Legal-SBERT: Creating a Sentence Tranformer for the Legal Domain and Generating Data

Stanford CS224N Custom Project

Jayendra Chauhan
Department of Computer Science
Stanford University
jchauhan@stanford.edu

Abstract

Legal texts are characterized by a variety of differences from everyday language to the extent that legal language can be classified as a sublanguage. One of the differences between legal and everyday texts is that, in legal contexts, semantically significant words are used with extremely specific meanings (which can be entirely different from the words' everyday meanings). There have been some efforts to create BERT-based encoders that capture the nuanced meanings of legal words, but there exists no sentence transformer model specialized for the legal domain. We leverage an existing legal BERT model and the Sentence-BERT architecture in order to create a legal sentence transformer and name it Legal-SBERT. Furthermore, we demonstrate methods to finetune and test sentence transformers on legal-domain specific data, addressing the lack of legal data labeled suitably for sentence transormers, by generating data from a Large Language Model, adapting binary classification data, and adapting multi-class classification data. Our work indicates that the Legal-SBERT architecture presented leads to performance gains on legal NLP tasks of reasonable difficulty and furthers the discussion around when pretraining on legal corpora is useful for improving performance on legal NLP tasks.

1 Key Information to include

- Mentor:
- External Collaborators (if you have any): Zehua Li
- Sharing project:

2 Introduction

Individuals routinely suffer from the high barrier to and complexity of navigating the legal system. This has generated significant interest in creating AI models that can help individuals interpret and navigate laws relevant to them. One of the challenges in creating such models is that the legal language is significantly different from everyday language. Many important legal words have significantly different semantic meanings in legal and everyday contexts, to the extent that legal language can be classified as a sublanguage (Chalkidis et al., 2020).

Previous efforts at creating NLP models more suitable for the legal domain have been focused on fine tuning the BERT encoder with legal corpora Zheng et al. (2021). However, a legal sentence transformer has yet to be explored in the literature. Compared to encoders, sentence transformers have a significant time-complexity advantage in tasks that involve computing the similarity between two sentences. For example, the BERT-based sentence transformer introduced by Reimers and Gurevych (2019) can find the most similar pair of sentences in a collection of 10,000 sentences within 5 seconds, compared to the 65 hours that it would take BERT. However, sentence transformers are limited in that they require pairs of sentences, labeled with the sentences' relationship, for training and evaluation.

One of the challenges in creating legal-domain specific NLP transformers, including sentence tranformers, is the mixed literature surrounding whether or not pre-training on domain-specific corpora improves model performance. Perplexingly, some previous work has shown no meaningful improvements in performance on legal NLP tasks after pre-training models on legal domain-specfic corpora (Elwany et al., 2019). Therefore, although encoders pre-trained on legal domain specific data have made gains on legal NLP tasks, this has raised confusion around when and what types of pretraining are actually useful for tuning models to the legal domain (Zheng et al., 2021). Given the distinct vocabulary of the legal "sublanguage", we hypothesize that tuning a sentence transformer in a manner designed to improve the models' representations of words with distinct legal meanings will improve the embeddings created by the sentence.

Another challenge in creating a legal sentence transformer is the lack of labeled legal data suitable for evaluating a sentence transformer. In general, creating labeled legal NLP datasets is difficult because of the extremely high cost of hiring professionals knowledgeable about law. The ideal task for evaluating a sentence transformer is Semantic Textual Similarity (STS), which is the task of computing how similar two pieces of text are. However, STS evaluation, as performed in the most commonly used benchmarking toolkits like SentEval Conneau and Kiela (2018), requires gold-standard similarity annotations between sentence pairs, which don't exist for any legal datasets.

3 Related Work

Legal Encoders There are two notable efforts to create a legal encoder in the literature. Legal-BERT is a family of BERT based models adapted to the legal domain through pre-training on legal corpora that include court cases, contracts, and legislation (Chalkidis et al., 2020). Three different strategies were used to train the Legal-BERT models: (a) using BERT as-is, (b) further pre-training BERT on the gathered legal corpora, and (c) pre-training BERT from scratch on the gathered legal corpora. Chalkidis et al. (2020)'s results indicate that the performance on legal tasks from options (b) and (c) vary by task, but are superior to the results from option (a), indicating that the best strategy for adopting BERT to a domain is either additional pre-training or pre-training from scratch depending on the task. The second notable legal encoder is CaseLaw-BERT (Zheng et al., 2021). Zheng et al. (2021) tune BERT using additional pretraining on court opinions (which are called case law). The authors of CaseLaw-BERT find that many legal NLP tasks are too simple to be useful in discerning whether there are benefits to legal pretraining, and that legal pretraining shows benefits when the task is sufficiently difficult (Zheng et al., 2021).

