CS230

Improving Language Model Performance on the United States Uniform Bar Exam

Lucia Zheng
Department of Computer Science
Stanford University
zlucia@stanford.edu

Abstract

This work explores methods for improving language model performance on the United States Uniform Bar Exam. Achieving strong machine performance on this task has implications for our ability to provide basic legal services at scale through AI systems because it signals that language models have the capacity to interpret and reason about the law at a comparable level to a human legal professional. We experimented with various pre-training and fine-tuning processes on BERT based Transformer models and found that unsupervised legal domain-specific pre-training, combined with fine-tuning on an adjacent legal task and further fine-tuning on the target task, resulted in the highest performing model, which achieved classification accuracy of 0.390.

1 Introduction

In this work, we aim to improve language model performance on a legal NLP task, multiple choice question answering on the Multistate Bar Examination (MBE) section of the United States Uniform Bar Exam (UBE). Demonstrating that a language model is capable of learning legal knowledge and reasoning comparable to a human lawyer provides evidence that AI systems could be utilized to expand critical legal services, such as claims adjudication, a significant result from an access-to-justice perspective.

The input for this task is the text from historical bar exam answer keys for the multiple choice MBE sections of bar exams administered from 1972 to 1998, which have been publicly released by the creators of the UBE, the National Conference of Bar Examiners (NCBE). The output for the task is the prediction of the correct answer label for an example. Only a limited set of historical bar exam answer keys have been released to the public, so the dataset contains 934 examples, considered small within a deep learning framework. Thus, a major challenge in this task is addressing the limitations of language models on few-shot text classification for a complex legal reasoning task.

2 Related Work

Some work has been done on question answering tasks for professional legal exams. Wyner, Fawei, and Pan [1] used the Excitement Open Platform (EOP) for textual entailment to predict answers to questions from the multiple-choice MBE section of the US Uniform Bar Exam. They found that the EOP can identify wrong answers (non-entailment) with a high F1 score, but it performs poorly in identifying the right answer (entailment).

Drawing from more recent advances in Transformer language models, several researcher have done work on applying the transfer learning paradigm of BERT [2] to language-based tasks in various

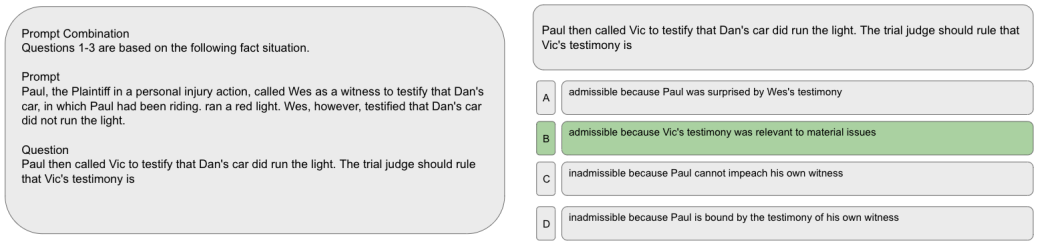
CS230: Deep Learning, Autumn 2020, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

knowledge domains. BERT makes use of Transformer, an attention mechanism that learns contextual relations between words, to train bidirectional word embeddings. In an unsupervised pre-training step, BERT is trained on large unlabeled text corpora, to learn patterns from language. In a supervised fine-tuning step, BERT can then be initialized with pre-trained parameters and further trained for a supervised task on a labeled dataset efficiently, transferring general learnings from the first step. Devlin et al. [2] showed that BERT, pre-trained on BookCorpus and English Wikipedia, and fine-tuned on the General Language Understanding Evaluation (GLUE) benchmark tasks, achieves strong performance. It has been shown by Lee et al. [3] and Beltagy, Lo, and Cohan [4] that domain-specific pre-training BERT can improve performance on downstream domain-specific language tasks.

This transfer learning paradigm has been applied to some common legal NLP tasks, such as classification and entity and relation extraction tasks. Many of these tasks are simple enough that current out-of-the-box Tranformer models, such as BERT initialized with pre-trained parameters, can achieve good results on these tasks with a few epochs of fine-tuning. For example, Elwany, Moore, and Oberoi [5] achieve an F1 score of 0.943 on a binary classification task for classifying the terms of a legal agreement as either "fixed" or "auto-renewing" using BERT with pre-trained parameters. However, since BERT achieves high performance for these tasks, it is difficult to distinguish the comparative advantage of legal domain-specific pre-training. We believe the bar exam task is more complex and thus, may be a more difficult task for BERT and may offer more illustrative results on the performance improvements achievable under domain-specific pre-training. Additionally, since the historical dataset is limited in size, this task gives us the opportunity to explore techniques for improving fine-tuning on domain-specific downstream tasks in few-shot learning contexts.

