

# BERTinchamps: Cost-Effective Training of Large Language Models for Medical Tasks in French

Amaury Fierens<sup>1</sup>, Sébastien Jodogne<sup>1</sup>

<sup>1</sup>*Institute of Information and Communication Technologies, Electronics and Applied Mathematics, Louvain School of Engineering, UCLouvain, 1348 Louvain-la-Neuve, Belgium*

## Abstract

Many medical applications are envisioned for Large Language Models (LLMs), such as the automated summary of the health condition of a patient, or the automated codification of electronic health records. Even though the training of LLMs directly inside hospitals is highly desirable to exploit the local clinical data while avoiding data privacy concerns, this process requires a costly, complex computing infrastructure. This paper explores the recent Cramming approach as a cost-effective way to train LLMs within medical institutions in one day using one GPU. We show that the Cramming approach that was originally designed for English can be transposed to French, and that the resulting models can be successfully fine-tuned to healthcare-related tasks in the French language. This research opens the path to the creation of LLMs that are tailored to the specific needs of institutions that handle sensitive textual data in another language than English.

## Keywords

Large language models, Medical documents, Downscaled training

## 1. Introduction

The field of Natural Language Processing (NLP) is currently attracting a lot of attention in the context of healthcare [2, 3]. Indeed, automating tasks such as the codification of electronic health records (EHRs) could be highly valuable to monitor the quality of treatments [4], to help with hospital payment reimbursement [5], to provide a summary of the health condition of a patient [6], yet to detect diseases at an early stage by using clinical codes as biomarkers [7].

In particular, the recent major advances in the field of Large Language Models (LLMs) are opening great opportunities for NLP [8, 9]. A growing number of physicians are enthusiastic about using modern tools such as the well-known ChatGPT chatbot to help with medical tasks [2, 10]. As of June 2023, ChatGPT internally uses the closed-source, proprietary LLMs GPT-3.5 and GPT-4 [9], which introduces a strong dependency upon the proprietary infrastructure of the OpenAI platform. But, the protection of patient privacy prevents the direct use of such proprietary LLMs in a clinical context because they are generally cloud-based, while regulations such as the General Data Protection Regulation (GDPR) in Europe forbid medical data from leaving hospitals unless it has been at least pseudonymized. Similar difficulties are encountered when a hospital seeks to exploit LLMs in the context of clinical research.

---


NL4AI 2023: *Seventh Workshop on Natural Language for Artificial Intelligence, November 6-7th, 2023, Rome, Italy* [1]

✉ amaury.fierens@uclouvain.be (A. Fierens); sebastien.jodogne@uclouvain.be (S. Jodogne)

ORCID 0009-0003-3668-8804 (A. Fierens); 0000-0001-6685-7398 (S. Jodogne)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

This calls for the development of LLMs that can be entirely self-hosted inside the infrastructure of a hospital. Self-hosting is evidently highly desirable for inference on the EHRs of the hospital. However, besides inference, it is also important to be able to train LLMs inside the hospitals, to fine-tune them to the population of patients of one hospital or of one clinical department of interest. Self-hosting can be notably achieved by taking advantage of LLMs whose architecture has been published as open-source code, and whose pre-trained weights are available as open data. Early LLMs available as open-source and open-data include BERT [11] and GPT-2 [12]. More advanced models such as BLOOM [13], Cerebras-GPT [14], or LLaMA [15] are now available.

A difficulty with open, general-purpose LLMs is that they have been primarily trained on English datasets, without a specialization on the clinical language. This has motivated researchers to train LLMs using corpora containing medical documents. This is possible for English, for which corpora of sufficient size have emerged over the years. For instance, BioBERT [16] was trained on a dataset made of PubMed abstracts, while ClinicalBERT [17] was trained on MIMIC-III [18]. This shows that even though the BERT architecture has a number of parameters that is much smaller than more recent models, the variations of BERT can still be considered as compact Large Language Models with interesting applications related to healthcare. Unfortunately, there is still a lack of large-scale medical corpora for most languages besides English. In the context of French, only a handful of pre-trained LLMs for the clinical language are currently available. Those include DrBERT [19] that leverages the RoBERTa architecture [20] and that is trained from scratch using the biomedical corpus NACHOS that is not publicly available yet at the time of writing. The evaluation of such models on real-world clinical tasks is still a work in progress due to the lack of sufficiently large amount of domain-specific data in French.

