Evaluating the impact of incorporating 'legalese' definitions and abstractive summarization on the categorization of legal cases by their holdings

Shiu Tin Ivan Ko [*1] **Daniela Cortes Bermudez** [*1] **Huiyun Zhang** [1] **Henry Han** [1]

Abstract

Legal text is difficult to understand and requires domain-specific knowledge to read. This work aims to investigate the effect that model stacking and input processing have on information fidelity with the motivation to explore possibilities of expanding the accessibility of legal texts. We developed a legal dictionary through the United States Courts' Glossary of Legal Terms[1] to map complex terms into simple English and used FLAN-T5 to summarize observations. To evaluate performance, we used binary text classification to predict case holdings using LLMs (Large Language Models) and evaluated the results with and without model pretraining. To assess information fidelity, we ask: "Does model stacking affect classification performance?" and "Does performance change with pretraining?"

1. Introduction

Legalese language is notoriously difficult to understand. This domain-specific terminology is characterized by lengthy, wordy, and complex sentence structure[2]. Its comprehension is mostly exclusive to individuals with an extensive legal background. Such difficulties have resulted in a movement for plain English legal text from the US government itself[3]. Similarly, the Plain Language Act of 2010[4] is likely a result of such movement, albeit covering more general areas. Professor Robert D. Eagleson defines plain English as clear, straightforward, and concise language. It avoids inflated vocabulary and complex sentence structure. Plain English allows the audience to easily understand a message[5]. The movement for plain English in government documents started almost a century ago, but recent studies give further support to such a movement. One key finding is that a writing style known as "center-embedding" makes legal documents hard to read - not just for laymen but also for trained lawyers (Martínez et al., 2022).

This research compares text classification models and explores whether the classification of a case statement to its holding will be affected by model stacking or data processing. Text summarization is a natural language processing approach that has practical applications to shrink documents while retaining information fidelity. The United States Courts' website has a glossary of over 200 legal terms available. We compiled each of the terms into a dictionary and utilized this to map legal terms to their definitions. We observed how the replacement of legal terms with their corresponding description, for both pre-summarization and post-summarization scenarios, affects the text classification of legal cases to its corresponding case holding [6]. For text classification, we selected three LLMs — BERT, LegalBERT, and GPT2. These LLMs were tested with and without pretraining. Then we used various *input processing*[7] on the data before performing the task of binary text classification. The data is borrowed from a subset of observations from the CaseHOLD dataset (Zheng et al., 2021). Previous work conducted to build legal domain-specific summarization and simplification models has stressed the importance of high-quality legal data for improved model performance, and the need for the development of text processing tools within the legal domain (Gallegos & George, 2019; Manor & Li, 2019).

In this study, we scale the model stack complexity down to only adding one to two extra stacks after the original input, before feeding the input into binary text classification. We chose to investigate model stacking due to the ubiquity of

[*]Equal contribution [1]Department of Computer Science and Engineering, Baylor University, Waco, Texas, United States. Correspondence to: Henry Han <Henry_Han@baylor.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

[1]https://www.uscourts.gov/glossary

[2]https://www.law.cornell.edu/wex/legalese

[3]https://www.plainlanguage.gov/

[4]https://www.dni.gov/index.php/plain-language-act

[5]https://www.plainlanguage.gov/about/definitions/short-definition/

[6]The case holding is the final decision the court reached

[7]In this study, this term is interchangeable with *treatment*

chaining multiple models together to perform automated tasks. In such, we reckon that most organizations may not have enough resources to pretrain their core model with different inputs, but rather, focus their core model with one type of input to perform specific tasks. However, as a model becomes more customized towards a particular set of inputs and tasks, we wonder if the custom model performs better or worse compared to pre-customization. Therefore, we attempt to shed some light on model stacking in this study as well.

The remainder of this work is organized as follows: In Section 2, we reference previous research related to our current approach and data sources; In Section 3, we describe the data used and the data preprocessing steps; In Section 4, we explore the approaches used in each step of the study; In Section 5, we showcase the results of running our predictive model on the data generated after summarization and definition mapping, and compare it to the prediction results from the original data; In Section 6, we discuss the implications of our results; In Section 7, we wrap up the study with the main conclusions.

