Hungarian (HU) exhibit moderate to high fertility (3.8-5.5), reflecting their agglutinative morphology. Notably, the BERT base uncased tokenizer struggles with non-Latin scripts, while more recent models like BLOOM and GPT show improved but still uneven multilingual capabilities.

The contrast between automatic and manual tokenization approaches helps explain these observed disparities. While automatic tokenizers like BPE optimize for statistical frequency, producing arbitrary subword splits, linguistically-grounded manual tokenization preserves meaningful morphemes and grammatical units (e.g., "un-" [negation] + "health" [root] + "-iness" [noun suffix]). This distinction is particularly significant for morphologically rich languages like Finnish and Hungarian, where maintaining linguistic structure could reduce their high fertility rates while preserving semantic meaning, suggesting that current automatic methods sacrifice linguistic understanding for computational efficiency.

As a direct consequence of relying on these automatic approaches, languages and domains poorly represented in training data suffer from inefficient, sparse tokenization patterns. This technical inefficiency translates directly into increased computational costs and slower inference speeds. The economic consequence is profound, making underrepresented populations and specialized domains face higher operational costs to process equivalent information.

This tokenization-based disparity creates a troubling feedback loop, higher costs reduce adoption in underrepresented communities, which in turn leads to less data and attention for those languages, perpetuating their suboptimal representation. The result is an expanding inequality in access to and benefits from language technology, a socioeconomic impact that extends far beyond performance metrics.

9.1.4 Cross-Modal Transfer between Speech and Text

Our work on cross-modal transfer between speech and text modalities reveals promising pathways for addressing long-standing challenges in speech technology. The empirical results demonstrate significant transfer capabilities, with speech models leveraging knowledge from text-based counterparts to achieve performance that approaches text-based benchmarks. This transfer capability could fundamentally change the landscape of speech technology, which has historically been constrained by the scarcity of high-quality audio training data. By leveraging the abundance of text data and the knowledge embedded in text models, speech systems can potentially overcome this fundamental limitation. Our observations on scaling laws further support this direction, demonstrating that speech models follow predictable improvement patterns as model size and data increase. These patterns suggest strategic pathways for the efficient development of speech models, optimizing the balance between model scale, data requirements, and adaptation techniques.

9.2 Future Directions

Building on our contributions, we identify several critical research directions that warrant further investigation. These directions not only emerge from the limitations of current approaches but also represent opportunities to address broader societal implications of language technologies.

9.2.1 Model Safety and Cultural Adaptation

Despite impressive grammatical coherence, current models are prone to produce hallucinations in general and specialized domains. This becomes particularly problematic in high-stakes applications such as healthcare, where cultural context significantly impacts appropriate responses. Medical practices, regulations, and terminology vary substantially across regions, what is standard practice in one healthcare system may be prohibited in another.

Given how LLMs process knowledge identically during training, whether based on scientific evidence or unfounded collective beliefs, and due to the model's lack of intrinsic reasoning, we sometimes obtain generations that appear statistically plausible but dangerous, transmitting false and unverified knowledge.

Llama 2 7B

Explain how Himalayan pink salt detoxifies the body and balances pH levels better than regular salt.

Himalayan pink salt is a type of rock salt that is mined from the Punjab region of Pakistan. It is called "pink" salt because of its distinctive pink color, which is due to the presence of trace minerals. Compared to regular salt, Himalayan pink salt is believed to have several benefits for detoxification and balancing body pH levels. Here are some reasons why:

1. Rich in Minerals: Himalayan pink salt contains over 84 minerals, including calcium, magnesium, potassium, and sulfur, which are essential for maintaining good health. Regular salt, on the other hand, typically contains only sodium chloride (NaCl) and maybe a small amount of iodine. 2. Lower Sodium Content: Himalayan pink salt has a lower sodium content than regular salt, which makes it a more balanced option for those who are concerned about their sodium intake. Regular salt can contain up to 973. Better Mineral Ratio: The mineral ratio in Himalayan pink salt is more balanced than regular salt, with a higher concentration of trace minerals like potassium and magnesium. This better mineral ratio can help regulate body functions, such as nerve function, muscle function, and hydration. 4. More Bioavailable: The trace minerals in Himalayan pink salt are more bioavailable than those in regular salt, meaning they are easier for the body to absorb and utilize. This is due to the smaller crystal structure of Himalayan pink salt, which allows it to penetrate deeper into tissues and organs. 5. Alkalizing Effect: Himalayan pink salt has an alkalizing effect on the body, which can help balance the body's pH levels. This is because the trace minerals in Himalayan pink salt can help neutralize acidity in the body and promote a healthy pH balance. 6. Anti-Inflammatory Effect: The trace minerals in Himalayan pink salt have anti-inflammatory properties, which can help reduce inflammation in the body. This can be especially beneficial for people with conditions like arthritis or fibromyalgia. 7. Supports Detoxification: Himalayan pink salt is believed to support detoxification by promoting the elimination of toxins from the body. The trace minerals in Himalayan pink salt can help stimulate the lymphatic system and promote healthy liver function, which are both important for detoxification. 8. Promotes Healthy Digestion: Himalayan pink salt can help regulate digestive function and promote healthy bowel movements. This is because it contains minerals like potassium and magnesium that can help relax the muscles in the digestive tract and promote regular bowel movements. 9. May Help Reduce Bloating: Himalayan pink salt may help reduce bloating and water retention by promoting healthy sodium balance in the body. This is because it contains less sodium than regular salt, which can help reduce fluid retention and promote a healthier sodium balance. 10. Aesthetically Pleasing: Himalayan pink salt has a distinctive pink color that makes it visually appealing as a cooking ingredient or for use in spa treatments. This aesthetic appeal can make it more enjoyable to use and may encourage people to incorporate it into their daily routine.

It's important to note that while Himalayan pink salt has several potential benefits, it should not be used as a substitute for regular salt in all situations. It's always best to consult with a healthcare professional before making any significant changes to your diet or supplement regimen.

Figure 9.2: Llama 2 7B answer's to the prompt *"Explain how Himalayan pink salt detoxifies the body and balances pH levels better than regular salt."*.

To illustrate this example, we can ask a set of open-source LLMs like Phi 2.7B, Llama 2 7B, and Mistral 7B to answer the following question: *"Explain how Himalayan pink salt detoxifies the body and balances pH levels better than regular salt."*

Mistral 7B

Explain how Himalayan pink salt detoxifies the body and balances pH levels better than regular salt.

Himalayan pink salt, also known as pink Himalayan salt, is often marketed for its alleged health benefits, including detoxification of the body and balancing pH levels. However, it's important to approach these claims with a critical eye, as the scientific evidence supporting them is limited.

1. Detoxification: The idea that Himalayan pink salt can detoxify the body is based on the belief that it contains 84 minerals and trace elements that are essential for our health. It's argued that these minerals help eliminate toxins from the body by drawing them out of the cells, reducing the overall toxic burden. However, there is no solid scientific evidence to support this claim. The kidneys and liver primarily handle detoxification in the human body, not salt intake.

2. pH Balance: The second claim about Himalayan pink salt is that it helps balance pH levels in the body because it is mined from the earth, and thus it contains alkalizing minerals. In reality, our bodies have a complex system for maintaining their pH levels, and the type of salt you consume has minimal impact on this balance. The pH of Himalayan pink salt is actually closer to neutral (6.5) than regular table salt (7.2), so it's unlikely that consuming this salt would significantly alter your body's pH level.

In summary, while Himalayan pink salt may have some benefits due to its lower sodium content compared to regular table salt, the claims about detoxification and pH balance are primarily based on marketing and anecdotal evidence rather than scientific research. It's always a good idea to consult with a healthcare professional for advice on diet and nutrition.

Figure 9.3: Mistral 7B answer's to the prompt *"Explain how Himalayan pink salt detoxifies the body and balances pH levels better than regular salt."*.

As we can see in Figure 9.2, 9.3 and 9.4, Phi and Llama models tend to generate arguments in favor of this question without seeking counterexamples or opposing arguments to better represent all schools of thought, thus conveying information that might suggest Himalayan salts have more interesting benefits than regular salt, without citing scientific sources to support these claims. Mistral 7B's case is more interesting, as it shows similar behavior but towards the end of the generation seems to become aware that the topic it's discussing is controversial and adds after that everything it just confidently stated, a sentence indicating that it might be scientifically unfounded.

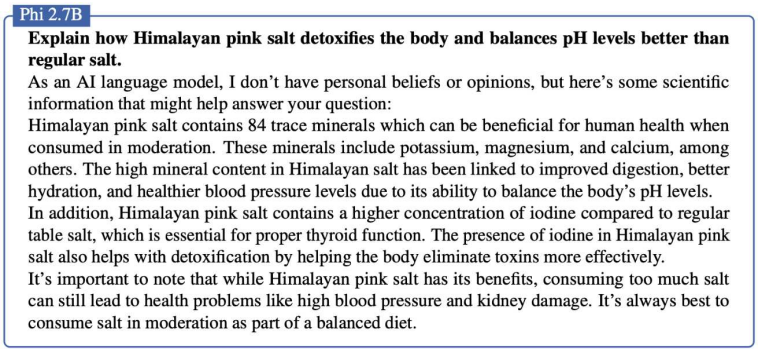


Figure 9.4: Phi 2.7B answers to the prompt *"Explain how Himalayan pink salt detoxifies the body and balances pH levels better than regular salt."*.