An alternative to the use of LLMs that are pre-trained for healthcare applications would be to train the LLMs locally, inside the hospital, directly on the EHRs it hosts. This approach would have the great advantage of training models that reflect the local patient population of the hospital, while bringing privacy by design. By training a BERT model from scratch directly on the local electronic health records of the hospital, the resulting language model could be better adapted to healthcare-related tasks in that specific hospital, with respect to one model that would have been obtained from the fine-tuning of a general French model such as CamemBERT [21]. However, it is commonly considered that training a sufficiently expressive LLM from scratch is extremely demanding in terms of time and computational resources, as standard training processes require dozens of days of computation on powerful Graphics Processing Unit (GPU) that come at a high budget. Nevertheless, recent work has shown that it is possible to drastically reduce the training time of BERT-like LLMs by slightly modifying the BERT architecture and the way datasets are preprocessed [22]. This simplification is referred to as "Cramming", and the resulting LLM is called "crammed BERT." To the best of our knowledge, the Cramming approach has only been studied in the context of the English language and has not been applied to the medical field so far.

In this paper, we investigate the use of the Cramming recipe to train LLMs on healthcare-related tasks in French. Our results show that our crammed BERT model, referred to as BERTinchamps, achieves a performance that is close to that of DrBERT on selected tasks related to the medical field, while requiring only one single day of training. This contribution opens the path to the training of LLMs directly inside institutions that generate sensitive textual data,

such as hospitals, at a reasonable cost, while preserving the privacy of data.

## 2. Related Work

The idea of optimizing a LLM architecture to train it using less resources has already been explored in the literature. In 2015, the idea of Knowledge Distillation was introduced in neural networks [23], which enabled the training of smaller models from a bigger one, with little loss in performance. This approach was used to create distilled versions of well-known LLMs such as DistilBERT [24] and DistilGPT-2 [25]. Another technique consists in the quantization of the model. Since its introduction in 1990 [26], quantization has been widely applied to the Transformer-based architectures that underpin BERT, for instance in BinaryConnect [27], in the Bondarenko et al. paper that introduced quantization for BERT [28], in Q8BERT [29], or in BinaryBERT [30]. An even more recent paper has presented QLoRA [31], an efficient fine-tuning method for quantized models.

While those methods are extremely useful to reduce the size of an already existing LLM, they are not designed to train LLMs from scratch at a decent cost. The Cramming recipe is a recent contribution to serve this purpose [22]. The main goal of Cramming is to modify the architecture and the training process of classical BERT-like models, while adapting the preprocessing of the data, in order to determine how well such so-called "crammed BERT" models can perform after having been trained on one single GPU for one single day.

The Cramming approach is motivated by the scaling laws described by Kaplan et al. [32] that hold in the low-resource regime. These scaling laws suggest that it is not necessarily useful to reduce the number of parameters of BERT-like architectures. Instead of reducing the number of parameters, the Cramming recipe optimizes the model architecture and adjusts the training setup. Cramming also explores architectural enhancements that speed up the computation of the gradients. Slight improvements are obtained by disabling the QKV biases in the multi-head attention block [33], together with changes in the embedding blocks. The Cramming recipe also proposes hyperparameters that are adapted to the training of crammed BERT models. Finally, careful selection and processing of the training data is applied to extract well-suited tokens, enhancing the overall performance of the crammed models. These contributions have been shown to bring significant improvements, enabling the fast training of BERT-like models for English. However, the application of the Cramming recipe to other languages is still largely unexplored. Our paper steps into this gap, by mirroring the advancements of the Cramming approach in the French language, and by exploring how well crammed BERT models for French behave on healthcare-related tasks.