SentenceTransformers The dominant family of models used to create sentence embeddings are SentenceTransformers. The original model is Sentence-BERT (SBERT), which was introduced by (Reimers and Gurevych, 2019). SBERT leverages either a Siamese or Triplet network structure between BERT encoders, choosing the number of encoders based on the input data and its labeling. Through its Siamese or Triplet network structure, SBERT finetunes the weights of, respectively, two or three pre-trained BERT encoders that have tied, identical weights in order to tune the BERT encoder to produce useful sentence embeddings. SBERT also uses a pooling strategy on the embeddings from the BERT encoders to compute fixed-size sentence embeddings, a concatenation strategy to combine the pooled vectors created from each pooling layer, and, as appropriate, applies a suitable loss function or classifier (such as Softmax). More recent SentenceTransformers share the same fundamental model architecture and differ only in their underlying encoders and fine-tuning datasets and training regimes. We leverage the same architecture as the SentenceTransformers.

4 Approach

Legal-SBERT In order to develop Legal-SBERT, a sentence transformer model with more useful embeddings for legal sentences, we re-implement the SBERT architecture originally defined in Reimers and Gurevych (2019) (see Figure 1) using Legal-BERT, a family of models based on the BERT architecture that are pre-trained and finetuned on legal corpora Chalkidis et al. (2020), as the underlying encoder. We use Legal-BERT as is but reimplement the Sentence Transformer architecture.

Pretrained Legal-SBERT We also implement Pretrained Legal-SBERT, which has the same architecture as Legal-SBERT but undergoes additional pretraining intended to teach the model to better distinguish between the legal and everyday meanings of words with important legal meanings. We select these words with important legal meanings by choosing 6 of the words identified by (Nyarko and Sanga, 2021) to have significantly different legal and everyday meanings. To generate sentences containing the legal and everyday meanings of these words, we generate data from the GPT 3.5 Large Language Model (LLM), exploring if LLMs can be of use in pretraining models.

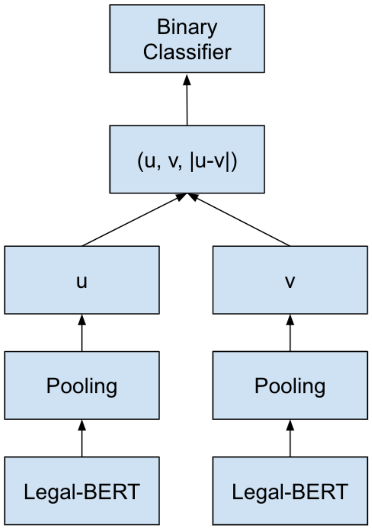

Figure 1: Legal-SBERT architecture, based on the SBERT architecture published by Reimers and Gurevych (2019).

Baselines To evaluate the effects of replacing BERT with Legal-BERT in the SBERT architecture and the effects of the additional pretraining applied to Pretrained Legal-SBERT, we use four baselines. The first is the pretrained SentenceTransformer all-mpnet-base-v2 (AMB2) model, which we download as is from Hugging Face. AMB2 has the highest overall performance out of the models in the SentenceTransformers family, and is built upon the underlying encoder MPNet, which combines permuted language modeling (PLM) with masked language modeling (MLM) to improve BERT. We also implement a second baseline in which we perform the same additional pretraining on AMB2 as is performed on Legal-SBERT. The third and fourth baselines are the original SBERT model, both as-is and with the additional pretraining.

Data Adaption To overcome the lack of labeled data suitable for legal sentence transformers, and model how researchers in other domains can navigate similar data shortfalls while evaluating sentence transformers, we create binary classification tasks for the sentence transformer by adapting data from three sources

- Word Sense Disambiguation (WSD) data. WSD is the task of identifying which of a word's multiple meanings is being referenced given a sentence containing the word.
- Binary classification data from encoder benchmarks
- Multiclass classification data from encoder benchmarks.

5 Experiments

5.1 Data

One pretraining dataset and three test datasets (Word Sense Disambiguation, Overruling, and Unfair Terms of Service) were generated and gathered. In order to format the data in a manner suitable for sentence transformer binary classificaiton, each dataset was made by splitting sentences into two classes and creating sentence pairs that either had sentences from the same or different classes.