3 Dataset and Features

The historical MBE dataset has 934 examples. Each example in the MBE dataset consists of a prompt combination, prompt, question, and 4 multiple-choice answers, with an answer label for the correct answer. In Figure 1, we show an example from the MBE dataset.



(a) Prompt/Question

(b) Answer Choices (correct label in green)

Figure 1: Example from MBE Dataset

One important procedural step we took in splitting the dataset was grouping questions with a common prompt context together, to prevent the model from potentially seeing prompt contexts in the validation or test set during training, since different questions in the dataset can share the same prompt contexts based on the design of the MBE.

The MBE dataset was randomly split into a training and test set, with proportions 70/15. The training set was then randomly split into 5-folds for cross-validation.

4 Methods

To build the models described in this section, we used the HuggingFace Transformers library [6], which provides a PyTorch implementation of the pre-trained BERT model from Devlin et al. [2].

The BERT language model is a trained Transformer Encoder stack. The distinctive feature of the encoder is the self-attention layer, introduced in Vaswani et al. [7]. As the model processes each position in the input sequence, self-attention allows it to look at other positions in the input sequence that contribute to the understanding of the word at the current position to produce a better encoding for each word. Since the encoders utilize input masking to condition on both the left and right context

of input in the pre-training step, BERT learns deep bidirectional representations from pre-training on unlabeled text. As a result, pre-trained BERT with just one additional output layer has been shown to perform well fine-tuned on a wide range of tasks in Devlin et al. [2]. Somewhat unique from other language models, as opposed to transferring just sentence embeddings, BERT transfers all parameters to initialize downstream task model parameters and trains all of the parameters of the model in the fine-tuning step.

4.1 Fine-tuning BERT for MBE Multiple Choice Target Task

BERT leverages semi-supervised learning in a two step training process, combining unsupervised training on large unlabeled text corpora in a pre-training step with supervised training on a downstream task with a labeled dataset, which is often smaller in size, in a fine-tuning step. For the baseline model, we use a pre-trained BERT models from Devlin et al. [2], BERT Base (uncased). We'll refer to this model as (1) **BERT-Base**.

In order to adapt the BERT architecture to fine-tune on a multiple choice task, we built a processor that applies an input transformation to each example in the MBE dataset, so that for each answer choice i of the $N = 4$ answer choices, the processor creates an input string with the text of the Prompt, and Question as *Context* and the text of the Answer Choice as *Answer i*, with special start (*Start*), delimiter (*Delim*), and end (*Extract*) tokens, as depicted in Figure 2.

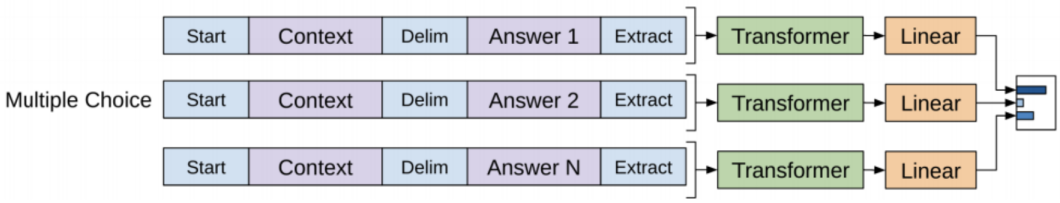


Figure 2: Fine-tuning BERT for Multiple Choice Target Task (Radford et al. [8])

These inputs are then fed into the *Transformer* in Figure 2, which is (1) BERT-Base for our baseline. This is followed with a *Linear* layer on top of the pooled output (embedding of start token) that outputs 4 scalar values corresponding to the 4 answer choices. Then, a softmax is taken over the 4 scalar values, and the model outputs a vector of prediction probabilities for the 4 answer labels. Cross-entropy loss is used as the loss function for the multi-class classification head.

$$L(y, \hat{y}) = -\sum_{i} y_i \log \hat{y}_i \tag{1}$$

4.2 Legal Domain-Specific Pre-training

It has been shown by Lee et al. [3] and Beltagy, Lo, and Cohan [4] with BioBERT and SciBERT that pre-training BERT on corpora that is domain-specific to the downstream supervised task leads to performance gains in the natural sciences. We pre-train BERT on legal domain-specific corpora to test whether this improves performance on the bar exam task.