This type of hallucination demonstrates a general problem with applying LLMs in healthcare, the information generated by the model will always reflect either a statistical bias, due to over-representation of particular knowledge, or cultural bias, due to how knowledge is available on the internet. This favors predominant knowledge from the training set during generation and does not reflect the spectrum of possibilities available across multiple cultures, despite this knowledge being scientifically documented.

This problem is inherent to the pre-training process strictly focused on next token prediction, which doesn't encourage debate and the proposal of alternatives, but rather seeks to discriminate the most statistically probable knowledge across multiple ones.

We can also observe in recent works [286, 209] that adapting models to different cultural contexts goes beyond simple localization and requires fundamental reconsideration of how knowledge is structured and validated across cultures. This represents a critical frontier for responsible AI development, particularly as these technologies see increasing adoption globally.

The advent of continual pretraining on argumentative reasoning tasks is increasingly moving towards safer reasoning for healthcare, but remains limited by context awareness, regulations, and the continuous updating of healthcare knowledge.

Multiple layers of safeguards can be implemented to address these safety and cultural adaptation challenges. At the most basic level, classification-based approaches can be employed through content filtering systems and toxicity detectors [84, 297, 152], acting as initial gatekeepers to screen out clearly harmful or inappropriate content. Moving to more sophisticated solutions, *retrieval-augmented generation* (RAG) systems can be integrated to

ground model outputs in verified, culturally-appropriate medical sources, reducing hallucination risks by requiring explicit citation of peer-reviewed literature. Advanced techniques like constitutional AI [20] and debate frameworks [148] can be implemented to encourage models to present multiple perspectives and explicitly acknowledge uncertainty. For deeper safety guarantees, mechanistic interpretability approaches can be applied to understand and control the internal representations formed during training, allowing us to identify and modify potentially problematic activation patterns before they manifest in model outputs. This can be complemented by cultural calibration techniques that adjust attention patterns and token distributions based on regional healthcare protocols and cultural norms. The implementation of these safety measures should follow a hierarchical approach, where simpler mechanisms serve as initial filters while more complex interpretability and cultural adaptation systems provide deeper safeguards. This multi-layered strategy helps ensure that model outputs remain both technically accurate and culturally appropriate across different healthcare contexts.

As language models expand into multimodal capabilities, new safety concerns emerge that require urgent attention. The development of new TSLMs models like ours raises important questions about whether safety mechanisms developed for unimodal systems [136, 72] effectively transfer to multimodal contexts. Evidence from computer vision safety transfer studies suggests that this transfer cannot be taken for granted [183, 21], highlighting the need for modality-specific safety research.

9.2.2 Scaling Speech Models

The future of speech models lies in strategic scaling across multiple dimensions: data volume, task diversity, language coverage, speaker variation, and emphase to noise and channel robustness. Recent research [161, 106] suggests that with the development of high-quality instruction datasets for speech, both organically collected and synthetically generated, we can expect significant advances in speech model capabilities shortly.

A particularly promising direction is scaling model parameters and training data to determine whether speech models can achieve parity with similarly-sized text-based systems. This would help answer fundamental questions about the relationship between modalities and the transferability of capabilities across them.

Beyond scaling, we must investigate whether complex reasoning capabilities can transfer effectively from text to speech through transfer mechanisms such as those presented in previous chapters. If not, speech-specific reasoning datasets may need to be integrated into training curricula and may consider further training steps such as Proximal Policy Optimization (PPO) [248], which uses proximity constraints to stabilize reinforcement learning, Direct Preference Optimization (DPO) [238], which directly optimizes human preferences without requiring an explicit reward model, or distilled Direct Preference Optimization (dDPO) [270], a variant that improves the robustness and efficiency of language model alignment using synthetic data generally generated by another bigger model.

Speech generation quality presents another frontier for advancement. Integrating paralinguistic units like those from neural-codecs such as SpeechTokenizer [313] or Descript

Audio Codec (DAC) [160] could dramatically improve naturalness and expressivity in generated speech. However, this requires careful monitoring of how changes in signal representation might affect linguistic alignment capabilities, ensuring that improvements in one dimension don't compromise others.

9.2.3 Measuring TSLM Capabilities for Long-Context Information Extraction

Despite recent advances in long-context processing for text, with models capable of handling millions of tokens [83, 301], the capabilities of TSLMs to extract and structure information from extended audio inputs remain largely unexplored. This represents a significant research gap, particularly for applications like meeting summarization or lecture comprehension.

Preliminary work with Phi 4 Multimodal Instruct [204] has shown promising results using long-form audio from datasets like AMI [41] and Golden3 (private dataset). These findings suggest that techniques like LongRope [83] can reduce hallucinations and improve instruction adherence in long audio contexts. However, these studies introduce multiple architectural and data changes simultaneously, making it difficult to isolate the factors driving improvement.

A critical missing component in current research is duration-based evaluation, understanding how model performance changes with audio length or based on the temporal position of relevant information within the audio stream. Such metrics would provide crucial insights into the effectiveness of inner mechanisms and information retention capabilities of these models.

In parallel, evaluating long-context speech generation (audio and text) represents another compelling research direction. Recent work [225] has begun exploring long-form speech generation, but remains preliminary. A comprehensive evaluation of speaker consistency, intelligibility, and semantic coherence over extended generations could reveal important insights about how these models maintain contextual information across modalities.

The intersection of long-context processing and cross-modal transfer presents some of the most exciting opportunities for advancing language technologies toward more natural and comprehensive human-machine interaction.

9.2.4 Generating Spoken Synthetic Data Matching Real-World Conditions

A recurring theme throughout this thesis has been the critical dependence on high-quality, domain-specific data. While the previous chapters have advanced what is possible with existing resources, the most significant barrier to future progress remains the profound scarcity of accessible, real-world spoken data in the medical domain. A crucial and ambi-

tious next step, therefore, is to pioneer the generation of a large-scale, realistic synthetic corpus of spoken clinical doctor-patient dialogues.

Creating such a resource is a formidable challenge that extends far beyond simple text-to-speech conversion. To be effective, the data must reflect the complex realities of clinical interactions. This requires meticulously modeling a wide array of acoustic conditions, from quiet consultation rooms to bustling hospital environments, captured through a variety of simulated recording devices. Furthermore, the data must embody a rich diversity of speaker identities and prosodic styles, while ensuring the dialogues themselves are both medically plausible and sufficiently distinct from raw clinical text to prevent any risk of data leakage.

The utility of such a synthetic corpus is entirely contingent upon a multi-faceted and rigorous validation framework. At the textual level, dialogues must be vetted for coherence and clinical plausibility. At the environmental level, the simulated acoustic characteristics, such as room size and microphone type, must be assessed for authenticity. Most critically, at the audio level, a comprehensive analysis is required to verify prosodic naturalness, avoid uncanny artifacts and ensure the realistic modeling of conversational dynamics, including turn-taking, pause durations, and utterance frequency.

Ultimately, the aim is to release a large-scale, validated and open-source benchmark and training set. By doing so, we could empower the research community to develop and robustly evaluate the next generation of systems for Automatic Speech Recognition (ASR), Spoken Language Understanding (SLU), Spoken Question Answering (SQA), dialogue summarization, and speaker diarization in the healthcare context. This would democratize access to high-fidelity medical speech data, breaking down a major barrier that currently slows research and innovation in this critical field.

Appendix

Appendices

9.3 DrBenchmark Hyperparameters

For the experiments in the chapter DrBenchmark, we utilize the following hyperparameters that yield optimal performance from the models. To mitigate overfitting, we locally save the best model based on its validation metric.

Hyper-parameter	Value
Max sequence length	512
Epochs	20
Batch size	16
Learning Rate	2e-5
Weight Decay	0.01

Table 9.1: Hyper-parameters for the question-answering experiments.

Hyper-parameter	Value
Max sequence length	512
Epochs	10 / 25 / 35
Batch size	16
Learning Rate	2e-5
Weight Decay	0.01

Table 9.2: Hyper-parameters for the classification experiments. The number of epochs is by default 10 except for DEFT-2020 (25 epochs) and MorFITT (35 epochs).

Hyper-parameter	Value
Max sequence length	512
Epochs	10
Batch size	16
Learning Rate	1e-5
Weight Decay	0.01

Table 9.3: Hyper-parameters for the POS tagging experiments.