Pretraining Data The sentences used for pretraining were generated by requesting GPT-3.5 to generate sentences using the legal and non-legal definitions for 6 words identified to have different legal and everyday meanings by Nyarko and Sanga (2021). The exact prompts used were "Write 1000 unique sentences using [word] that do refer to its (legal / everyday) definition" substituting [word] and choosing legal or everyday as appropriate. The words are capacity, claim, depose, discovery, hearing, motion. Note that due to the token context limit that GPT 3.5 models have, the model was unable to create 1000 unique sentences in a single request. To address this, multiple requests were made, discarding repeat sentences, until the corpus was of suitable size. Repeat sentences were excluded. In total, there are 18719 sentences and approximately 1300 sentences per word per context. To create a dataset suitable for training, we, for each word, pair the sentences incorporating that word using the following split: 25% sentence pairs that both use the legal definition of the word (positively labeled), 25% sentence pairs that both use the everyday definition of the word (positively labeled), and 50% sentence pairs where one sentence uses the legal definition of the word and one sentences uses the everyday definition (negatively labeled). The sentences for each word are then concatenated into one single larger dataset.

Word Sense Disambiguation A WSD dataset was created by following the same steps performed to create the pretraining dataset. However, sentences were only generated for one word: relief, which was also identified as a to have different legal and everyday meanings by Nyarko and Sanga (2021) and is not one of the words used to generate pretraining data. Because the data format is identical to that of the pretraining data, the dataset is entirely used for testing.

Overruling The overruling dataset was generated by Zheng et al. (2021) and consists of 2400 binary-classified sentences that are either overruling or not overruling previous court statements. The dataset was designed as a binary classification benchmark for encoders. The 2400 words are split into an 80/20 training/test split. 8000/2000 unique training/testing sentences pairs are then generated by randomly and uniformly sampling from the 1920/480 sentences, respectively. A 50/50 split between positive and negative sentence pairs is created using the same methodology as the pretraining data.

Unfair Terms of Service The unfair terms of service dataset was generated by Lippi et al. (2018) and consists of approximately 9414 sentences labeled with none or some of 8 labels for unfair types of terms of service. The dataset was designed as a multi-class classification benchmark for encoders. The 9414 words are split into an 80/20 training/test split. 8000/2000 unique training/testing sentences pairs are then generated by randomly and uniformly sampling from the 7531/1883 sentences, respectively. A 50/50 split between positive and negative sentence pairs is created using the same methodology as the pretraining data.

5.2 Evaluation method

The evaluation metric used is the binary classification F1 score. The binary classification task is to determine whether or not the pair of sentences provided to the sentence transformer are from the same class.

5.3 Experimental details

For all experiments, the model was trained for 3 epochs using a batch size of 16. The epoch size was chosen based on loss plateaus and the batch size was chosen due to memory constraints. Training time varied between 5 and 15 minutes. Training was performed using an AdamW optimizer with a learning rate of 2e-5 and with a weight decay of 0.01. Gradient clipping was performed with a max gradient norm of 1. These hyperparameters were chosen based on the original sentence transformers experimentation (Reimers and Gurevych, 2019).

5.4 Results

Word Sense Disambiguation

Model	F1
SBERT	98.71
SBERT (pretrained)	98.57
AMB2	99.54
AMB2 (pretrained)	**99.88**
Legal-SBERT	99.37
Legal-SBERT (pretrained)	99.74

The models have extremely high classification accuracy, reflecting the inadequacy of WSD using GPT-3.5 sentences to meaningfully identify gaps in performance between the different models. The resuls are likely due to the context words that appear in sentences from each class being too predictable.

Overruling

Model	F1
SBERT	75.87
SBERT (pretrained)	68.86
AMB2	90.44
AMB2 (pretrained)	84.35
Legal-SBERT	**90.81**
Legal-SBERT (pretrained)	84.35

The models have lower and more varied performance, suggesting this task is more suitable to evaluating model performance. Legal-SBERT achieves the best performance, followed closely by AMB2. This suggests that the pretraining regimes of Legal-BERT and MPNet (the base encoder of AMB2) confer similar advantages. Suprisingly, the pretrained models all achieve worse performance than their counterparts, suggesting that the pretraining regime was disadvantageous for this task.

