Improving Legal Judgement Prediction in Romanian with Long Text Encoders

Mihai Masala*†, Traian Rebedea†, Horia Velicu‡
*Institute for Logic and Data Science, †University Politehnica of Bucharest, ‡BRD Groupe Societe Generale
mihai_dan.masala@upb.ro, traian.rebedea@upb.ro, horia.velicu@brd.ro

Abstract

In recent years,the entire field of Natural Language Processing (NLP) has enjoyed amazing novel results achieving almost human-like performance on a variety of tasks. Legal NLP domain has also been part of this process, as it has seen an impressive growth. However, general-purpose models are not readily applicable for legal domain. Due to the nature of the domain (e.g. specialized vocabulary, long documents) specific models and methods are often needed for Legal NLP. In this work we investigate both specialized and general models for predicting the final ruling of a legal case, task known as Legal Judgment Prediction (LJP). We particularly focus on methods to extend to sequence length of Transformer-based models to better understand the long documents present in legal corpora. Extensive experiments on 4 LJP datasets in Romanian, originating from 2 sources with significantly different sizes and document lengths, show that specialized models and handling long texts are critical for a good performance.

Keywords: legal judgement prediction, long context encoding, Romanian language

1. Introduction

The Transformer architecture (Vaswani et al., 2017) initially proposed for machine translation has become almost ubiquitous for many Machine Learning tasks. Transformer based architectures (Devlin et al., 2018; Lewis et al., 2019) are used to develop state-of-the-art solution in a variety of fields, ranging from Natural Language Processing (Sun et al., 2020; Devaraj et al., 2022) to Computer Vision (Dosovitskiy et al., 2020; Patrick et al., 2021), Audio Signal Processing (Radford et al., 2023) and image/video synthesis (Ding et al., 2022; Ge et al., 2022). Recently, Large Language Models (Brown et al., 2020) became capable of understanding and producing human-like text, leading to the advent of powerful conversational agents (Touvron et al., 2023b; Chiang et al., 2023; Ouyang et al., 2022). Besides capable of engaging in human-like conversations, due to the huge amounts of pre-training and fine-tuning data, Large Language Models (LLMs) obtain state-of-the-art results on several tasks (OpenAI, 2023).

Nevertheless, especially for highly specialized domains, there is still a need for custom models and methods. As such, legal (Chalkidis et al., 2020b; Shao et al., 2020; Masala et al., 2021; Niklaus and Giofré, 2022; Cui et al., 2023), medical (Lee et al., 2020; Rasmy et al., 2021; Liu et al., 2022), chemical (Chithrananda et al., 2020; Ahmad et al., 2022) or financial (Yang et al., 2020; Hillebrand et al., 2022; Wu et al., 2023; Yang et al., 2023) models have been proposed for a variety of languages.

In this work we investigate how to effectively process the long documents in the legal domain for a low-resource language (Romanian). We experiment with four different datasets, provided by a one of the top banks in Romania, from two different sources. We are, to the best of our knowledge, the first to prove that SLED (Ivgi et al., 2023) encoding applied on long documents for the legal judgement prediction tasks significantly improves performance compared to baseline methods. This is especially important for low-resource languages, such as Romanian, where language-specific LLMs with long-context support are not yet available and existing multi-language LLMs have low performance, at least for Romanian as this study demonstrates.

The main takeaways from the experiments are that: 1) encoding long documents with SLED can provide an important increase of performance, 2) multi-lingual LLMs are currently under-performing on LJP in Romanian both on smaller and larger documents.

2. Related Work

Transformer (Vaswani et al., 2017) architectures use self-attention as a central component. This mechanism connects all tokens in a sequence in a graph-like manner, using a relatedness pooling operation. While powerful, self-attention comes at a great cost as it has a quadratic complexity with the input length. As documents in the legal domain can be very long, scaling the self-attention to such documents quickly becomes infeasible.