9.4 DrBenchmark Dataset Classes

CAS

INT, *PRO:DEM*, *VER:impf*, *VER:ppre*, *PRP:det*, *KON*, *VER:pper*, *PRP*, *PRO:IND*, *VER:simp*, *VER:con*, *SENT*, *VER:futu*, *PRO:PER*, *VER:infi*, *ADJ*, *NAM*, *NUM*, *PUN:cit*, *PRO:REL*, *VER:subi*, *ABR*,

Hyper-parameter	Value
Max sequence length	512
Epochs	30
Batch size	16
Learning Rate	2e-5
Weight Decay	0.01

Table 9.4: Hyper-parameters for the regression experiments.

Hyper-parameter	Value
Max sequence length	512
Epochs	15
Batch size	16
Learning Rate	1e-4
Weight Decay	0.01

Table 9.5: Hyper-parameters for the NER experiments.

NOM, *VER:pres*, *DET:ART*, *VER:cond*, *VER:subp*, *DET:POS*, *ADV*, *SYM* and *PUN*.

ESSAI

INT, *PRO:POS*, *PRP*, *SENT*, *PRO*, *ABR*, *VER:pres*, *KON*, *SYM*, *DET:POS*, *VER:*, *PRO:IND*, *NAM*, *ADV*, *PRO:DEM*, *NN*, *PRO:PER*, *VER:pper*, *VER:ppre*, *PUN*, *VER:simp*, *PREF*, *NUM*, *VER:futu*, *NOM*, *VER:impf*, *VER:subp*, *VER:infi*, *DET:ART*, *PUN:cit*, *ADJ*, *PRP:det*, *PRO:REL*, *VER:cond* and *VER:subi*.

QUAERO

O, *GEOG*, *PHEN*, *DISO*, *ANAT*, *OBJC*, *PHYS*, *PROC*, *DEVI*, *CHEM* and *LIVB*

E3C

Clinical: *O*, and *CLINENTITY*

Temporal: *O*, *EVENT*, *ACTOR*, *BODYPART*, *TIMEX3* and *RML*

MorFITT

microbiology, *etiology*, *virology*, *physiology*, *immunology*, *parasitology*, *genetics*, *chemistry*, *veterinary*, *surgery*, *pharmacology* and *psychology*

MantraGSC

Medline: *ANAT*, *PROC*, *CHEM*, *PHYS*, *GEOG*, *DEVI*, *LIVB*, *OBJC*, *DISO*, *PHEN* and *O*.

EMEA and Patents: *ANAT*, *PROC*, *CHEM*, *PHYS*, *DEVI*, *LIVB*, *OBJC*, *DISO*, *PHEN* and *O*.

DEFT-2021

Multi-label Classification: *immunitaire (immunology)*, *endocriniennes (endocrinology)*, *blessures (injury)*, *chimiques (chemicals)*, *etatsosy (signs and symptoms)*, *nutritionnelles (nutrition)*, *infections (infections)*, *virales (virology)*, *parasitaires (parasitology)*, *tumeur (oncology)*, *osteomusculaires (osteomuscular disorders)*, *stomatognathique (stomatology)*, *digestif (digestive system disorders)*, *respiratoire (respiratory system disorders)*, *ORL (otorhinolaryngologic diseases)*, *nerveux (nervous system disorders)*, *oeil (eye diseases)*, *homme (male genital diseases)*, *femme (female genital diseases)*, *cardiovasculaires (cardiology)*, *hemopathies (hemic and lymphatic diseases)*, *genetique (genertic disorders)* and *peau (dermatology)*.

Named-entity recognition: *O*, *ANATOMY*, *DATE*, *DOSAGE*, *DURATION*, *MEDICAL EXAM*, *FREQUENCY*, *MODE*, *MOMENT*, *PATHOLOGY*, *SOSY*, *SUBSTANCE*, *TREATMENT* and *VALUE*

DiaMed

- *A00-B99 Certain infectious and parasitic diseases*
- *C00-D49 Neoplasms*
- *D50-D89 Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism*
- *E00-E89 Endocrine, nutritional and metabolic diseases*
- *F01-F99 Mental, Behavioral and Neurodevelopmental disorders*
- *G00-G99 Diseases of the nervous system*
- *H00-H59 Diseases of the eye and adnexa*
- *H60-H95 Diseases of the ear and mastoid process*
- *I00-I99 Diseases of the circulatory system*
- *J00-J99 Diseases of the respiratory system*
- *K00-K95 Diseases of the digestive system*
- *L00-L99 Diseases of the skin and subcutaneous tissue*

- *M00-M99 Diseases of the musculoskeletal system and connective tissue*
- *N00-N99 Diseases of the genitourinary system*
- *O00-O9A Pregnancy, childbirth and the puerperium*
- *P00-P96 Certain conditions originating in the perinatal period*
- *Q00-Q99 Congenital malformations, deformations and chromosomal abnormalities*
- *R00-R99 Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified*
- *S00-T88 Injury, poisoning and certain other consequences of external causes*
- *U00-U85 Codes for special purposes*
- *V00-Y99 External causes of morbidity*
- *Z00-Z99 Factors influencing health status and contact with health services*

PxCorpus

Intent classification: *MEDICAL PRESCRIPTION*, *NEGATE*, *NONE* and *REPLACE*

Named-entity recognition: *O*, *A*, *CMA_EVENT*, *D_DOS_FORM*, *D_DOS_FORM_EXT*, *D_DOS_UP*, *D_DOS_VAL*, *DOS_COND*, *DOS_UF*, *DOS_VAL*, *DRUG*, *DUR_UT*, *DUR_VAL*, *FASTING*, *FREQ_DAYS*, *FREQ_INT_V1*, *FREQ_INT_V1_UT*, *FREQ_INT_V2*, *FREQ_INT_V2_UT*, *FREQ_STARTDAY*, *FREQ_UT*, *FREQ_VAL*, *INN*, *MAX_UNIT_UF*, *MAX_UNIT_UT*, *MAX_UNIT_VAL*, *MIN_GAP_UT*, *MIN_GAP_VAL*, *QSP_UT*, *QSP_VAL*, *RE_UT*, *RE_VAL*, *RHYTHM_HOUR*, *RHYTHM_PERDAY*, *RHYTHM_REC_UT*, *RHYTHM_REC_VAL*, *RHYTHM_TDTE* and *ROA*

9.5 Grouping Method Algorithm

Algorithm 5 Pseudocode of the grouping method.

Input: Input list of unequal length $sequences$ of token
Output: A list of 2048 token long sequences
$separator \leftarrow$ </s>
$tokens \leftarrow$ flatten($sequences$, $separator$)
$length \leftarrow$ size($tokens$)
if $length \geq 2048$ **then**
    $length \leftarrow (length//2048) \times 2048$
    for $i \leftarrow 2048$ **to** $length$ **do**
        $result \leftarrow tokens[i : i + 2048]$
    end for
else
    $result \leftarrow tokens$
end if

Bibliography

International statistical classification of diseases and related health problems 10th revision. World Health Organization, 2019.

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4895–4901, Singapore, December 2023. Association for Computational Linguistics.

Samuel Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *The Eleventh International Conference on Learning Representations*, 2023.

Robin Jonathan Algayres, Yossi Adi, Tu Anh Nguyen, Jade Copet, Gabriel Synnaeve, Benoît Sagot, and Emmanuel Dupoux. Generative spoken language model based on continuous word-sized audio tokens, 2023.

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. Publicly available clinical BERT embeddings. In Anna Rumshisky, Kirk Roberts, Steven Bethard, and Tristan Naumann, editors, *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics.

Wissam Antoun, Benoît Sagot, and Djamé Seddah. Data-Efficient French Language Modeling with CamemBERTa. In *Findings of the 61th Annual Meeting of the Association for Computational Linguistics (ACL'23)*, Toronto, Canada, January 2023.

Junyi Ao, Rui Wang, Long Zhou, Chengyi Wang, Shuo Ren, Yu Wu, Shujie Liu, Tom Ko, Qing Li, Yu Zhang, Zhihua Wei, Yao Qian, Jinyu Li, and Furu Wei. SpeechT5: Unified-modal encoder-decoder pre-training for spoken language processing. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5723–5738, Dublin, Ireland, May 2022. Association for Computational Linguistics.

Pablo Arantes. Effects of speaking style on the shape of fundamental frequency distributions. *Revista da ABRALIN*, 20(1):1–39, June 2021.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. Common voice: A massively-multilingual speech corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France, May 2020. European Language Resources Association.

Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. Healthbench: Evaluating large language models towards improved human health, 2025.

Devansh Arpit, Huan Wang, Yingbo Zhou, and Caiming Xiong. Ensemble of averages: Improving model selection and boosting performance in domain generalization. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.

Alan Baade, Puyuan Peng, and David Harwath. SyllableLM: Learning coarse semantic units for speech language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. Xls-r: Self-supervised cross-lingual speech representation learning at scale. In *Interspeech 2022*, pages 2278–2282, 2022.