Unfair Terms of Service

Model	F1
SBERT	57.99
SBERT (pretrained)	58.86
AMB2	57.21
AMB2 (pretrained)	54.53
Legal-SBERT	58.13
Legal-SBERT (pretrained)	**61.51**

All models demonstrate surprisingly low performance, potentially due to the challenges imposed by merging 8 categories of unfair sentences into one category. Notably, the pretrained Legal-SBERT model performs significantly better than any other model. In addition, the Legal-SBERT model has performance on par with the SBERT-model, suggesting no advantage was conferred by the Legal-BERT encoder. Pretraining produces performance gains for both the Legal-SBERT and SBERT models, indicating that the additional pretraining may be beneficial for challenging tasks.

6 Analysis

Word Sense Disambiguation using LLM Sentences The data indicates that WSD on sentences generated by a LLM is a too simple of a task to be an adequate benchmark for sentence transformers. Sentences generated by an LLM contain too little of the natural variation present in human langauge and are much more prone to following predictable patterns. For example, nearly 40% of legal sentences generated by the model contain the word "plaintiff", and nearly 87% of sentences contained the words "plaintiff", "defendant", and/or "legal", making it trivial for a sentence transformer to achieve extremely high accuracy by predicting upon these words alone.

6.1 Sentence Embeddings from Overruling Dataset

The overruling dataset indicates that the pretraining regimes of the MPNet and Legal-BERT models produce comparable performance gains. Compared to BERT, Legal-BERT's pretraining regime involves adding domain-specific data while MPNet's training regime improves upon BERT by also applying PLM, which solves the problem of BERT neglecting dependencies among predicted tokens. This suggests that model improvements from training data relevance and from training methodology are comparable in this case.

We visualize the sentence embeddings created by the SBERT and Legal-SBERT models to further investigate the effects on the embeddings of the following two choices: (a) switching the BERT encoder for the Legal-BERT encoder and (b) finetuning affect the sentences embeddings.

We first visualize the differences in the embeddings created by the Legal-SBERT and SBERT models on overruling and nonoverruling sentences from the Overruling dataset (see Figures 2 and 3). These images visually indicate that the replacing the BERT encoder with Legal-BERT leads to a significant shift in the models ability to comprehend nuances contained in embeddings. In conjunction with the previous quantitive results, it appears that an encoder change leaders to a meaningful improvement in model comprehension and performance.

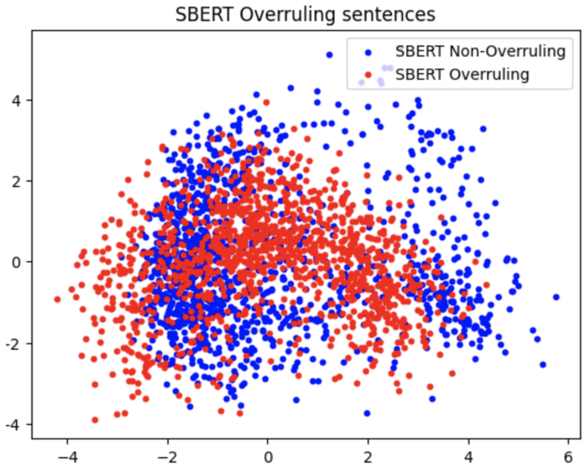

Figure 2: SBERT embeddings of non-overruling and overruling sentences

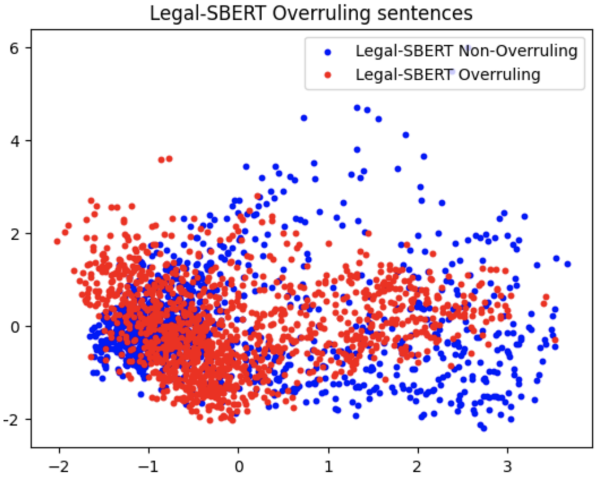

Figure 3: Legal-SBERT embeddings of non-overruling and overruling sentences

We next visualize the effects of applying the pretraining regime to the model (see Figures 3 and 4). Perplexingly, pretraining the model leads to extremely similar embeddings between the overruling and non-overruling sentences. One reason for this may be that, as previously mentioned, "legal" sentences generated by a LLM reuse many of the same judicial words. Therefore, pretraining a model on "legal" sentences from an LLM leads to the model excessively grouping together sentences containing the judicial words, leading to the model's comparative inability to distinguish types of legal words from one another. Therefore, by learning to distinguish the legal and everyday meanings of words present in the pretraining date, the model loses its ability to distinguish between different types of legal sentences.