We pre-trained BERT on a legal corpus consisting of US case law from 1965 onwards. We'll refer to this model as (2) **LegalBERT**. In the unsupervised pre-training step, we followed the method described in Lee et al. [3] for training BioBERT closely, initializing (2) LegalBERT with the vocabulary and parameters from pre-trained (1) BERT-Base and training for 1M training steps on the case law corpus on the two pre-training step tasks for BERT, masked LM (MLM) and next sentence prediction (NSP), as in Devlin et al. [2]. Whole word masking, instead of token masking, was applied to the input text, as in Cui et al. [9], since the citation format in case law can separate words into several tokens. The fine-tuning structure described in Section 4.1 is used to fine-tune (2) LegalBERT for the downstream MBE multiple choice task.

4.3 Intermediate Legal Holding Multiple Choice Task Fine-tuning

It has been shown by Pruksachatkun et al. [10] that common sense reasoning tasks that require inference tend to make good intermediate tasks for smaller downstream target tasks. Common sense reasoning tasks are intended to require the model to go beyond pattern recognition and use "common sense" or world knowledge to make inferences. This term encompasses tasks like multiple choice question answering and pronoun entailment.

Motivated by this result, we constructing a dataset for a legal holding multiple choice task, where given a judicial decision context, a language model predicts the legal holding, the legal principle derived from a judicial decision, from 5 answer choices. This dataset is larger than the MBE dataset, with 40,000 total examples, randomly split into train, validation, and test sets with proportions 80/10/10. Here, we propose a model with an intermediate fine-tuning step on the legal holding multiple choice task, which we'll refer to as (3) **LegalBERT-Holding**. (3) LegalBERT-Holding was initialized with the vocabulary and parameters from pre-trained (1) BERT-Base and fine-tuned on the intermediate legal holding multiple choice task for 1 epoch, learning rate of 1e-5, batch size of 8, and maximum sequence length of 128. These hyperparameters were established by a hyperparameter search done for a series of legal NLP tasks outside of this work. In a subsequent fine-tuning step, (3) LegalBERT-Holding was fine-tuned for the target MBE multiple choice task. The fine-tuning for the intermediate and target multiple choice tasks were done using the fine-tuning structure described in Section 4.1, with cross-entropy loss (1) used as the loss function for the multi-class classification head on both fine-tuning tasks.

4.4 Unsupervised LM Fine-tuning on MBE Dataset

Howard and Ruder [11] an unsupervised LM fine-tuning step on the target task can increase performance because the data of the target task will likely come from a different distribution compared to the data for pre-training, regardless of how diverse the general-domain data used for pre-training is. This additional step of fine-tuning has been shown by Howard and Ruder to perform well on text classification tasks such as sentiment analysis on movie reviews. Though they introduce several novel methods for more controlled unsupervised LM fine-tuning, we try only a full unsupervised LM fine-tuning step, where we trained models on an unsupervised mask LM (MLM) task with token masking on the unlabeled text from the MBE dataset, excluding the answer choice labels.

We daisy chain this step into the models described in Section 4.2 and Section 4.3 to create two more models: (4) **LegalBERT-MLM** and (5) **LegalBERT-Holding-MLM**. (4) LegalBERT-MLM is initialized with the vocabulary and parameters of (2) LegalBERT and (5) LegalBERT-Holding-MLM is initialized with the vocabulary and parameters of (3) LegalBERT-Holding, Both (4) LegalBERT-MLM and (5) LegalBERT-Holding-MLM are further fine-tuned using the MLM task with token masking on the unlabeled text of the MBE dataset, excluding the test set. The MLM task masks tokens according to the procedure described in Devlin et al. [2] and computes loss on predictions for masked tokens using cross entropy loss (1).

5 Experiments/Results/Discussion

5.1 Hyperparameter Search

Due to constraints on GPU memory, we used a fixed batch size of 8 and maximum sequence length of 256 and selected the train/validation set for one fold to use for hyperparameter search. For similar reasons, in cross validating our results on all 5 folds, we used a fixed batch size of 4 and a maximum sequence length of 512.