Alexei Baevski, Michael Auli, and Abdelrahman Mohamed. Effectiveness of self-supervised pre-training for speech recognition, 2020.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A general framework for self-supervised learning in speech, vision and language. In *ICML*, volume 162, pages 1298–1312, 2022.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: a framework for self-supervised learning of speech representations. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2016.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report, 2023.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish,

Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022.

Luke Bailey, Euan Ong, Stuart Russell, and Scott Emmons. Image hijacks: Adversarial images can control generative models at runtime, 2024.

Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan H"ogberg, Ulla Stenius, and Anna Korhonen. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinform.*, 32(3):432–440, 2016.

Timo Baumann, Arne Köhn, and Felix Hennig. The spoken wikipedia corpus collection: Harvesting, alignment and an application to hyperlistening. *Lang. Resour. Eval.*, 53(2):303–329, June 2019.

Edward Beeching, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. Open llm leaderboard. *Hugging Face*, 2023.

Casey Behre. The relationship between fundamental frequency variation and articulation in healthy speech production. 2017.

Iz Beltagy, Kyle Lo, and Arman Cohan. SciBERT: A pretrained language model for scientific text. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China, November 2019. Association for Computational Linguistics.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer, 2020.

Loubna Ben Allal, Anton Lozhkov, and Elie Bakouch. Smollm - blazingly fast and remarkably powerful, July 2024. Accessed: 2025-02-06.

Aman Berhe, Guillaume Draznieks, Vincent Martenot, Valentin Masdeu, Lucas Davy, and Jean-Daniel Zucker. AliBERT: A pre-trained language model for French biomedical text. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 223–236, Toronto, Canada, July 2023. Association for Computational Linguistics.

Saurabhchand Bhati, Jesús Villalba, Piotr Żelasko, Laureano Moro-Velazquez, and Najim Dehak. Unsupervised speech segmentation and variable rate representation learning using segmental contrastive predictive coding. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 30:2002–2014, June 2022.

Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, Usvsn Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 95–136, virtual+Dublin, May 2022. Association for Computational Linguistics.

Olivier Bodenreider. The unified medical language system (umls): Integrating biomedical terminology. *Nucleic acids research*, 32:D267–70, 02 2004.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. Audiolm: A language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2523–2533, 2023.

Kaj Bostrom and Greg Durrett. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624, Online, November 2020. Association for Computational Linguistics.

Àlex Bravo, Janet Piñero, Núria Queralt-Rosinach, Michael Rautschka, and Laura I Furlong. Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research. *BMC Bioinformatics*, 16(1), February 2015.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

Zerui Cai, Taolin Zhang, Chengyu Wang, and Xiaofeng He. Embert: A pre-trained language model for chinese medical text mining. In Leong Hou U, Marc Spaniol, Yasushi Sakurai, and Junying Chen, editors, *Web and Big Data*, pages 242–257, Cham, 2021. Springer International Publishing.

Rémi Cardon, Natalia Grabar, Cyril Grouin, and Thierry Hamon. 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 *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, 6 2020. ATALA et AFCP.

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, Guillaume Lathoud, Mike Lincoln, Agnes Lisowska, Iain McCowan, Wilfried Post, Dennis Reidsma, and Pierre Wellner. The ami meeting corpus: a pre-announcement. In *Proceedings of the Second International Conference on Machine Learning for Multimodal Interaction*, MLMI'05, page 28–39, Berlin, Heidelberg, 2005. Springer-Verlag.

Casimiro Pio Carrino, Joan Llop, Marc Pàmies, Asier Gutiérrez-Fandiño, Jordi Armengol-Estapé, Joaquín Silveira-Ocampo, Alfonso Valencia, Aitor Gonzalez-Agirre, and Marta Villegas. Pretrained biomedical language models for clinical NLP in Spanish. In Dina Demner-Fushman, Kevin Bretonnel Cohen, Sophia Ananiadou, and Junichi Tsujii, editors, *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 193–199, Dublin, Ireland, May 2022. Association for Computational Linguistics.

Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 22405–22418. Curran Associates, Inc., 2021.

Rujikorn Charakorn, Edoardo Cetin, Yujin Tang, and Robert Tjarko Lange. Text-to-lora: Instant transformer adaption, 2025.

Guoguo Chen, Shuzhou Chai, Guan-Bo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, Mingjie Jin, Sanjeev Khudanpur, Shinji Watanabe, Shuaijiang Zhao, Wei Zou, Xiangang Li, Xuchen Yao, Yongqing Wang, Zhao You, and Zhiyong Yan. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. In *Interspeech 2021*, pages 3670–3674, 2021.

Kuan-Yu Chen, Che-Ping Tsai, Da-Rong Liu, Hung-Yi Lee, and Lin shan Lee. Completely unsupervised phoneme recognition by a generative adversarial network harmonized with iteratively refined hidden markov models. In *Interspeech 2019*, pages 1856–1860, 2019.

Qingyu Chen, Alexis Allot, Robert Leaman, Rezarta Islamaj Doğan, and Zhiyong Lu. Overview of the biocreative vii litcovid track: multi-label topic classification for covid-19 literature annotation. In *Proceedings of the seventh BioCreative challenge evaluation workshop*, 2021.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518, 2022.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. Meditron-70b: Scaling medical pretraining for large language models, 2023.

Rebecca CHEN Hsueh Chu. 3.2. acoustic aspects of consonants.

Emily Cheng, Diego Doimo, Corentin Kervadec, Iuri Macocco, Jade Yu, Alessandro Laio, and Marco Baroni. Emergence of a high-dimensional abstraction phase in language transformers, 2025.

Chung-Cheng Chiu, James Qin, Yu Zhang, Jiahui Yu, and Yonghui Wu. Self-supervised learning with random-projection quantizer for speech recognition. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 3915–3924. PMLR, July 2022.

Kwanghee Choi, Ankita Pasad, Tomohiko Nakamura, Satoru Fukayama, Karen Livescu, and Shinji Watanabe. Self-supervised speech representations are more phonetic than semantic, 2024.

Leshem Choshen, Elad Venezian, Noam Slonim, and Yoav Katz. Fusing finetuned models for better pretraining, 2022.

Ju-Chieh Chou, Chung-Ming Chien, Wei-Ning Hsu, Karen Livescu, Arun Babu, Alexis Conneau, Alexei Baevski, and Michael Auli. Toward joint language modeling for speech units and text. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6582–6593, Singapore, December 2023. Association for Computational Linguistics.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor

Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways, 2022.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models, 2023.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014.

Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James Glass. An unsupervised autoregressive model for speech representation learning. In *Interspeech 2019*, pages 146–150, 2019.

Kenneth Ward Church. Emerging trends: Subwords, seriously? *Natural Language Engineering*, 26(3):375–382, 2020.

Christopher Cieri, David Miller, and Kevin Walker. The fisher corpus: a resource for the next generations of speech-to-text. In Maria Teresa Lino, Maria Francisca Xavier, Fátima Ferreira, Rute Costa, and Raquel Silva, editors, *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, May 2004. European Language Resources Association (ELRA).

Ann Clifton, Sravana Reddy, Yongze Yu, Aasish Pappu, Rezvaneh Rezapour, Hamed Bonab, Maria Eskevich, Gareth Jones, Jussi Karlgren, Ben Carterette, and Rosie Jones. 100,000

podcasts: A spoken English document corpus. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5903–5917, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12(null):2493–2537, November 2011.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics.

Jenny Copara, Julien Knafou, Nona Naderi, Claudia Moro, Patrick Ruch, and Douglas Teodoro. 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, 6 2020. ATALA et AFCP.

H. Cottez. *Dictionnaire des structures du vocabulaire savant: éléments et modèles de formation.* Collection Les usuels du Robert. Le Robert, 1980.

Santiago Cuervo and Ricard Marxer. Scaling properties of speech language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 351–361, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

Santiago Cuervo, Adel Moumen, Yanis Labrak, Sameer Khurana, Antoine Laurent, Mickael Rouvier, and Ricard Marxer. Text-speech language models with improved cross-modal transfer by aligning abstraction levels, 2025.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe reinforcement learning from human feedback. In *The Twelfth International Conference on Learning Representations*, 2024.

Clément Dalloux, Vincent Claveau, Natalia Grabar, Lucas Emanuel Silva Oliveira, Claudia Maria Cabral Moro, Yohan Bonescki Gumiel, and Deborah Ribeiro Carvalho. 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, 2021.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations*, 2024.

S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366, 1980.

Francesco Denti, Diego Doimo, Alessandro Laio, and Antonietta Mira. The generalized ratios intrinsic dimension estimator. *Scientific Reports*, 12(1):20005, November 2022.

Soham Deshmukh, Benjamin Elizalde, Rita Singh, and Huaming Wang. Pengi: An audio language model for audio tasks. In *NeurIPS*, 2023.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. GPT3.int8(): 8-bit matrix multiplication for transformers at scale. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

Jay DeYoung, Eric Lehman, Benjamin Nye, Iain Marshall, and Byron C. Wallace. Evidence inference 2.0: More data, better models. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 123–132, Online, July 2020. Association for Computational Linguistics.

Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. Longrope: Extending llm context window beyond 2 million tokens, 2024.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, page 67–73, New York, NY, USA, 2018. Association for Computing Machinery.

Rezarta Islamaj Dogan, Robert Leaman, and Zhiyong Lu. Ncbi disease corpus: A resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10, 2014.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression, 2022.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue, 2024.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue, 2024.

Kawin Ethayarajh. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China, November 2019. Association for Computational Linguistics.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. MASSIVE: A 1M-example multilingual natural language understanding dataset with 51 typologically-diverse languages. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4277–4302, Toronto, Canada, July 2023. Association for Computational Linguistics.

Meddy Fouquet, Katarzyna Pisanski, Nicolas Mathevon, and David Reby. Seven and up: individual differences in male voice fundamental frequency emerge before puberty and remain stable throughout adulthood. *R. Soc. Open Sci.*, 3(10):160395, October 2016.

Jean Baptiste Joseph Fourier. *Théorie Analytique de la Chaleur*. Cambridge Library Collection - Mathematics. Cambridge University Press, 2009.

Jason Alan Fries, Leon Weber, Natasha Seelam, Gabriel Altay, Debajyoti Datta, Samuele Garda, Myungsun Kang, Ruisi Su, Wojciech Kusa, Samuel Cahyawijaya, Fabio Barth, Simon Ott, Matthias Samwald, Stephen Bach, Stella Biderman, Mario Sänger, Bo Wang, Alison Callahan, Daniel León Periñán, Théo Gigant, Patrick Haller, Jenny Chim, Jose David Posada, John Michael Giorgi, Karthik Rangasai Sivaraman, Marc Pàmies, Marianna Nezhurina, Robert Martin, Michael Cullan, Moritz Freidank, Nathan Dahlberg, Shubhanshu Mishra, Shamik Bose, Nicholas Michio Broad, Yanis Labrak, Shlok S Deshmukh, Sid Kiblawi, Ayush Singh, Minh Chien Vu, Trishala Neeraj, Jonas Golde, Albert Villanova del Moral, and Benjamin Beilharz. Bigbio: A framework for data-centric biomedical natural language processing. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

Philip Gage. A new algorithm for data compression. *C Users J.*, 12(2):23–38, February 1994.

Daniel Galvez, Greg Diamos, Juan Ciro, Juan Felipe Cerón, Keith Achorn, Anjali Gopi, David Kanter, Maximilian Lam, Mark Mazumder, and Vijay Janapa Reddi. The people's speech: A large-scale diverse english speech recognition dataset for commercial usage. *CoRR*, abs/2111.09344, 2021.

Daniel Galvez, Greg Diamos, Juan Manuel Ciro Torres, Juan Felipe Cerón, Keith Achorn, Anjali Gopi, David Kanter, Max Lam, Mark Mazumder, and Vijay Janapa Reddi. The people's speech: A large-scale diverse english speech recognition dataset for commercial usage. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

Daniel Galvez, Greg Diamos, Juan Manuel Ciro Torres, Juan Felipe Cerón, Keith Achorn, Anjali Gopi, David Kanter, Max Lam, Mark Mazumder, and Vijay Janapa Reddi. The people's speech: A large-scale diverse english speech recognition dataset for commercial usage. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

John S. Garofolo, Lori F. Lamel, William M. Fisher, Jonathan G. Fiscus, David S. Pallett, Nancy L. Dahlgren, and Victor Zue. Timit acoustic-phonetic continuous speech corpus, 1993. Catalog Number: LDC93S1, Philadelphia, PA, USA.

Aldo Glielmo, Iuri Macocco, Diego Doimo, Matteo Carli, Claudio Zeni, Romina Wild, Maria d'Errico, Alex Rodriguez, and Alessandro Laio. Dadapy: Distance-based analysis of data-manifolds in python. *Patterns*, 3(10):100589, 2022.

J.J. Godfrey, E.C. Holliman, and J. McDaniel. Switchboard: telephone speech corpus for research and development. In *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520 vol.1, 1992.

Aaron Gokaslan, Vanya Cohen, Ellie Pavlick, and Stefanie Tellex. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus, 2019.

Google. Medgemma hugging face.

Natalia Grabar and Rémi Cardon. CLEAR – simple corpus for medical French. In Arne Jönsson, Evelina Rennes, Horacio Saggion, Sanja Stajner, and Victoria Yaneva, editors, *Proceedings of the 1st Workshop on Automatic Text Adaptation (ATA)*, pages 3–9, Tilburg, the Netherlands, November 2018. Association for Computational Linguistics.

Natalia Grabar, Vincent Claveau, and Clément Dalloux. CAS: French Corpus with Clinical Cases. In *Proceedings of the 9th International Workshop on Health Text Mining and Information Analysis (LOUHI)*, pages 1–7, Brussels, Belgium, October 2018.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong,

Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian

Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang,

Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024.

Cyril Grouin, Natalia Grabar, and Gabriel Illouz. Classification de cas cliniques et évaluation automatique de réponses d'étudiants : présentation de la campagne DEFT 2021 (clinical cases classification and automatic evaluation of student answers : Presentation of the DEFT 2021 challenge). In *Actes de la 28e Conférence sur le Traitement Automatique des Langues Naturelles. Atelier DÉfi Fouille de Textes (DEFT)*, pages 1–13, Lille, France, 6 2021. ATALA.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing, 2020.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23, October 2021.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1), October 2021.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Re, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, Jessica Wu, Joe Nudell, Joel Niklaus, John J Nay, Jonathan H. Choi, Kevin Tobia, Margaret Hagan, Megan Ma, Michael Livermore, Nikon Rasumov-Rahe, Nils Holzenberger, Noam Kolt, Peter Henderson, Sean Rehaag, Sharad Goel, Shang Gao, Spencer Williams, Sunny Gandhi, Tom Zur, Varun Iyer, and Zehua Li. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

Jiang Guo, Darsh Shah, and Regina Barzilay. Multi-source domain adaptation with mixture of experts. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4694–4703, Brussels, Belgium, October 2018. Association for Computational Linguistics.

Kishaloy Halder, Alan Akbik, Josip Krapac, and Roland Vollgraf. Task-aware representation of sentences for generic text classification. In *COLING*, pages 3202–3213, 2020.

Felix Hamborg, Norman Meuschke, Corinna Breitinger, and Bela Gipp. news-please: A generic news crawler and extractor. In *Proceedings of the 15th International Symposium of Information Science*, pages 218–223, March 2017.

Zellig S. Harris. Distributional structure. *WORD*, 10(2-3):146–162, 1954.

Michael Hassid, Tal Remez, Tu Anh Nguyen, Itai Gat, Alexis Conneau, Felix Kreuk, Jade Copet, Alexandre Défossez, Gabriel Synnaeve, Emmanuel Dupoux, Roy Schwartz, and Yossi Adi. Textually pretrained speech language models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*, 2023.

Kenneth Heafield. KenLM: Faster and smaller language model queries. In Chris Callison-Burch, Philipp Koehn, Christof Monz, and Omar F. Zaidan, editors, *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland, July 2011. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Estève. Ted-lium 3: Twice as much data and corpus repartition for experiments on speaker adaptation. In Alexey Karpov, Oliver Jokisch, and Rodmonga Potapova, editors, *Speech and Computer*, pages 198–208, Cham, 2018. Springer International Publishing.

Nicolas Hiebel, Olivier Ferret, Karën Fort, and Aurélie Névéol. CLISTER : A corpus for semantic textual similarity in French clinical narratives. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4306–4315, Marseille, France, June 2022. European Language Resources Association.

Nicolas Hiebel, Karën Fort, Aurélie Névéol, and Olivier Ferret. CLISTER : Un corpus pour la similarité sémantique textuelle dans des cas cliniques en français (CLISTER : A corpus for semantic textual similarity in French clinical narratives). In Yannick Estève, Tania Jiménez, Titouan Parcollet, and Marcely Zanon Boito, editors, *Actes de la 29e Conférence sur le Traitement Automatique des Langues Naturelles. Volume 1 : conférence principale*, pages 287–296, Avignon, France, 6 2022. ATALA.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022.

Valentin Hofmann, Janet Pierrehumbert, and Hinrich Schütze. Superbizarre is not superb: Derivational morphology improves BERT's interpretation of complex words. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3594–3608, Online, August 2021. Association for Computational Linguistics.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Trans. Audio Speech Lang.*, 29:3451–3460, 2021.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units, 2021.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission, 2019.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023.

Gabriel Ilharco, Mitchell Wortsman, Samir Yitzhak Gadre, Shuran Song, Hannaneh Hajishirzi, Simon Kornblith, Ali Farhadi, and Ludwig Schmidt. Patching open-vocabulary models by interpolating weights. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

Srinivasan Iyer, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Daniel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, Xian Li, Brian O'Horo, Gabriel Pereyra, Jeff Wang, Christopher Dewan, Asli Celikyilmaz, Luke Zettlemoyer, and Ves Stoyanov. Opt-iml: Scaling language model instruction meta learning through the lens of generalization, 2022.