Therefore, a great amount of work has been done to address this limitation. One such category of solutions tries to reduce the quadratic complexity of the self-attention mechanism by restricting the number of tokens a particular token can attend to. In sparse attention, each token can and is influ-

SIGUL2024 Workshop, pages 126–132

enced by a constant number of tokens, based on fixed (Child et al., 2019; Ainslie et al., 2020; Zaheer et al., 2020; Beltagy et al., 2020) or learned patterns (Kitaev et al., 2020; Roy et al., 2021). Usually, a small constant number of global tokens (attending all the other tokens) are kept at each layer.

Longformer (Beltagy et al., 2020) makes use of dilated sliding windows enabling long-range coverage while keeping sparsity. This is accomplished by having gaps in the attention patterns, increasing them as the model goes deeper. Accordingly, lower levels have strong local patterns while higher levels are capable of modeling long-range interactions. Finally, global attention is added on a small number of fixed input locations.

Instead of trying to increase the effective sequence length, SLED (Ivgi et al., 2023) proposes an efficient way of splitting the text into smaller blocks with partial overlap to allow longer sequences to be encoded. This mechanism is akin to local attention, as "full" self-attention is applied in each block. We adapt this mechanism for classification tasks, generating a representation for each token, representations that are further aggregated and fed to a decision layer.

Transformer-based models already assist legal practitioners on a multitude of tasks such as judgement prediction (Chalkidis et al., 2019a; Huang et al., 2021), information extraction (Chen et al., 2020; Hendrycks et al., 2021) or text classification (Chalkidis et al., 2019b, 2020a). Popular benchmarks devised for the legal domain (Chalkidis et al., 2021b; Niklaus et al., 2023) usually contain long documents, beyond the maximum length of standard BERT-like models. Popular approaches (Niklaus et al., 2022) split a document into equal-length blocks and encode them separately. All the obtained embeddings are further fed into another Transformer, followed by a max-pooling operation, thus obtaining an embedding for the document. This method first builds context-unaware paragraph representations that are further contextualized at paragraph level by the second stage Transformer.

3. Experimental Setup

3.1. Datasets

All datasets that we employ stem from Romanian civil cases in which clients sue a banking institution. Given the client's plea the task is to determine the outcome of the case. We treat this task as a binary classification task (win for client or bank). In this work we use two sources that contain different types of documents for the cases. The first data source is a collection of banking cases that took place between 2010 and 2018. We will further refer to this corpus as **BankingCases**. Each case contains the summary of the plaintiff written by the judge presiding over the case. In most cases, the judge restructures and rewrites the original arguments, even distorting some arguments to make the ruling more convincing. While this adds a certain bias and does not represent a realistic use case, using such data as an intermediate finetuning dataset greatly increases performance on real-world scenarios (Masala et al., 2021).

Finally, we collect a set of real-world cases (cases that contain raw pleas as opposed to summaries), **BRDCases** provided by the juridical department of bank BRD Group Societe Generale. Compared to BankingCases, these documents represent the plaintiff's raw plea, a collection of requests, proofs and other relevant documents. We pass all documents through a specialized OCR in Romanian juridical domain and anonymize personal information. For this reasons, the dataset contains less structured data, longer documents and more noise stemming, in part, due to the nature of the OCR extraction process.

From both sources we extract two common types of cases of interest to the banking domain, namely administration fee litigations (ADM) and enforcement appeals (ENF). In Table 1 we present detailed statistics for each dataset employed in this work. Note the large discrepancy between both the number of samples and the length of each case between BankingCases and BRDCases. In the real-world setting (BRDCases) we have extremely long texts, very few samples and in the case of ENF rather unbalanced data. For all cases, we automatically extract the year and the county where the case was filed. We inject this information in the form of one-hot encoding after the Transformer output, before the final decision and we further refer to it as handcrafted features.

To summarize, we collect datasets from two sources. The first dataset (BankingCases) contains a set of cases where the input is represented by the summary of arguments of both sides provided by the judge presiding the case at the end of the trial. The second dataset (BRDCases) contains a set of real-world argumentation of the plaintiff submitted to the court at the beginning of the trial, in exactly the same format they are received by the legal department of the bank. This means that for BRDCases, the input contains only the arguments of one side (i.e. the plaintiff), consists of much longer documents that come in the form of scanned files that need to be digitized. Our main goal is to provide an efficient automated method for predicting the outcome of a case in this real-world scenario. Such a method allows legal teams to efficiently assign resources, filtering out *unwinnable* cases.