## 3. Methods

In this section, we describe our methodology and the parameters we used to mirror the Cramming recipe in the French language. The resulting crammed BERT model is referred to as BERTinchamps. The fine-tuning of BERTinchamps on selected healthcare-related tasks is then discussed.

### 3.1. Pre-training

BERTinchamps was trained on the French part of the OSCAR dataset [34]. The OSCAR dataset is an extensive multilingual corpus acquired through language classification and filtering of the Common Crawl corpus employing the goclassy architecture. A subset of 17GB of the French part of the OSCAR dataset was selected. Tokens were extracted using the WordPiece algorithm [35], with a vocabulary size  $n_{vocab} = 32768$ . After tokenization, the dataset size was 37GB. The cross-entropy loss was optimized, as it is usually the case if training BERT models. However, to accelerate the training, the context was reduced from a maximum sequence length of 512 tokens to 128 tokens, as recommended in the Cramming paper. In the same vein, the training objective was kept as masked language modeling, with the same masking proportions as in the original setup described by Devlin et al. [11].

AdamW was used as the optimizer [36], which is a modified version of the Adam optimizer [37] where weight decay is performed after controlling the parameter-wise step size. The parameters of AdamW were set as follows: Weight decay equals 0.01,  $\beta_1 = 0.9$ ,  $\beta_2 = 0.98$ , and  $\epsilon = 10^{-12}$ . A gradient clipping of 0.5 was also included. The learning rate was set to  $10^{-3}$  and, contrarily to the original instructions of the Cramming recipe, a slanted triangular learning rate was used as the scheduler, with a base percentage of 25% and a falloff of 0.25 as parameters. This adaptation was experimentally found to be more efficient for the French language than the one-cycle learning rate used for English language in the original paper. This might be caused either by the difference language structures, or by the content of the training dataset itself.

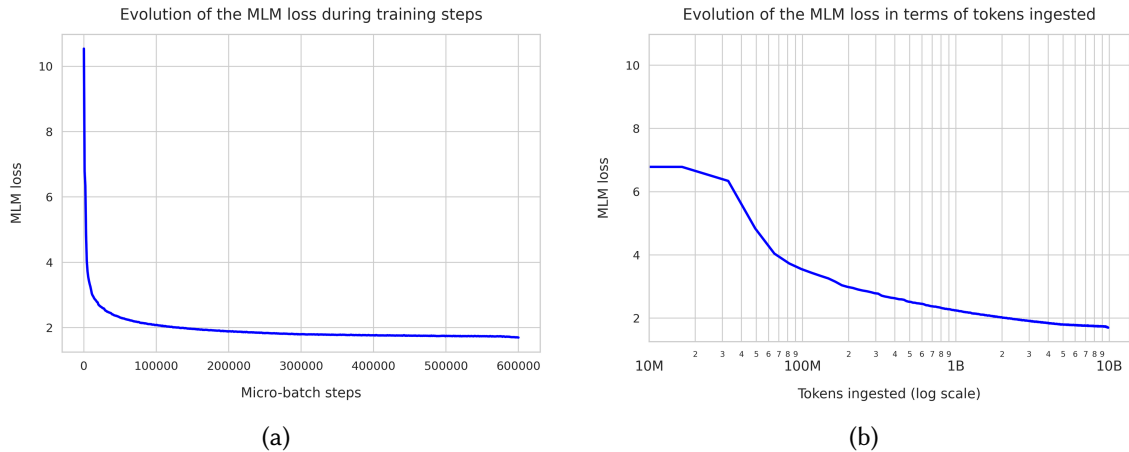
A micro-batch size of 128 and a batch size of 4096 were used, the Cramming setup being limited to the presence of one single GPU. This policy is rescheduled by linearly increasing the averaged number of micro-batches over the training time. The model was trained for 24 hours, as required by the Cramming recipe, on one NVIDIA A100 GPU with 40GB, and was called BERTinchamps. Figure 1 plots the evolution of the Masked Language Model (MLM) loss during the training.