We performed a hyperparameter search over hyperparameters recommended in Devlin et al. [2] for the number of epochs and the learning rate. Since the space of recommended hyperparameters was small and had been shown to work well across a wide range of language tasks, we chose to grid search exhaustively instead of using a uniform or logarithmic scale uniform random search. We implemented this hyperparameter search using Ray Tune. In early experiments, we found that several of the models were prone to overfitting on the small MBE dataset, so we increased dropout and weight decay parameters to increase regularization and added smaller hyperparameter settings to

the search space for the number of epochs and learning rate to allow for fewer or slower parameter updates.

Model	Number Epochs	Learning Rate	Dropout	Weight Decay
(1) BERT-Base	1, 2, 3, 4	5e-6, 1e-5, 2e-5, 5e-5	0.1	0.0
(2) LegalBERT	1, 2, 3, 4	5e-6, 1e-5, 2e-5, 5e-5	0.3	0.1
(3) LegalBERT-Holding	1, 2, 3, 4	5e-6, 1e-5, 2e-5, 5e-5	0.1	0.1
(4) LegalBERT-MLM	1, 2, 3, 4	5e-6, 1e-5, 2e-5, 5e-5	0.1	0.1
(5) LegalBERT-Holding-MLM	1, 2, 3, 4	5e-6, 1e-5, 2e-5, 5e-5	0.1	0.1

Table 1: Hyperparameter Search Space

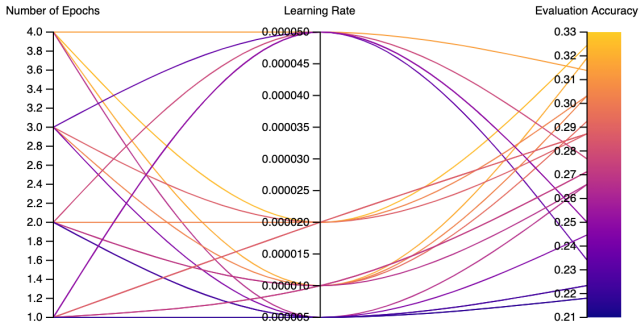


Figure 3: Hyperparameter Search Results for (5) LegalBERT-Holding-MLM

After performing hyperparameter search, we found that the hyperparameter setttings that achieved highest evaluation accuracy on the validation set for models (3) LegalBERT-Holding and (4) LegalBERT-MLM were still prone to overfitting, so we increased weight decay for both models. In Table 2, we list the model hyperparameters set for the reported results.

Model	Number Epochs	Learning Rate	Dropout	Weight Decay
(1) BERT-Base	3	5e-5	0.1	0.0
(2) LegalBERT	3	5e-5	0.3	0.1
(3) LegalBERT-Holding	3	5e-5	0.1	0.2
(4) LegalBERT-MLM	4	5e-5	0.1	0.2
(5) LegalBERT-Holding-MLM	4	2e-5	0.1	0.1

Table 2: Model Hyperparameters for Reported Results

5.2 Model Performance

The MBE dataset has a uniform label distribution, with the true labels for each of the 4 answer choices making up about 25% of the dataset. We would expect a naïve model that chooses an answer uniformly at random for each example to achieve around 25% classification accuracy. Since our dataset is class balanced and this task simulates taking the bar exam, for which performance is measured based on accuracy, we use classification accuracy as our metric for evaluating model performance. Note that for comparison, the accuracy threshold that is considered passing for a human test taker on the MBE section of the bar exam is 60%.

To measure the performance of each model, we trained and evaluated performance across the 5 folds to mitigate the effects of potentially high model performance variance at evaluation time due to split choice over the small MBE dataset on reported result.

Since we plan to develop this model in future work, we report classification accuracy as the 5-fold cross-validated classification accuracy, instead of the test set classification accuracy, in Table 3. Unlike a typical multi-class classification task, the class labels (A, B, C, D) for the MBE multiple choice task do not have underlying meaning since the correct answer choice is labeled arbitrarily, so

we do not use a traditional confusion matrix visualization. Instead, in the Appendix A, we include additional visualizations of the attention layers of the models, which display the learned dependencies between words in example inputs.

Model	Accuracy
(1) BERT-Base	0.271
(2) LegalBERT	0.245
(3) LegalBERT-Holding	0.390
(4) LegalBERT-MLM	0.375
(5) LegalBERT-Holding-MLM	0.360

Table 3: MBE Multiple Choice Task Results

The best model, (3) LegalBERT-Holding on the MBE dataset, achieved classification accuracy of 0.390. This accuracy is a significant gain over the (1) BERT-Base model with no domain-specific pre-training or additional fine-tuning on other tasks before fine-tuning on the target MBE multiple choice task. (2) LegalBERT did not perform better than (1) BERT-Base on either the MBE dataset or the augmented dataset, but this may be caused by the dropout used to prevent overfitting; it is likely not the case in general that legal domain-specific pre-training does not improve performance on legal reasoning language tasks.