Data source	Size	Class balance	jurBERT #tokens	Llama2 #tokens
BankingCases ADM	14367	1.59:1	2201 / 1161	3115 / 1684
BankingCases ENF	15044	1.51:1	2374 / 1225	3561 / 1874
BRDCases ADM	236	1.11:1	14280 / 10952	24047 / 17358
BRDCases ENF	90	3.10:1	6536 / 4270	10601 / 6912

Table 1: Dataset statistics - for number of tokens, the mean and median are shown for each tokenizer.

3.2. Models and Approaches

We employ a variety of methods to adapt the jurBERT model (Masala et al., 2021) to handle texts longer than 512 tokens. The first and simplest method is to make jurBERT process more than one block of 512 tokens in parallel. Therefore we experiment with the first and last 512 tokens of a document (denoted as *2*512*); similar for the first, middle, and last 512 tokens (denoted as *3*512*). Aggregating results from multiple blocks of a document is done by concatenating the [CLS] token representation for each block. This approach allows for handling longer documents, does not add a lot of complexity, and keeps the running time low. However, it is a rather rudimentary approach as it treats different blocks completely independently as there is no self-attention between blocks.

Next, we build Longformer versions of jurBERT, increasing the maximum sequence length and adapting the attention mechanism. This effectively increases the maximum sequence length of the model, and we experiment with sizes up to 4096. We also adapt SLED (Ivgi et al., 2023) input preprocessing for our task (just dropping the decoder part): we split the text into 32 chunks of 256 tokens (with a symmetric left-right overlap of 32 tokens each). Thus we obtain a representation for each token, followed by a max-pooling operation and the final decision. We found max-pooling to significantly outperform mean-pooling by over 10 points in mean AUC. Applying mean-pooling on such long sequences dilutes the content and the strong arguments making the final decision harder. Conversely, max-pooling works more as a focusing lens, making it better suited for the task as hand.

Recent LLMs are already pretrained using large contexts. Llama2 (Touvron et al., 2023b) is multilingual LLM with a context length of 4096, while the Romanian Okapi (Lai et al., 2023), a version of Llama (Touvron et al., 2023a), shares the same maximum sequence length. We finetune 7B variants of both Llama2 and Okapi using a classification framework (i.e. classification head on top of the last token) coupled with LORA (Hu et al., 2021) for computational efficiency.

Previous work has shown that Transformer-based solutions outperform several other approaches such as LSTMs, CNNs or SVMs with String Kernels (Lodhi et al., 2002) for Romanian legal judgement prediction in a very similar setting (Masala et al., 2021). Compared to the datasets used by Masala et al. (2021), we have collected a larger set of real-world cases and we pre-process them with several tools for the Romanian language that have shown an improvement in accuracy (i.e. a specialized Romanian juridical domain OCR extractor, a personal identifiable information anonymizer and a Romanian diacritics restoration tool). Overall, our real-world data is greater in size, more diverse and less noisy.

For these reasons, in this work we decide to limit our experiments to the best performing Transformer-based models from Masala et al. (2021) as a baseline and to show a significant improvement over them with long-context support.

3.3. Training Setup

Each model is trained using 5-fold cross-validation, over a maximum of 10 epochs. After each epoch, we save the AUC on the current "test" split and select the final result as the highest mean (over all folds) AUC for each epoch. Due to comutational limits, for BankingCases we take only one run, while for BRDCases we run each model 3 times (forp a total of 15 runs).

Note that all experiments on BRDCases are using models that are first finetuned on BankingCases sharing the same model architecture and hyperparameters. For computational reasons we finetune Llama2 and Okapi models on BankingCases using only a sequence length of 1024.

4. Results and Discussions

The results on BankingCases are presented in Table 2 and Table 3. The top part of the tables contain results for vanilla and Longformer variants of jurBERT. In the middle section of the tables the SLED alternative is introduced, while in the last section results using LLMs are showed. Note that results in the bottom part of tables do not use handcrafted features.