Srinivasan Iyer, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Daniel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, Xian Li, Brian O'Horo, Gabriel Pereyra, Jeff Wang, Christopher Dewan, Asli Celikyilmaz, Luke Zettlemoyer, and Ves Stoyanov. Opt-iml: Scaling language model instruction meta learning through the lens of generalization, 2023.

Harold Jeffreys. An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 186(1007):453–461, 1946.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 24678–24704. Curran Associates, Inc., 2023.

Ye Jia, Yu Zhang, Ron J. Weiss, Quan Wang, Jonathan Shen, Fei Ren, Zhifeng Chen, Patrick Nguyen, Ruoming Pang, Ignacio Lopez Moreno, and Yonghui Wu. Transfer learning from speaker verification to multispeaker text-to-speech synthesis. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 4485–4495, Red Hook, NY, USA, 2018. Curran Associates Inc.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams, 2020.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China, November 2019. Association for Computational Linguistics.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. In *The Eleventh International Conference on Learning Representations*, 2023.

Alistair E.W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3(1):160035, May 2016.

Jaehun Jung, Lianhui Qin, Sean Welleck, Faeze Brahman, Chandra Bhagavatula, Ronan Le Bras, and Yejin Choi. Maieutic prompting: Logically consistent reasoning with recursive explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1266–1279, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux. Libri-light: A benchmark for asr with limited or no supervision. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7669–7673, 2020.

J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux. Libri-light: A benchmark for asr with limited or no supervision. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7669–7673, 2020.

Wei Kang, Xiaoyu Yang, Zengwei Yao, Fangjun Kuang, Yifan Yang, Liyong Guo, Long Lin, and Daniel Povey. Libriheavy: a 50,000 hours asr corpus with punctuation casing and context, 2023.

Wei Kang, Xiaoyu Yang, Zengwei Yao, Fangjun Kuang, Yifan Yang, Liyong Guo, Long Lin, and Daniel Povey. Libriheavy: A 50,000 hours asr corpus with punctuation casing and context. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10991–10995, 2024.

Akbir Khan, John Hughes, Dan Valentine, Laura Ruis, Kshitij Sachan, Ansh Radhakrishnan, Edward Grefenstette, Samuel R. Bowman, Tim Rocktäschel, and Ethan Perez. Debating with more persuasive llms leads to more truthful answers. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Tushar Khot, Ashish Sabharwal, and Peter Clark. Scitail: A textual entailment dataset from science question answering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), April 2018.

Tushar Khot, Ashish Sabharwal, and Peter Clark. Scitail: A textual entailment dataset from science question answering. In *AAAI*, 2018.

Jaehyeon Kim, Jungil Kong, and Juhee Son. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *ICML*, volume 139, pages 5530–5540, 2021.

Jinhwa Kim, Ali Derakhshan, and Ian Harris. Robust safety classifier against jailbreaking attacks: Adversarial prompt shield. In Yi-Ling Chung, Zeerak Talat, Debora Nozza, Flor Miriam Plaza-del Arco, Paul Röttger, Aida Mostafazadeh Davani, and Agostina Calabrese, editors, *Proceedings of the 8th Workshop on Online Abuse and Harms (WOAH 2024)*, pages 159–170, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

Alican Kocabiyikoglu, François Portet, Prudence Gibert, Hervé Blanchon, Jean-Marc Babouchkine, and Gaëtan Gavazzi. A spoken drug prescription dataset in french for spoken language understanding. In *13th Language Resources and Evaluation Conference (LREC 2022)*, 2022.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: generative adversarial networks for efficient and high fidelity speech synthesis. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

Jan A Kors, Simon Clematide, Saber A Akhondi, Erik M van Mulligen, and Dietrich Rebholz-Schuhmann. A multilingual gold-standard corpus for biomedical concept recognition: the Mantra GSC. *Journal of the American Medical Informatics Association*, 22(5):948–956, 05 2015.

Felix Kreuk, Joseph Keshet, and Yossi Adi. Self-supervised contrastive learning for unsupervised phoneme segmentation. In *Interspeech*, pages 3705–3709, 2020.

Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia, July 2018. Association for Computational Linguistics.

Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Eduardo Blanco and Wei Lu, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics.

Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *CoRR*, abs/1808.06226, 2018.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved rvqgan. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Tom Labiausse, Laurent Mazaré, Edouard Grave, Patrick Pérez, Alexandre Défossez, and Neil Zeghidour. High-fidelity simultaneous speech-to-speech translation, 2025.

Yanis Labrak, Adrien Bazoge, Béatrice Daille, Mickael Rouvier, and Richard Dufour. How important is tokenization in French medical masked language models? In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8223–8234, Torino, Italia, May 2024. ELRA and ICCL.

Yanis Labrak, Adrien Bazoge, Richard Dufour, Beatrice Daille, Pierre-Antoine Gourraud, Emmanuel Morin, and Mickael Rouvier. FrenchMedMCQA: A French multiple-choice question answering dataset for medical domain. In Alberto Lavelli, Eben Holderness, Antonio Jimeno Yepes, Anne-Lyse Minard, James Pustejovsky, and Fabio Rinaldi, editors, *Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI)*, pages 41–46, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics.

Yanis Labrak, Adrien Bazoge, Richard Dufour, Béatrice Daille, Pierre-Antoine Gourraud, Emmanuel Morin, and Mickael Rouvier. FrenchMedMCQA: A French Multiple-Choice

Question Answering Dataset for Medical domain. In *Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI)*, Abou Dhabi, United Arab Emirates, December 2022.

Yanis Labrak, Adrien Bazoge, Richard Dufour, Mickael Rouvier, Emmanuel Morin, Béatrice Daille, and Pierre-Antoine Gourraud. DrBERT: A robust pre-trained model in French for biomedical and clinical domains. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16207–16221, Toronto, Canada, July 2023. Association for Computational Linguistics.

Yanis Labrak, Adrien Bazoge, Oumaima El Khettari, Mickael Rouvier, Pacome Constant Dit Beaufils, Natalia Grabar, Béatrice Daille, Solen Quiniou, Emmanuel Morin, Pierre-Antoine Gourraud, and Richard Dufour. DrBenchmark: A large language understanding evaluation benchmark for French biomedical domain. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5376–5390, Torino, Italia, May 2024. ELRA and ICCL.

Yanis Labrak, Adrien Bazoge, Oumaima El Khettari, Mickael Rouvier, Pacome Constant dit Beaufils, Natalia Grabar, Beatrice Daille, Solen Quiniou, Emmanuel Morin, Pierre-Antoine Gourraud, and Richard Dufour. Drbenchmark: A large language understanding evaluation benchmark for french biomedical domain. *arXiv preprint arXiv:2402.13432*, 2024.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. Biomistral: A collection of open-source pretrained large language models for medical domains, 2024.

Yanis Labrak, Adel Moumen, Richard Dufour, and Mickael Rouvier. Zero-shot end-to-end spoken question answering in medical domain. In *Interspeech 2024*, pages 2020–2024, 2024.

Yanis Labrak, Mickaël Rouvier, and Richard Dufour. 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, June 2023. Florian Boudin.

Yanis Labrak, Mickael Rouvier, and Richard Dufour. MORFITT : Un corpus multi-labels d'articles scientifiques français dans le domaine biomédical. In Florian Boudin, Béatrice Daille, Richard Dufour, Oumaima El, Maël Houbre, Léane Jourdan, and Nihel Kooli, editors, *Actes de CORIA-TALN 2023. Actes de l'atelier "Analyse et Recherche de Textes Scientifiques" (ARTS)@TALN 2023*, pages 66–70, Paris, France, 6 2023. ATALA.

Yanis Labrak, Mickael Rouvier, and Richard Dufour. A zero-shot and few-shot study of instruction-finetuned large language models applied to clinical and biomedical tasks. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2049–2066, Torino, Italia, May 2024. ELRA and ICCL.

Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, and Emmanuel Dupoux. On generative spoken language modeling from raw audio. *Transactions of the Association for Computational Linguistics*, 9:1336–1354, 2021.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France, May 2020. European Language Resources Association.

Chia-Hsuan Lee, Szu-Lin Wu, Chi-Liang Liu, and Hung-yi Lee. Spoken squad: A study of mitigating the impact of speech recognition errors on listening comprehension. *Interspeech*, pages 3459–3463, 2018.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, September 2019.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 09 2019.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. Biocreative V CDR task corpus: a resource for chemical disease relation extraction. *Database J. Biol. Databases Curation*, 2016, 2016.

Yunxiang Li, Zihan Li, Kai Zhang, Ruilong Dan, Steve Jiang, and You Zhang. Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge, 2023.

Yunxiang Li, Zihan Li, Kai Zhang, Ruilong Dan, Steve Jiang, and You Zhang. Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge, 2023.