### 3.2. Fine-tuning

In a first phase, three of the four datasets in the FLUE benchmark were used to assess the overall performance of BERTinchamps. FLUE is an equivalent of the GLUE benchmark for the French language [38]. In a second phase, to determine whether such a crammed model was promising for tasks related to the medical domain, BERTinchamps was fine-tuned on QUAEROFrenchMed [39], a French medical corpus that is made of two datasets, EMEA and MEDLINE. The use of QUAEROFrenchMed enables the comparison of BERTinchamps against DrBERT [19], a LLM for biomedical tasks in French.

#### 3.2.1. FLUE

The FLUE benchmark is widely used to evaluate LLMs for the French language. It has notably been used to test CamemBERT [21] and FlauBERT [38], two of the most powerful LLMs for French at the time of writing. BERTinchamps was evaluated on three of the four datasets that are comprised in the FLUE benchmark, namely the XNLI, CLS and PAWS-X datasets. Each FLUE dataset serves a specific role. The XNLI dataset, a French subset of MNLI, is related to the



**Figure 1:** Evolution of the MLM loss during the training, as a function of (a) the number of micro-batch steps, and (b) the number of tokens ingested by the model. These plots show that, as expected, the loss lowers as either of those numbers increases. Note that plot (b) is on a semi-log scale, with the  $x$ -axis expressed in base 10.

Natural Language Inference (NLI) task that identifies logical relationships between premise and hypothesis sentences. The CLS dataset is employed for text classification, using star-based labels to categorize Amazon reviews of books, DVDs, and music. Despite its initial separation into the latter three categories of reviews, we posit that classifier performance is not significantly impacted by consolidating these three categories. The PAWS-X dataset targets paraphrasing identification, where paired sentences are tagged as 1 for semantic equivalence or 0 otherwise.

Most of the FLUE tasks involve fine-tuning for sequence classification, for which the approach described in the Cramming paper was used. The AdamW optimizer was accordingly used for the fine-tuning, with parameters  $\beta_1 = 0.9$ ,  $\beta_2 = 0.98$ ,  $\epsilon = 10^{-6}$ , and a smaller learning rate of  $4 \cdot 10^{-5}$ . The cosine-decay scheduler was experimentally found to provide better performance for the fine-tuning. The model was trained for 10 epochs on CLS and PAWS-X, and for 5 epochs on XNLI. The training batch size was set to 16, while the testing batch size was set to 128.

### 3.2.2. QUAEROFrenchMed

The QUAEROFrenchMed benchmark is a Medical Named Entity Recognition task in the French language, whose purpose is to associate a clinical entity to each token of a medical text. The QUAEROFrenchMed benchmark is made of two distinct datasets, namely EMEA and MEDLINE, that share the same clinical entities of interest. MEDLINE is composed of a lot of short sentences coming from MEDLINE article titles, while EMEA is composed of a few long documents coming from drug descriptions.

The QUAEROFrenchMed benchmark involves the classification of tokens, which contrasts with the FLUE benchmark for which sequence classification fine-tuning was needed. To this end, a Linear Layer was added at the end of the BERTinchamps pre-trained model, with an output size equal to the number of clinical entities in QUAEROFrenchMed. This Linear Layer

was trained using cross-entropy loss. Moreover, the original datasets had to be adapted to meet the requirements associated with this classification setup. For MEDLINE, the annotations had to be processed, as some of the entities had multiple spans. As far as EMEA is concerned, the dataset was first made like MEDLINE by splitting the long documents into their individual sentences. The annotations also had to be processed for the same reason as MEDLINE. Each word of both datasets was tokenized and each sentence of EMEA was identified using the French blank model of the spaCy Python package [40] along with its Sentencizer tool. For both EMEA and MEDLINE, all the resulting words, along with their labels, were stored as a JSON file for further processing.