The first thing to note is the strong performance of the jurBERT baseline with a maximum sequence length of 512. For the Longformer variants, we believe their lack of performance is due to the limited training data (15k total, 12k training samples) that does not allow the model to properly learn how

Model	Seq Len	Mean AUC	Std AUC
jurBERT	512	78.20	0.56
jurBERT	2*512	**78.37**	1.05
jurBERT	3*512	78.50	1.16
jurBERT†	1024	74.27	0.73
jurBERT†	2048	70.65	3.02
jurBERT†	4096	67.20	0.92
jurBERT	32*256‡	67.57	0.66
jurBERT*	512	78.13	1.02
Llama2*	1024	69.88	0.79
Okapi*	1024	69.66	1.20

Table 2: Results on BankingCases ADM. * marks models that do not use handcrafted features, † marks Longformer variants and ‡ marks SLED input. We mark the top performer with **bold**.

Model	Seq Len	Mean AUC	Std AUC
jurBERT	512	75.26	0.56
jurBERT	2*512	**78.57**	0.42
jurBERT	3*512	77.93	0.59
jurBERT†	1024	66.76	3.36
jurBERT†	2048	56.33	5.26
jurBERT†	4096	54.24	2.56
jurBERT	32*256‡	78.03	0.76
jurBERT*	512	75.08	0.47
Llama2*	1024	65.03	2.19
Okapi*	1024	64.11	2.16

Table 3: Results on BankingCases ENF. Notations are the same as in Table 2.

to handle longer sequences. In the case of BankingCases, as documents are basically summaries written by judges, in most cases the strongest argument in favor of the final ruling is present in the first part of the document. This is in stark contrast with arguments of the plaintiff where the order and even the quality of documents is not always "best first". Understanding, validating, and ranking such arguments requires highly specialized work that is done by the judge and represents the very essence of a juridical trial.

The rather limited training data problem is aggravated in the case of LLMs. Both Llama2 and Okapi are both general language models, not specialized in the legal domain. This is also clearly visible by the statistics about the number of tokens presented in Table 1. jurBERT uses a specialized vocabulary (in Romanian juridical domain) and therefore is much more efficient in encoding legal texts compared to the general multi-language vocabulary used by Llama2/Okapi models. Furthermore, both Llama2 and Okapi have been trained on very few texts in Romanian.

Interestingly, in a setting with extremely low number of documents that are also very long (BRDCases dataset), the hierarchy of models changes.

Model	Seq Len	Mean AUC	Std AUC
jurBERT	512	68.38	5.49
jurBERT	2*512	64.28	4.55
jurBERT	3*512	63.15	7.42
jurBERT†	1024	71.33	7.38
jurBERT†	2048	71.55	5.25
jurBERT†	4096	71.56	5.38
jurBERT	32*256‡	**72.71**	5.99
jurBERT*	512	62.73	4.82
Llama2*	1024	63.60	6.93
Okapi*	1024	61.35	6.74

Table 4: Results on BRDCases ADM. Notations are the same as in Table 2.

Model	Seq Len	Mean AUC	Std AUC
jurBERT	512	63.80	11.25
jurBERT	2*512	**69.63**	10.37
jurBERT	3*512	60.54	11.32
jurBERT†	1024	60.87	10.14
jurBERT†	2048	56.92	13.81
jurBERT†	4096	41.60	22.79
jurBERT	32*256‡	65.48	12.01
jurBERT*	512	60.53	8.16
Llama2*	1024	60.19	12.42
Okapi*	1024	63.56	11.22

Table 5: Results on BRDCases ENF. Notations are the same as in Table 2.

As seen in Table 4, processing longer sequences generates better results, with the SLED variant obtaining the best result. For enforcement appeals (Table 5), we find jurBERT with first and last 512 tokens yields the best performance. Also, due to very limited and unbalanced data (only 90 samples, with a 3.10:1 distribution) note the very high standard deviation values. In some extreme cases, the model is unable to learn on some folds, leading to extremely poor results (under 0.5 mean AUC). As for each fold only 72 samples are used for training and the evaluation is performed only on 18 samples, the standard deviation is very high for most models. For enforcement appeals there is a chance that relevant information could be present at the begging and end of the documents and this should be investigated. Nonetheless, SLED encoding still provides a good performance being the second best option for enforcement appeal cases.