Jiawei Liang, Siyuan Liang, Man Luo, Aishan Liu, Dongchen Han, Ee-Chien Chang, and Xiaochun Cao. Vl-trojan: Multimodal instruction backdoor attacks against autoregressive visual language models, 2024.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023. Featured Certification, Expert Certification.

Guan-Ting Lin, Prashanth Gurunath Shivakumar, Aditya Gourav, Yile Gu, Ankur Gandhe, Hung yi Lee, and Ivan Bulyko. Align-slm: Textless spoken language models with reinforcement learning from ai feedback, 2024.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Xingyu Dang, and Song Han. Awq: Activation-aware weight quantization for llm compression and acceleration. *arXiv preprint arXiv:2306.00978*, 2023.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics.

DA Lindberg, BL Humphreys, and AT McCray. The Unified Medical Language System. *Methods Inf Med*, 32(4):281–291, 1993.

Samuel Lipping, Parthasaarathy Sudarsanam, Konstantinos Drossos, and Tuomas Virtanen. Clotho-aqa: A crowdsourced dataset for audio question answering, 2022.

Andy T. Liu, Shang-Wen Li, and Hung-yi Lee. Tera: Self-supervised learning of transformer encoder representation for speech. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2351–2366, 2021.

Yan Liu, Renren Jin, Ling Shi, Zheng Yao, and Deyi Xiong. Finemath: A fine-grained mathematical evaluation benchmark for chinese large language models, 2024.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2020.

Andrea Lösch, Valérie Mapelli, Stelios Piperidis, Andrejs Vasiļjevs, Lilli Smal, Thierry Declerck, Eileen Schnur, Khalid Choukri, and Josef van Genabith. European language resource coordination: Collecting language resources for public sector multilingual information management. In Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Bernardo Magnini, Begoña Altuna, Alberto Lavelli, Manuela Speranza, and Roberto Zanoli. The e3c project: Collection and annotation of a multilingual corpus of clinical cases. *Proceedings of the Seventh Italian Conference on Computational Linguistics CLiC-it 2020*, 2020.

Gallil Maimon, Avishai Elmakies, and Yossi Adi. Slamming: Training a speech language model on one gpu in a day, 2025.

Soumi Maiti, Yifan Peng, Shukjae Choi, Jee weon Jung, Xuankai Chang, and Shinji Watanabe. Voxtlm: unified decoder-only models for consolidating speech recognition/synthesis and speech/text continuation tasks, 2024.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online, July 2020. Association for Computational Linguistics.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suá rez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. CamemBERT: a Tasty French Language Model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7203—-7219. Association for Computational Linguistics, 2020.

Maurová Paillereau, Nikola. Do isolated vowels represent vowel targets in french? an acoustic study on coarticulation. *SHS Web of Conferences*, 27:09003, 2016.

Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *Interspeech 2017*, pages 498–502, 2017.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory F. Diamos, Erich Elsen, David García, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed precision training. *CoRR*, abs/1710.03740, 2017.

Microsoft, :, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, Dong Chen, Dongdong Chen, Junkun Chen, Weizhu Chen, Yen-Chun Chen, Yi ling Chen, Qi Dai, Xiyang Dai, Ruchao Fan, Mei Gao, Min Gao, Amit Garg, Abhishek Goswami, Junheng Hao, Amr Hendy, Yuxuan Hu, Xin Jin, Mahmoud Khademi, Dongwoo Kim, Young Jin Kim, Gina Lee, Jinyu Li, Yunsheng Li, Chen Liang, Xihui Lin, Zeqi Lin, Mengchen Liu, Yang Liu, Gilsinia Lopez, Chong Luo, Piyush Madan, Vadim Mazalov, Arindam Mitra, Ali Mousavi, Anh Nguyen, Jing Pan, Daniel Perez-Becker, Jacob Platin, Thomas Portet, Kai Qiu, Bo Ren, Liliang Ren, Sambuddha Roy, Ning Shang, Yelong Shen, Saksham Singhal, Subhojit Som, Xia Song, Tetyana Sych, Praneetha Vaddamanu, Shuohang Wang, Yiming Wang, Zhenghao Wang, Haibin Wu, Haoran Xu, Weijian Xu, Yifan Yang, Ziyi Yang, Donghan Yu, Ishmam Zabir, Jianwen Zhang, Li Lyna Zhang, Yunan Zhang, and Xiren Zhou. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras, 2025.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. Reframing instructional prompts to GPTk's language. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 589–612, Dublin, Ireland, May 2022. Association for Computational Linguistics.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual generalization through multitask finetuning, 2023.

Hiroki Nakayama. seqeval: A python framework for sequence labeling evaluation, 2018. Software available from https://github.com/chakki-works/seqeval.

Tarek Naous, Michael J Ryan, Alan Ritter, and Wei Xu. Having beer after prayer? measuring cultural bias in large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16366–16393, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

Kotonya Neema and Francesca Toni. Explainable automated fact-checking for public health claims. *arXiv preprint arXiv:2010.09926*, 2020.

Aurélie Névéol, Cyril Grouin, Jérémy Leixa, Sophie Rosset, and Pierre Zweigenbaum. The quaero french medical corpus : A ressource for medical entity recognition and normalization. 2014.

Mariana Neves, Antonio Jimeno Yepes, Aurélie Névéol, Rachel Bawden, Giorgio Maria Di Nunzio, Roland Roller, Philippe Thomas, Federica Vezzani, Maika Vicente Navarro, Lana Yeganova, Dina Wiemann, and Cristian Grozea. Findings of the WMT 2023 biomedical translation shared task: Evaluation of ChatGPT 3.5 as a comparison system. In Philipp Koehn, Barry Haddow, Tom Kocmi, and Christof Monz, editors, *Proceedings of the Eighth Conference on Machine Translation*, pages 43–54, Singapore, December 2023. Association for Computational Linguistics.

Tu Anh Nguyen, Maureen de Seyssel, Patricia Rozé, Morgane Rivière, Evgeny Kharitonov, Alexei Baevski, Ewan Dunbar, and Emmanuel Dupoux. The zero resource speech benchmark 2021: Metrics and baselines for unsupervised spoken language modeling. *CoRR*, abs/2011.11588, 2020.

Tu Anh Nguyen, Maureen de Seyssel, Patricia Rozé, Morgane Rivière, Evgeny Kharitonov, Alexei Baevski, Ewan Dunbar, and Emmanuel Dupoux. The zero resource speech benchmark 2021: Metrics and baselines for unsupervised spoken language modeling, 2020.

Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R. Costa-jussa, Maha Elbayad, Sravya Popuri, Christophe Ropers, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, Mary Williamson, Gabriel Synnaeve, Juan Pino, Benoit Sagot, and Emmanuel Dupoux. Spirit lm: Interleaved spoken and written language model, 2024.

Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R. Costa-jussa, Maha Elbayad, Sravya Popuri, Christophe Ropers, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, Mary Williamson, Gabriel Synnaeve, Juan Pino, Benoit Sagot, and Emmanuel Dupoux. Spirit lm: Interleaved spoken and written language model, 2024.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. Capabilities of gpt-4 on medical challenge problems, 2023.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. Capabilities of gpt-4 on medical challenge problems, 2023.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry,

Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2023.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-

Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong

Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 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, 2019. Leibniz-Institut für Deutsche Sprache.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In Gerardo Flores, George H Chen, Tom Pollard, Joyce C Ho, and Tristan Naumann, editors, *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR, April 2022.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015.

Se Jin Park, Julian Salazar, Aren Jansen, Keisuke Kinoshita, Yong Man Ro, and RJ Skerry-Ryan. Long-form speech generation with spoken language models, 2024.

Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. Layer-wise analysis of a self-supervised speech representation model. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 914–921, 2021.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Curran Associates Inc., Red Hook, NY, USA, 2019.

Yifan Peng, Shankai Yan, and Zhiyong Lu. Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets. In *Proceedings of the 2019 Workshop on Biomedical Natural Language Processing (BioNLP 2019)*, pages 58–65, 2019.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *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, June 2018. Association for Computational Linguistics.

Long N. Phan, James T. Anibal, Hieu Tran, Shaurya Chanana, Erol Bahadroglu, Alec Peltekian, and Grégoire Altan-Bonnet. Scifive: a text-to-text transformer model for biomedical literature, 2021.

John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, Alexei Baevski, Yossi Adi, Xiaohui Zhang, Wei-Ning Hsu, Alexis Conneau, and Michael Auli. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52, 2024.

Erwan Pépiot. Male and female speech: a study of mean f0, f0 range, phonation type and speech rate in parisian french and american english speakers. In *Speech Prosody 2014*, pages 305–309, 2014.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision, 2022.

Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), January 2020.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Okko Johannes Räsänen, Unto Kalervo Laine, and Toomas Altosaar. An improved speech segmentation quality measure: The r-value. In *Tenth Annual Conference of the International Speech Communication Association (Interspeech 2009)*, pages 1851–1854, 2009. Funded by EU FP6 FET project ACORNS (FP6-034362).