The AdamW optimizer was again used to fine-tune BERTinchamps on the EMEA and MEDLINE tasks. The default implementation of the AdamW trainer in the PyTorch package was used [41], with a smaller learning rate of  $10^{-4}$ . The slanted triangular learning rate was used as the scheduler, as it provided better performance in this case. The model was trained for 100 epochs, for both EMEA and MEDLINE. Both the training and testing batch sizes were set to 8.

## 4. Results

This section first presents the overall performance of the BERTinchamps model according to the FLUE benchmark. Secondly, the specific capabilities of the model related to the medical language are tested on the QUAEROFrenchMed benchmark.

### 4.1. FLUE

As explained in Section 3.2.1, BERTinchamps was compared to CamemBERT and FlauBERT, two state-of-the-art LLMs for the French language, on the CLS, PAWS-X, and XNLI tasks of the FLUE benchmark. Because the BERTinchamps model is a crammed BERT model with 110 millions of parameters, versions of CamemBERT and FlauBERT with a comparable number of parameters were considered (i.e., CamemBERT<sub>base</sub> and FlauBERT<sub>base</sub> that respectively contain 110 and 138 millions of parameters). Table 1 reports the final accuracy of each model on each task. The results show a difference of less than 4% in performance between the BERTinchamps crammed model and the state-of-the-art models.

**Table 1**

Accuracy comparison between the three LLMs on the three considered tasks of the FLUE benchmark.

Model	CLS	PAWS-X	XNLI
CamemBERT <sub>base</sub> <sup>†</sup>	<b>93.33*</b>	<b>90.14</b>	<b>81.2</b>
FlauBERT <sub>base</sub> <sup>†</sup>	93.21*	89.49	80.6
BERTinchamps	90.14	86.56	77.59

<sup>†</sup>Results reported in FlauBERT paper [38]

\*Results averaged from the 3 categories

**Table 2**

Accuracy comparison between the three LLMs on EMEA and MEDLINE, the two parts of QUAEROFrenchMed corpus.

Model	MEDLINE	EMEA
CamemBERT <sub>base</sub>	<b>80.60</b>	<b>90.54</b>
DrBERT <sub>7GB</sub>	79.54	89.87
BERTinchamps	79.43	89.38



**Table 3**

Label-level  $F_1$ -scores for DrBERT and BERTinchamps on EMEA and MEDLINE, along with the support in terms of token appearances.

Label	MEDLINE				EMEA			
	DrBERT		BERTinchamps		DrBERT		BERTinchamps	
	$F_1$	support	$F_1$	support	$F_1$	support	$F_1$	support
0	<b>88.6</b>	10006	88.4	8767	94.3	10785	<b>94.5</b>	9494
ANAT	48.8	622	<b>54</b>	745	<b>60.3</b>	119	49.8	131
CHEM	72.1	786	<b>74.9</b>	955	83	1874	<b>89.2</b>	2456
DEVI	13.3	85	<b>20.4</b>	80	30.8	191	<b>47.7</b>	207
DISO	73.4	2421	<b>76.3</b>	2905	<b>77.4</b>	624	74.1	819
GEOG	28.1	109	<b>62.7</b>	75	58.3	28	<b>70.3</b>	22
LIVB	61.8	603	<b>73</b>	598	<b>85.2</b>	315	81.1	321
OBJC	3.5	57	<b>9.2</b>	53	4	70	<b>4.6</b>	59
PHEN	2.4	78	<b>18.3</b>	78	15.1	39	<b>18.9</b>	36
PHYS	30.7	264	<b>31.3</b>	265	<b>53.1</b>	118	48.6	147
PROC	59.9	1084	<b>62.4</b>	1235	<b>70.4</b>	345	64.1	388

## 4.2. QUAEROFrenchMed

The capabilities of BERTinchamps on healthcare-related tasks in the French language were evaluated on EMEA and MEDLINE, the two datasets of the QUAEROFrenchMed benchmark. To this end, BERTinchamps was compared to both DrBERT, a recent LLM trained on biomedical data, and CamemBERT. The three LLMs were fine-tuned on QUAEROFrenchMed for the classification of tokens, using the experimental setup described in Section 3.2.2. The results are reported in Table 2. As can be seen in this table, the accuracy of BERTinchamps is close to DrBERT, and slightly worse than CamemBERT on the investigated tasks.