5. Conclusions

In this work we investigated the applicability of language models on the task of Legal Judgement Prediction, in a low-resource language (i.e. Romanian). We proved that integrating longer sequences, especially using SLED-style encoding, allows for a better understanding of documents, leading in the end to a overall increase in performance in our low-resources and long-documents setting.

Experiments on four different datasets highlight the need for methods that allow language models to parse long sequences and specialized vocabularies. As seen in Table 1, a specialized vocabulary is more efficient in encoding such documents, effectively allowing more information to be processed under the same sequence length limit. But a long sequence length is not enough. Especially in the case of BRDCases, the more relevant dataset of the two as it represents a real-world scenario, a long context size does not guarantee a competitive performance with Longformer and Llama2/Okapi variants underperforming and SLED offering the only improvement. At the same time, LLMs trained on huge amounts of (multi-lingual) data still lag behind more specialized solutions in this low-resource setting.

6. Limitations and Ethical Statement

In this work, we employ legal judgement prediction mainly to help one of the sides in a trial, in this case the defendant (a bank) to understand its chances of winning a trial. We do not aim to substitute the juridical process and, at the same time, understand that having such a system might provide important additional information for the side using it.

Our work focuses on a low-resource language and uses (very) small datasets. Moreover, the BRDCases dataset might have some biases as it contains legal documents received from a single Romanian bank. Therefore, the results presented in the paper might not be relevant for other languages, might not transfer to different tasks or even data from other parties on the same task.

On the other hand, this scenario is very relevant and useful for the legal department of a large bank, and we consider that this scenario is of interested for other researchers working on real-world datasets and use-cases.

While legal documents contain personal identifiable (PII), we want to highlight that in our experiments PII data has been removed using an external API for Romanian. Again, we consider that this preprocessing is important to remove any spurious correlations and might also be relevant for other real-world use-cases.

7. Acknowledgements

This work was partially supported by a research grant from BRD Groupe Societe Generale.

8. Bibliographical References

Walid Ahmad, Elana Simon, Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. 2022. Chemberta-2: Towards chemical foundation models. *arXiv preprint arXiv:2209.01712*.

Joshua Ainslie, Santiago Ontanon, Chris Alberti, Vaclav Cvicek, Zachary Fisher, Philip Pham, Anirudh Ravula, Sumit Sanghai, Qifan Wang, and Li Yang. 2020. Etc: Encoding long and structured inputs in transformers. *arXiv preprint arXiv:2004.08483*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. 2019a. Neural legal judgment prediction in english. *arXiv preprint arXiv:1906.02059*.

Ilias Chalkidis, Manos Fergadiotis, Sotiris Kotitsas, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020a. An empirical study on large-scale multi-label text classification including few and zero-shot labels. *arXiv preprint arXiv:2010.01653*.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020b. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2019b. Large-scale multi-label text classification on eu legislation. *arXiv preprint arXiv:1906.02192*.

Ilias Chalkidis, Manos Fergadiotis, Dimitrios Tsarapatsanis, Nikolaos Aletras, Ion Androutsopoulos, and Prodromos Malakasiotis. 2021a. Paragraph-level rationale extraction through regularization: A case study on european court of human rights cases. *arXiv preprint arXiv:2103.13084*.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Martin Katz, and Nikolaos Aletras. 2021b. Lexglue: A benchmark dataset for legal language understanding in english. *arXiv preprint arXiv:2110.00976*.

Yanguang Chen, Yuanyuan Sun, Zhihao Yang, and Hongfei Lin. 2020. Joint entity and relation extraction for legal documents with legal feature enhancement. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1561–1571.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*.

Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. 2020. Chemberta: large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*.