Though (5) LegalBERT-Holding-MLM had worse classification accuracy than (3) LegalBERT-Holding and (4) LegalBERT-MLM, one interesting result was that (5) had much better convergence behavior than (3) and (4), as seen in Figure 4. Even applying stronger regularization techniques to models (3) and (4) beyond those set by the hyperparameters in Table 2, we were unable to prevent overfitting displayed through increased evaluation loss over training time steps. On the other hand, model (5), which achieved a significant 8.9 percentage point accuracy margin (1) BERT-Base, had consistently good convergence behavior across all 5 folds, signalling that it may have better generalization capacity than model (3), which achieved a 11.9 percentage point accuracy margin (1) BERT-Base on our specific MBE dataset.

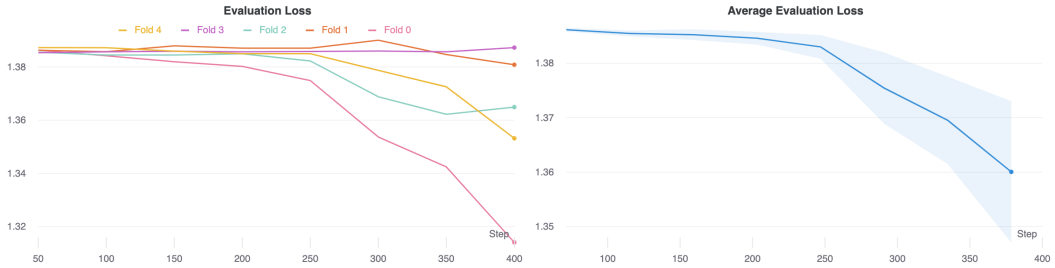


Figure 4: (5) LegalBERT-Holding-MLM Convergence Behavior

6 Conclusion/Future Work

To conclude, the model with the highest performance was (3) LegalBERT-Holding, a model initialized with the vocabulary and parameters of (1) BERT-Base, pre-trained an unlabeled legal text corpus consisting of case law from 1965 onwards, fine-tuned with an intermediate legal holding multiple choice task (similar to the target task), and finally fine-tuned on the target MBE multiple choice task. (3) LegalBERT-Holding achieved classification accuracy of 0.390.

One significant issue we ran into was the tendency for the BERT based models, with large parameter counts and many layers, to overfit on the relatively small MBE dataset. In future work, we would like to work on increasing the size of the dataset to mitigate this problem. One possible way to access more training data beyond the publicly available historical bar exam answer key bank could be to aggregate bar exam answer keys from practice materials from bar exam preparation courses, such as BARBRI. Another solution might be to artificial generate additional training data from the existing data by adding "noise" through techniques like random word replacement with synonyms.

Since (5) LegalBERT-Holding-MLM achieved relatively high performance, classification accuracy of 0.360, and more promising convergence behavior than (3) LegalBERT-Holding, we would also be interested in working to improve the performance of this model by using more controlled techniques for updating parameters of the model in the unsupervised LM fine-tuning step as presented in Howard and Ruder [11], like discriminative fine-tuning, which allows different layers of the model to be tuned with different rates, and slanted triangular learning rates, which decays over training time steps.

7 Contributions

This work is done in collaboration with the Stanford Regulation, Evaluation, and Governance Lab. We are grateful to be advised by Prof. Daniel Ho on this work.

References

[1] A. Z. Wyner, B. J. Fawei, and J. Z. Pan, "Passing a usa national bar exam: A first corpus for experimentation," in *LREC 2016, Tenth International Conference on Language Resources and Evaluation*, LREC, 2016.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2019. arXiv: 1810.04805 [cs.CL].

[3] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "Biobert: A pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, J. Wren, Ed., Sep. 2019, ISSN: 1460-2059. DOI: 10.1093/bioinformatics/btz682.

[4] I. Beltagy, K. Lo, and A. Cohan, *Scibert: A pretrained language model for scientific text*, 2019. arXiv: 1903.10676 [cs.CL].

[5] E. Elwany, D. Moore, and G. Oberoi, *Bert goes to law school: Quantifying the competitive advantage of access to large legal corpora in contract understanding*, 2019. arXiv: 1911.00473 [cs.CL].