2. Related Works

2.1. CaseHOLD

The CaseHOLD benchmark dataset contains 53,000+ multiple choice questions, each observation containing a cited case and the options being five different holdings. Only one of the presented holding statements is the right answer (Zheng et al., 2021). This dataset was constructed from the Harvard Law Library case law corpus [8].

2.2. BillSum

The BillSum corpus is a benchmark dataset for legal document summarization, comprised of US Congressional and California state bills (Kornilova & Eidelman, 2019). This dataset includes a corpus of 22,218 reference summaries split into 18,949 rows of training samples, and 3,269 validation samples from US Congressional bills. It also contains an additional test set of 1,237 California bills and reference summaries.

2.3. "Legalese" glossary

Due to a lack of existing legal term dictionary datasets, we compiled a legal glossary by extracting each legal term and a corresponding definition from the United States Courts' Glossary of Legal Terms. If a definition included more than one sentence, only the first sentence was registered as its definition. If two terms had the same definition, they were included as two separate observations in the compiled dataset. If the term is written in Latin, its English equivalent was registered as its definition. Examples provided in the glossary were excluded from the registered definition. If a term had two or more numbered definitions, all of them were included in a single observation. Each definition was processed to exclude ending periods.

3. Data

The classification dataset is borrowed from "When Does Pretraining Help?" (Zheng et al., 2021), which is derived from Harvard Law School's Caselaw Access Project. Since the researchers offered a subset of their data for free, and legal text is difficult to obtain, we decided to use it to save resources. This dataset is sufficient for our purpose to discern how well machine-translated legal text performs on text classification. Our input consists of the version presented in the CaseHOLD benchmark dataset. This version consists of a pairing between a case abstract and different holdings of that case, and a label marking whether that pairing is correct. The summarization methods were trained using the BillSum dataset (Kornilova & Eidelman, 2019).

The original CaseHOLD data has 264,890 rows of data considering each cited case paired with the five different possible answers. Due to the nature of the pairing, with only one correct observation, the percentage of positive labels is thus 20% of the total labels. We used two different subsets of the CaseHOLD dataset. Through random selection, we selected 5,000 and 60,000 rows of data. We used the 60,000 rows dataset to pre-train the three LLM models — which generates three custom models we called "Gen1" — and we used the 5,000 rows to test the models.

3.1. Terminology and Models

Some terms are used interchangeably in this paper. The term "definition" refers to the lexicon replacement of legal terms with their plain English definition. The terms "input" and "dataset" both refer to the data used. "Treatment" and "treated input" refers to the input that received data processing, such as summarization or definition. The dataset with 5,000 random samples will be referred to as the "5k" dataset, and the one with 60,000 samples as the "60k" dataset. We also use shorthand terms for the input variance: D for "defined", and S for "summarized". Hence, D + S references inputs that were defined and then summarized. The 5k dataset prior to input processing is termed the "original" version. The base models used are the original BERT (Devlin et al., 2018), LegalBERT (Chalkidis et al., 2020), and GPT2 (Radford et al., 2019) models from HuggingFace. Base models after pretraining will be referred to as "Gen1".

[8] https://case.law/

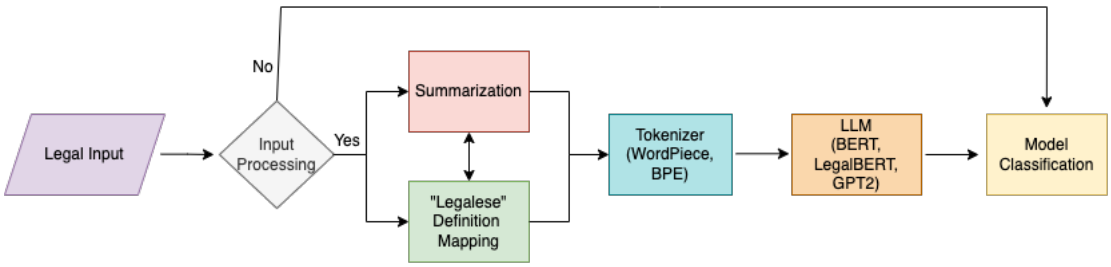

Figure 1. Flowchart of steps in the case-to-holding classification process

3.2. Data Processing

We processed the 5k dataset and created five total datasets (See Table A1):

1. Original
2. D
3. D + S
4. S
5. S + D

We consider S to be the only stacking we add on top of each model, as it is done by the FLAN-T5 model (See section 4.1). Given that D is a simple mapping of a key to a value, it does not use a base model for its function (See section 4.3). Therefore, it is not considered part of the model stacking.