Jiaxi Cui, Zongjian Li, Yang Yan, Bohua Chen, and Li Yuan. 2023. Chatlaw: Open-source legal large language model with integrated external knowledge bases. *arXiv preprint arXiv:2306.16092*.

Ashwin Devaraj, William Sheffield, Byron C Wallace, and Junyi Jessy Li. 2022. Evaluating factuality in text simplification. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2022, page 7331. NIH Public Access.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. 2022. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *Advances in Neural Information Processing Systems*, 35:16890–16902.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. 2022. Long video generation with time-agnostic vqgan and time-sensitive transformer. In *European Conference on Computer Vision*, pages 102–118. Springer.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. *arXiv preprint arXiv:2103.06268*.

Lars Hillebrand, Tobias Deußer, Tim Dilmaghani, Bernd Kliem, Rüdiger Loitz, Christian Bauckhage, and Rafet Sifa. 2022. Kpi-bert: A joint named entity recognition and relation extraction model for financial reports. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 606–612. IEEE.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Yunyun Huang, Xiaoyu Shen, Chuanyi Li, Jidong Ge, and Bin Luo. 2021. Dependency learning for legal judgment prediction with a unified text-to-text transformer. *arXiv preprint arXiv:2112.06370*.

Maor Ivgi, Uri Shaham, and Jonathan Berant. 2023. Efficient long-text understanding with short-text models. *Transactions of the Association for Computational Linguistics*, 11:284–299.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A Rossi, and Thien Huu Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. *arXiv preprint arXiv:2307.16039*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Ning Liu, Qian Hu, Huayun Xu, Xing Xu, and Mengxin Chen. 2022. Med-bert: A pretraining framework for medical records named entity recognition. *IEEE Transactions on Industrial Informatics*, 18(8):5600–5608.

Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. 2002. Text classification using string kernels. *Journal of machine learning research*, 2(Feb):419–444.

Mihai Masala, Radu Cristian Alexandru Iacob, Ana Sabina Uban, Marina Cidota, Horia Velicu, Traian Rebedea, and Marius Popescu. 2021. jurbert: A romanian bert model for legal judgement prediction. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 86–94.

Joel Niklaus and Daniele Giofré. 2022. Budgetlongformer: Can we cheaply pretrain a sota legal language model from scratch? *arXiv preprint arXiv:2211.17135*.

Joel Niklaus, Veton Matoshi, Pooja Rani, Andrea Galassi, Matthias Stürmer, and Ilias Chalkidis. 2023. Lextreme: A multi-lingual and multi-task benchmark for the legal domain. *arXiv preprint arXiv:2301.13126*.

Joel Niklaus, Matthias Stürmer, and Ilias Chalkidis. 2022. An empirical study on cross-x transfer for legal judgment prediction. *arXiv preprint arXiv:2209.12325*.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Mandela Patrick, Dylan Campbell, Yuki Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and Joao F Henriques. 2021. Keeping your eye on the ball: Trajectory attention in video transformers. *Advances in neural information processing systems*, 34:12493–12506.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR.

Laila Rasmy, Yang Xiang, Ziqian Xie, Cui Tao, and Degui Zhi. 2021. Med-bert: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *NPJ digital medicine*, 4(1):86.

Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. 2021. Efficient content-based sparse attention with routing transformers. *Transactions of the Association for Computational Linguistics*, 9:53–68.

Yunqiu Shao, Jiaxin Mao, Yiqun Liu, Weizhi Ma, Ken Satoh, Min Zhang, and Shaoping Ma. 2020. Bert-pli: Modeling paragraph-level interactions for legal case retrieval. In *IJCAI*, pages 3501–3507.

Lichao Sun, Congying Xia, Wenpeng Yin, Tingting Liang, Philip S Yu, and Lifang He. 2020. Mixup-transformer: dynamic data augmentation for nlp tasks. *arXiv preprint arXiv:2010.02394*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. 2023. Fingpt: Open-source financial large language models. *arXiv preprint arXiv:2306.06031*.

Yi Yang, Mark Christopher Siy Uy, and Allen Huang. 2020. Finbert: A pretrained language model for financial communications. *arXiv preprint arXiv:2006.08097*.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297.