[6] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Huggingface's transformers: State-of-the-art natural language processing," *ArXiv*, vol. abs/1910.03771, 2019.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, *Attention is all you need*, 2017. arXiv: 1706.03762 [cs.CL].

[8] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, *Improving language understanding by generative pre-training*, 2018.

[9] Y. Cui, W. Che, T. Liu, B. Qin, Z. Yang, S. Wang, and G. Hu, *Pre-training with whole word masking for chinese bert*, 2019. arXiv: 1906.08101 [cs.CL].

[10] Y. Pruksachatkun, J. Phang, H. Liu, P. M. Htut, X. Zhang, R. Y. Pang, C. Vania, K. Kann, and S. R. Bowman, *Intermediate-task transfer learning with pretrained models for natural language understanding: When and why does it work?* 2020. arXiv: 2005.00628 [cs.CL].

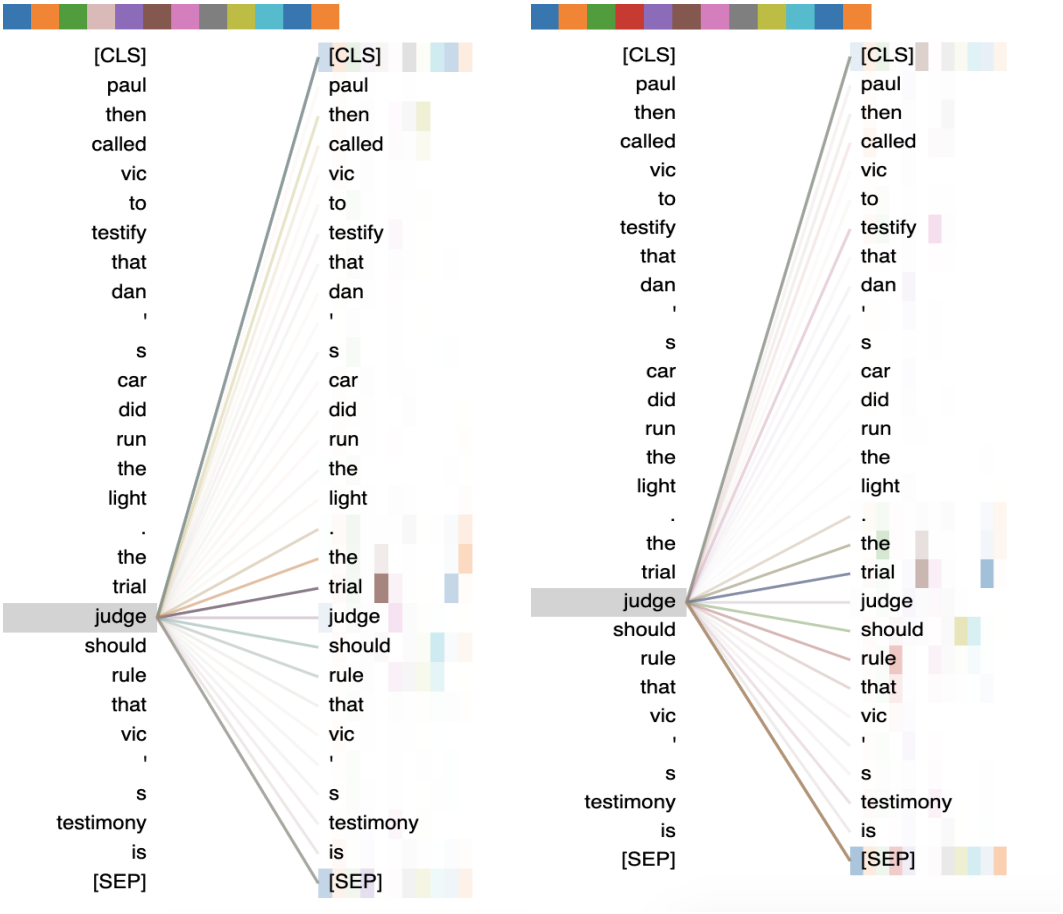
[11] J. Howard and S. Ruder, *Universal language model fine-tuning for text classification*, 2018. arXiv: 1801.06146 [cs.CL].

[12] J. Vig, "A multiscale visualization of attention in the transformer model," *arXiv preprint arXiv:1906.05714*, 2019.

A Model Attention Layers

The visualizations for the model attention layers in this section were created using BertViz [12].