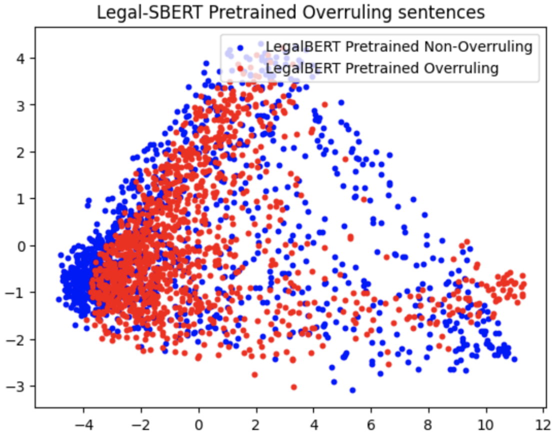

Figure 4: Pretrained Legal-SBERT embeddings of non-overruling and overruling sentences

Pretrained Legal-SBERT Performance on Pretraining Words To investigate whether the pretraining regime helps the model better understand the meanings of sentences containing the words it was pretrained on WSD for, we test Pretrained Legal-SBERT's performance on sentences in the Overruling dataset that include at least one of the 6 words it was pretrained on.

Overruling (Model: Pretrained Legal-SBERT):

Dataset	F1
All Sentences	84.35
Sentences Containing Pretraining Words	70.43

Unfair Terms of Service (Model: Pretrained Legal-SBERT):

Dataset	F1
All Sentences	61.51
Sentences Containing Pretraining Words	58.98

Notably, Pretrained Legal-SBERT's performance for words that it was pretrained on is worse than its overall performance for both datasets. However, for the more challenging task presented by the Unfair Terms of Service, the additional pretraining still provides an improvement in overall model performance. This indicates that, although pretraining causes the model to lose some of its nuanced understanding of the words it was pretrained on, pretraining may still have some overall benefits in more challenging datasets.

7 Conclusion

This project demonstrates the value of creating legal specific sentence transformers by leveraging the Sentence-BERT architecture and encoders pretrained on legal corpora. It indicates the performance gains of training BERT-based encoders on legal domain corpora is comparable to or surpasses the performance gains achieved by improving the BERT training regime through practices such as PLM. It also suggests that further pretraining, using data such as sentences generated by a LLM, can improve model performance depending on the legal tasks' difficulty. It also presents a methodology for adapting encoder benchmark data to create binary-classification datasets suitable for sentence transformers in domains where there may not be sufficient data labeled for sentence transformers. Lastly, it indicates that WSD datasets generated by LLMs are limited in their usefulness as pretraining datasets because they fail to capture the true variety of natural language contained within legal corpora.

Given that our work surfaces benefits of the pretraining regime on model performance for especially challenging legal tasks, an interesting next step would be to assess the effects of the pretraining regime when performed using sentences gathered from real-world datasets, instead of sentences synthetically generated.

References

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: the muppets straight out of law school. *CoRR*, abs/2010.02559.

Alexis Conneau and Douwe Kiela. 2018. Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*.

Emad Elwany, Dave Moore, and Gaurav Oberoi. 2019. BERT goes to law school: Quantifying the competitive advantage of access to large legal corpora in contract understanding. *CoRR*, abs/1911.00473.

Marco Lippi, Przemyslaw Palka, Giuseppe Contissa, Francesca Lagioia, Hans-Wolfgang Micklitz, Giovanni Sartor, and Paolo Torroni. 2018. CLAUDETTE: an automated detector of potentially unfair clauses in online terms of service. *CoRR*, abs/1805.01217.

Julian Nyarko and Sarath Sanga. 2021. A statistical test for legal interpretation: Theory and applications. *The Journal of Law, Economics, and Organization*, 38(2):539–569.

Nils Reimers and Iryna Gurevych. 2019. 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.

Lucia Zheng, Neel Guha, Brandon R. Anderson, Peter Henderson, and Daniel E. Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the casehold dataset. In *Proceedings of the 18th International Conference on Artificial Intelligence and Law*. Association for Computing Machinery.