Tables 3 shows the label-level results for BERTinchamps and DrBERT. The  $F_1$ -score is reported together with the support for each label. Mismatch between the support counts is due to the use of different tokenizers in the two models, as the labels are put on the tokens that compose each word. Interestingly, while Table 2 tends to indicate that DrBERT provides better accuracy than BERTinchamps, Table 3 shows that BERTinchamps outperforms DrBERT in the MEDLINE dataset on every label but the 0 label, which is the default label of words without annotations. Moreover, BERTinchamps outperforms DrBERT on 6 out of 11 labels of the EMEA dataset. This tends to show that BERTinchamps and DrBERT share similar performance on the considered tasks.

## 4.3. Discussion

The results of Section 4.1 on the FLUE benchmark provide evidence that the Cramming recipe can be transposed to the French language, even though it was originally designed for English. The BERTinchamps crammed model performs well compared to state-of-the-art LLMs, even though it has only been trained on a single NVIDIA A100 GPU for 24 hours, which amounts to 4.1 exaFLOP. In comparison, CamemBERT has been trained on 256 NVIDIA V100 GPUs for 24 hours, for a total of 100 exaFLOP, while FlauBERT has been trained on 32 NVIDIA V100 GPUs

for 410 hours, for a total of 210 exaFLOP.

In addition, Section 4.2 indicates that the BERTinchamps model is promising on medical-related tasks, competing with a specialized LLM like DrBERT on an experimental setup derived from the QUAEROFrenchMed benchmark. The training of DrBERT has required 128 NVIDIA V100 GPUs for 20 hours, for a total of 41 exaFLOP. Summarizing, given the overall performance of BERTinchamps together with its low training cost, the Cramming recipe is a highly promising path to the local training of LLMs directly inside hospitals.

## 5. Conclusion

This paper contains two significant findings. Firstly, the Cramming recipe that was originally designed for English can be applied to the French language, with comparable effectiveness, resulting in the BERTinchamps model. This finding also suggests that Cramming is likely to be useful in other languages. Secondly, BERTinchamps can be fine-tuned to tasks that are related to the French medical language. Taken together, these two findings suggest that the self-hosted training of LLMs from scratch is within the reach of French-speaking institutions handling sensitive data, which includes the hospitals. By accelerating the training of LLMs by an order of magnitude, the Cramming recipe has the potential to strongly reduce the cost and complexity of the infrastructure to train LLMs from scratch in different languages. This not only opens the door to numerous applications within the realm of medical NLP, but also allows the creation of LLMs that are tailored to the very specific needs of the institutions where they are deployed.

Future work will consist in demonstrating the feasibility of training crammed models inside a hospital, for a real-world clinical task such as the automated codification of the EHRs managed by the hospital. Another promising research path will consist in leveraging federated learning for the collaborative training of one crammed model that is shared by a coalition of hospitals.

## References

- [1] E. Bassignana, D. Brunato, M. Polignano, A. Ramponi, Preface to the Seventh Workshop on Natural Language for Artificial Intelligence (NL4AI), in: Proceedings of the Seventh Workshop on Natural Language for Artificial Intelligence (NL4AI 2023) co-located with 22th International Conference of the Italian Association for Artificial Intelligence (AI\* IA 2023), 2023.
- [2] J. Li, A. Dada, J. Kleesiek, J. Egger, ChatGPT in Healthcare: A Taxonomy and Systematic Review, 2023. doi:10.1101/2023.03.30.23287899, pages: 2023.03.30.23287899.
- [3] B. Zhou, G. Yang, Z. Shi, S. Ma, Natural Language Processing for Smart Healthcare, IEEE Reviews in Biomedical Engineering abs/2110.15803 (2022) 1–17. doi:10.1109/RBME.2022.3210270, conference Name: IEEE Reviews in Biomedical Engineering.
- [4] P. J. Pronovost, M. D. Cole, R. M. Hughes, Remote Patient Monitoring During COVID-19: An Unexpected Patient Safety Benefit, JAMA 327 (2022) 1125–1126. doi:10.1001/jama.2022.2040.
- [5] L. Zhou, C. Cheng, D. Ou, H. Huang, Construction of a semi-automatic icd-10 coding