For the example input question: *Paul then called Vic to testify that Dan's car did run the light. The trial judge should rule that Vic's testimony is*, comparing the visualizations of the Layer 3 attention head for the (1) BERT-Base and (5) LegalBERT-Holding-MLM models, we can see (5) LegalBERT-Holding-MLM learned encodings that capture a comparatively better understanding of the strong relationship between the word *judge* and words such as *testify*, *rule*, and *testimony* than (1) BERT-Base.



(a) (1) BERT-Base

(b) (5) LegalBERT-Holding-MLM

Figure 5: Model Layer 3 Attention Heads

We also provide a model view of all of the layers and heads of the (5) LegalBERT-Holding-MLM model, with layers increasing from 0 to 11 vertically and heads increasing from 0 to 11 horizontally, on example input question: *Paul then called Vic to testify that Dan's car did run the light. The trial judge should rule that Vic's testimony is*, and input answer choice: *admissible because Vic's testimony was relevant to material issues*. We see that the model produces a rich array of attention patterns.

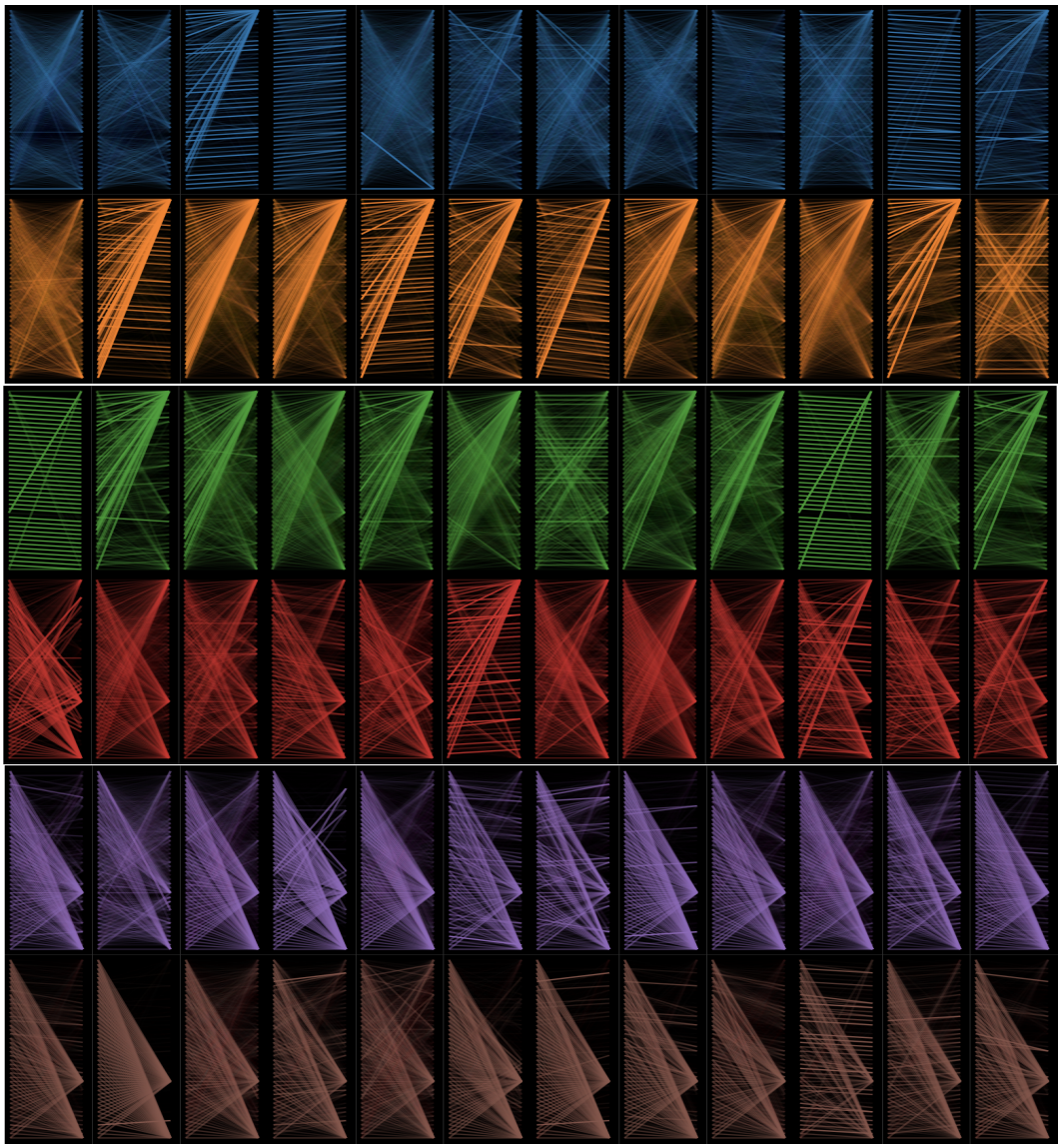


Figure 6: (5) LegalBERT-Holding-MLM Model Attention Layers 0-5

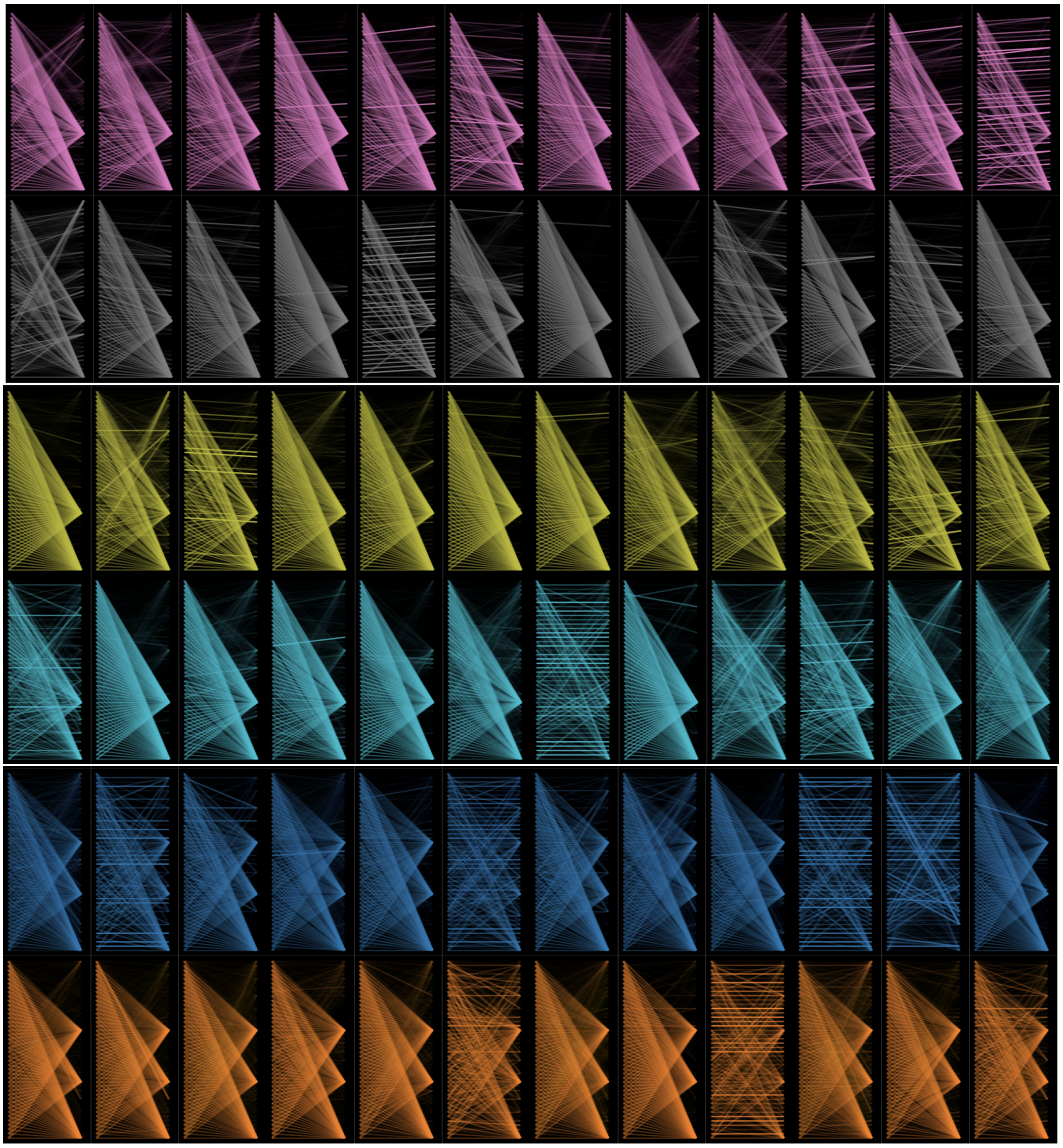


Figure 7: (5) LegalBERT-Holding-MLM Model Attention Layers 6-11