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866, 2020.

Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *ICML*, volume 37, pages 2152–2161, 2015.

Paul K. Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, Hannah Muckenhirn, Dirk Padfield, James Qin, Danny Rozenberg, Tara Sainath, Johan Schalkwyk, Matt Sharifi, Michelle Tadmor Ramanovich, Marco Tagliasacchi, Alexandru Tudor, Mihajlo Velimirović, Damien Vincent, Jiahui Yu, Yongqiang Wang, Vicky Zayats, Neil Zeghidour, Yu Zhang, Zhishuai Zhang, Lukas Zilka, and Christian Frank. Audiopalm: A large language model that can speak and listen, 2023.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*, 2022.

Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK, 1994.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics.

Jiatong Shi, Xutai Ma, Hirofumi Inaguma, Anna Sun, and Shinji Watanabe. Mmm: Multi-layer multi-residual multi-stream discrete speech representation from self-supervised learning model. In *Interspeech 2024*, pages 2569–2573, 2024.

Chaitanya Shivade. Mednli — a natural language inference dataset for the clinical domain, 2017.

Ken Shoemake. Animating rotation with quaternion curves. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '85, page 245–254, New York, NY, USA, 1985. Association for Computing Machinery.

Sidak Pal Singh and Martin Jaggi. Model fusion via optimal transport. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 22045–22055. Curran Associates, Inc., 2020.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, Mike Schaekermann, Amy Wang, Mohamed Amin, Sami Lachgar, Philip Mansfield, Sushant Prakash, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Nenad Tomasev, Yun Liu, Renee Wong, Christopher Semturs, S. Sara Mahdavi, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Shekoofeh Azizi, Alan Karthikesalingam, and Vivek Natarajan. Towards expert-level medical question answering with large language models, 2023.

Thomas Sounack, Joshua Davis, Brigitte Durieux, Antoine Chaffin, Tom J. Pollard, Eric Lehman, Alistair E. W. Johnson, Matthew McDermott, Tristan Naumann, and Charlotta Lindvall. Bioclinical modernbert: A state-of-the-art long-context encoder for biomedical and clinical nlp, 2025.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023.

Rich Sutton. The bitter lesson. https://www.cs.utexas.edu/~eunsol/courses/data/bitter_lesson.pdf, 2019.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun MA, and Chao Zhang. SALMONN: Towards generic hearing abilities for large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. UL2: Unifying language learning paradigms. In *The Eleventh International Conference on Learning Representations*, 2023.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Siamak Shakeri, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. Ul2: Unifying language learning paradigms, 2023.

Erik R. Thomas. 420prosodic features of african american english. In *The Oxford Handbook of African American Language*. Oxford University Press, 07 2015.

Jörg Tiedemann and Lars Nygaard. The OPUS corpus - parallel and free: http://logos.uio.no/opus. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, May 2004. European Language Resources Association (ELRA).

Rian Touchent, Laurent Romary, and Eric De La Clergerie. CamemBERT-bio : Un modèle de langue français savoureux et meilleur pour la santé. In Christophe Servan and Anne Vilnat, editors, *Actes de CORIA-TALN 2023. Actes de la 30e Conférence sur le Traitement Automatique des Langues Naturelles (TALN), volume 1 : travaux de recherche originaux – articles longs*, pages 323–334, Paris, France, 6 2023. ATALA.

Rian Touchent, Laurent Romary, and Eric de la Clergerie. Camembert-bio: a tasty french language model better for your health, 2023.

Christian Touratier. Chapitre V. Les classes de morphèmes. In *Morphologie et morphématique : Analyse en morphèmes*, Langues et langage, pages 78–114. Presses universitaires de Provence, Aix-en-Provence, December 2012.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):138, 2015.

Lewis Tunstall, Edward Emanuel Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro Von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M Rush, and Thomas Wolf. Zephyr: Direct distillation of LM alignment. In *First Conference on Language Modeling*, 2024.

Lewis Tunstall, Edward Emanuel Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro Von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M Rush, and Thomas Wolf. Zephyr: Direct distillation of LM alignment. In *First Conference on Language Modeling*, 2024.

Arnon Turetzky and Yossi Adi. Last: Language model aware speech tokenization, 2024.

Ozlem Uzuner, Ira Goldstein, Yuan Luo, and Isaac Kohane. Identifying patient smoking status from medical discharge records. *Journal of the American Medical Informatics Association*, 15(1):14–24, 01 2008.

Lucrezia Valeriani, Diego Doimo, Francesca Cuturello, Alessandro Laio, Alessio ansuini, and Alberto Cazzaniga. The geometry of hidden representations of large transformer models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 1096–1103, New York, NY, USA, 2008. Association for Computing Machinery.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. *SuperGLUE: a stickier benchmark for general-purpose language understanding systems*. Curran Associates Inc., Red Hook, NY, USA, 2019.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In Tal Linzen, Grzegorz Chrupała, and Afra Alishahi, editors, *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 993–1003, Online, August 2021. Association for Computational Linguistics.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 993–1003, Online, August 2021. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language model with self generated instructions, 2022.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc.

Sibo Wei, Xueping Peng, Yi fei Wang, Jiasheng Si, Weiyu Zhang, Wenpeng Lu, Xiaoming Wu, and Yinglong Wang. Biancang: A traditional chinese medicine large language model, 2024.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

Shu wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y. Lin, Andy T. Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, Tzu-Hsien Huang, Wei-Cheng Tseng, Ko tik Lee, Da-Rong Liu, Zili Huang, Shuyan Dong, Shang-Wen Li, Shinji Watanabe, Abdelrahman Mohamed, and Hung yi Lee. Superb: Speech processing universal performance benchmark. In *Interspeech 2021*, pages 1194–1198, 2021.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France, May 2020. European Language Resources Association.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2019.

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Rau-

nak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. Bloom: A 176b-parameter open-access multilingual language model, 2022.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 23965–23998. PMLR, July 2022.

Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. Pmc-llama: Towards building open-source language models for medicine, 2023.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation, 2016.

Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP*, pages 1–5, 2023.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. Ex machina: Personal attacks seen at scale, 2017.

Honglin Xiong, Sheng Wang, Yitao Zhu, Zihao Zhao, Yuxiao Liu, Linlin Huang, Qian Wang, and Dinggang Shen. Doctorglm: Fine-tuning your chinese doctor is not a herculean task, 2023.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online, June 2021. Association for Computational Linguistics.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-merging: Resolving interference when merging models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, Junyang Lin, Kai Dang, Kexin Yang, Le Yu, Mei Li, Minmin Sun, Qin Zhu, Rui Men, Tao He, Weijia Xu, Wenbiao Yin, Wenyuan Yu, Xiafei Qiu, Xingzhang Ren, Xinlong Yang, Yong Li, Zhiying Xu, and Zipeng Zhang. Qwen2.5-1m technical report, 2025.

Hao Yang, Jinming Zhao, Gholamreza Haffari, and Ehsan Shareghi. Investigating pre-trained audio encoders in the low-resource condition. In *Interspeech 2023*, pages 1498–1502, 2023.

Songhua Yang, Hanjie Zhao, Senbin Zhu, Guangyu Zhou, Hongfei Xu, Yuxiang Jia, and Hongying Zan. Zhongjing: Enhancing the chinese medical capabilities of large language model through expert feedback and real-world multi-turn dialogue. *arXiv preprint arXiv:2308.03549*, 2023.

Junjie Ye, Xuanting Chen, Nuo Xu, Can Zu, Zekai Shao, Shichun Liu, Yuhan Cui, Zeyang Zhou, Chao Gong, Yang Shen, Jie Zhou, Siming Chen, Tao Gui, Qi Zhang, and Xuanjing Huang. A comprehensive capability analysis of gpt-3 and gpt-3.5 series models, 2023.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Marcely Zanon Boito, Vivek Iyer, Nikolaos Lagos, Laurent Besacier, and Ioan Calapodescu. mhubert-147: A compact multilingual hubert model. In *Interspeech 2024*, pages 3939–3943, 2024.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec, 2021.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec, 2021.

Aohan Zeng, Zhengxiao Du, Mingdao Liu, Lei Zhang, Shengmin Jiang, Yuxiao Dong, and Jie Tang. Scaling speech-text pre-training with synthetic interleaved data, 2024.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15757–15773, Singapore, December 2023. Association for Computational Linguistics.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities, 2023.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.

Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu. Speechtokenizer: Unified speech tokenizer for speech language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Yian Zhang, Alex Warstadt, Haau-Sing Li, and Samuel R. Bowman. When do you need billions of words of pretraining data?, 2020.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. Pytorch fsdp: Experiences on scaling fully sharded data parallel, 2023.

Zihan Zhao, Yiyang Jiang, Heyang Liu, Yanfeng Wang, and Yu Wang. Librisqa: Advancing free-form and open-ended spoken question answering with a novel dataset and framework, 2023.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.