- system, *BMC medical informatics and decision making* 20 (2020) 1–12. doi:10.1186/s12911-020-1085-4.
- [6] V. J. Watzlaf, J. H. Garvin, S. Moeini, P. Anania-Firouzan, The Effectiveness of ICD-10-CM in Capturing Public Health Diseases, *Perspectives in Health Information Management / AHIMA, American Health Information Management Association* 4 (2007) 6.
- [7] T. Poongodi, D. Sumathi, P. Suresh, B. Balusamy, Deep Learning Techniques for Electronic Health Record (EHR) Analysis, in: A. K. Bhoi, P. K. Mallick, C.-M. Liu, V. E. Balas (Eds.), *Bio-inspired Neurocomputing, Studies in Computational Intelligence*, Springer, Singapore, 2021, pp. 73–103. doi:10.1007/978-981-15-5495-7\_5.
- [8] D. Khurana, A. Koli, K. Khatter, S. Singh, Natural language processing: state of the art, current trends and challenges, *Multimedia Tools and Applications* 82 (2023) 3713–3744. doi:10.1007/s11042-022-13428-4.
- [9] OpenAI, GPT-4 Technical Report, 2023. doi:10.48550/arXiv.2303.08774.
- [10] S. S. Biswas, Role of Chat GPT in Public Health, *Annals of Biomedical Engineering* 51 (2023) 868–869. doi:10.1007/s10439-023-03172-7.
- [11] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019. doi:10.48550/arXiv.1810.04805.
- [12] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, *OpenAI blog* 1 (2019) 9.
- [13] T. L. Scao, et al., BLOOM: A 176B-Parameter Open-Access Multilingual Language Model, 2023. doi:10.48550/arXiv.2211.05100.
- [14] N. Dey, G. Gosal, Zhiming, Chen, H. Khachane, W. Marshall, R. Pathria, M. Tom, J. Hestness, Cerebras-GPT: Open Compute-Optimal Language Models Trained on the Cerebras Wafer-Scale Cluster, 2023. doi:10.48550/arXiv.2304.03208.
- [15] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, G. Lample, LLaMA: Open and Efficient Foundation Language Models, 2023. doi:10.48550/arXiv.2302.13971.
- [16] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, *Bioinformatics* 36 (2019) 1234–1240. doi:10.1093/bioinformatics/btz682.
- [17] K. Huang, J. Altosaar, R. Ranganath, ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission, 2020. doi:10.48550/arXiv.1904.05342.
- [18] A. E. W. Johnson, T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, R. G. Mark, MIMIC-III, a freely accessible critical care database, *Scientific Data* 3 (2016) 160035. doi:10.1038/sdata.2016.35, number: 1 Publisher: Nature Publishing Group.
- [19] Y. Labrak, A. Bazoge, R. Dufour, M. Rouvier, E. Morin, B. Daille, P.-A. Gourraud, DrBERT: A Robust Pre-trained Model in French for Biomedical and Clinical domains, 2023. doi:10.48550/arXiv.2304.00958.
- [20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, 2019. doi:10.48550/ARXIV.1907.11692.
- [21] L. Martin, B. Muller, P. J. Ortiz Suárez, Y. Dupont, L. Romary, É. de la Clergerie, D. Seddah, B. Sagot, CamemBERT: a tasty French language model, in: *Proceedings of the 58th Annual*

- Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 7203–7219. URL: <https://aclanthology.org/2020.acl-main.645>. doi:10.18653/v1/2020.acl-main.645.
- [22] J. Geiping, T. Goldstein, Cramming: Training a Language Model on a Single GPU in One Day, 2022. doi:10.48550/arXiv.2212.14034.
- [23] G. Hinton, O. Vinyals, J. Dean, Distilling the Knowledge in a Neural Network, 2015. doi:10.48550/arXiv.1503.02531.
- [24] V. Sanh, L. Debut, J. Chaumond, T. Wolf, DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, 2020. doi:10.48550/arXiv.1910.01108.
- [25] T. Li, Y. E. Mesbahi, I. Kobyzev, A. Rashid, A. Mahmud, N. Anchuri, H. Hajimolahoseini, Y. Liu, M. Rezagholizadeh, A Short Study on Compressing Decoder-Based Language Models, 2021.
- [26] E. Fiesler, A. Choudry, H. J. Caulfield, Weight discretization paradigm for optical neural networks, in: H. Bartelt (Ed.), SPIE Proceedings, volume 1281 of *Optical Interconnections and Networks*, SPIE, The Hague, Netherlands, 1990, pp. 164–173. doi:10.1117/12.20700.
- [27] M. Courbariaux, Y. Bengio, J.-P. David, BinaryConnect: Training Deep Neural Networks with binary weights during propagations, 2016. doi:10.48550/arXiv.1511.00363.
- [28] Y. Bondarenko, M. Nagel, T. Blankevoort, Understanding and Overcoming the Challenges of Efficient Transformer Quantization, 2021. doi:10.48550/arXiv.2109.12948.
- [29] O. Zafrir, G. Boudoukh, P. Izsak, M. Wasserblat, Q8bert: Quantized 8bit bert, in: 2019 Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing - NeurIPS Edition (EMC2-NIPS), IEEE, Vancouver, Canada, 2019, pp. 36–39. doi:10.1109/EMC2-NIPS53020.2019.00016.
- [30] H. Bai, W. Zhang, L. Hou, L. Shang, J. Jin, X. Jiang, Q. Liu, M. Lyu, I. King, BinaryBERT: Pushing the Limit of BERT Quantization, 2021. doi:10.48550/arXiv.2012.15701.
- [31] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, QLoRA: Efficient Finetuning of Quantized LLMs, 2023.
- [32] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, D. Amodei, Scaling Laws for Neural Language Models, 2020. doi:10.48550/arXiv.2001.08361.
- [33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: *Advances in Neural Information Processing Systems*, volume 30, Curran Associates, Inc., Los Angeles, USA, 2017, pp. 5998–6008.
- [34] P. J. Ortiz Suárez, L. Romary, B. Sagot, A monolingual approach to contextualized word embeddings for mid-resource languages, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 1703–1714. URL: <https://aclanthology.org/2020.acl-main.156>. doi:10.18653/v1/2020.acl-main.156.
- [35] M. Schuster, K. Nakajima, Japanese and korean voice search, in: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, Kyoto, Japan, 2012, pp. 5149–5152.
- [36] I. Loshchilov, F. Hutter, Decoupled Weight Decay Regularization, 2019. doi:10.48550/arXiv.1711.05101.
- [37] D. P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, 2017. doi:10.48550/

arXiv.1412.6980.

- [38] H. Le, L. Vial, J. Frej, V. Segonne, M. Coavoux, B. Lecouteux, A. Allauzen, B. Crabbé, L. Besacier, D. Schwab, FlauBERT: Unsupervised Language Model Pre-training for French, 2020. doi:10.48550/arXiv.1912.05372.
- [39] A. Névéol, C. Grouin, J. Leixa, S. Rosset, P. Zweigenbaum, The QUAERO French medical corpus: A resource for medical entity recognition and normalization, in: Proc of BioTextMining Work, ELRA, Reykjavik, Iceland, 2014, pp. 24–30.
- [40] M. Honnibal, I. Montani, spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing, 2017. To appear.
- [41] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems 32, Vancouver, Canada, 2019, pp. 8024–8035.