4. Methods

In this study, we explored three tasks: text classification, "legalese" definition mapping, and summarization. The text classification methodology examines the effects that term-to-definition replacement and summarization have on the classification of a case observation and its holding when presented with different holding options. There are two main types of text summarization in natural language processing, extractive and abstractive. We implemented an abstractive summarization model in this work.

4.1. Abstractive summarization using fine-tuned FLAN-T5 base model

Extractive summarizations are composed of sentences contained in the original text. On the other hand, abstractive summarization approaches create paraphrases of the current text, generating new sentences rather than picking words from the original text (Widyassari et al., 2022). One of the common uses of the T5 Model family, the "Text-to-Text Transfer Transformer" family, is for text generation such as abstractive summarization (Raffel et al., 2019). For our implementation of abstractive summarization, we used Google's FLAN-T5 as the baseline model (Chung et al., 2022). The FLAN-T5 model was trained for summarization and fined-tuned for legal data using the BillSum corpus (Kornilova & Eidelman, 2019). This model fine-tuning was implemented by HuggingFace Chief Evangelist, Julien Simon (2023). The model was trained using 18,949 training samples and 2,369 validation samples and made available on the HuggingFace Models page.

4.2. Case-holding classification and model capabilities

We established the baseline of text classification by inputting our 5k dataset into our three base models to determine how well each can classify whether a holding matches an observed case. We are aware of the various capabilities and design intentions of the different LLMs, but we decided to measure their performance regardless. Because we want to test out different model architectures and their classification capacities, both BERT and GPT2 provide a good basis for what we are testing for. GPT2 is a unidirectional decoder-based architecture and is better at text generation, whereas BERT is a bidirectional encoder-based architecture and performs better at text classification and fill-in-the-blanks. Therefore, BERT is expected to outperform GPT2 in the binary text classification task, and the results support this theory.

4.3. "Legalese" definition mapping

Using the legal terms dictionary we generated, we mapped each term included in the dictionary to its registered definition. The mapping returns a modified observation, where each legal lexicon in an observation that matches a term in the legal dictionary is replaced with its corresponding plain English definition.

4.4. Process Framework

A diagram of our full approach is shown in Figure 1. Each input has five paths to the task: 1. Input → Classification; 2. Input → "Legalese" Definition Mapping → Classification; 3. Input → Summarization → Classification; 4. Input

→ "Legalese" Definition Mapping → Summarization → Classification; 5. Input → Summarization → "Legalese" Definition Mapping → Classification.

The five paths are repeated for the "Gen1" version of each model. In short, a total of 30 trials are run ($3\ models * 2\ generations * 5\ inputs$).

Our results are based on 5-fold cross-validation using the 5k dataset. The 60k data is simply passed through an 80/20 train-test split as pretraining for our models. For fine-tuning the parameters, we referred to "How to Fine-Tune BERT for Text Classification" (Sun et al., 2020). Each input is also vectorized using HuggingFace's Tokenizer before feeding into the LLMs. GPT2 uses BPE while BERT-based models use Wordpiece as tokenizers. The tokenization is truncated to the "left", meaning that once the token limit is reached, newer token inputs are added to the end, and older input from the start is discarded. For example, if we have a token limit of 3, and two sentences: A. "How are you?" and B. "I'm fine thank you". The tokenization of A would result in ['How', 'are', 'you'], whereas tokenization of B would return ['fine', 'thank', 'you'].

5. Results

5.1. Metrics Selection

Accuracy, F1 score, and D-index were chosen for performance evaluation. Accuracy is included because it's a common metric. Same for the F1 score. The D-index, or diagnostic index, is a novel machine-learning evaluation method introduced by Dr. Henry Han (2022). This evaluation metric is designed to detect small performance differences between models and combines multiple measures to provide a comprehensive and interpretable machine learning evaluation. It detects and takes into account data imbalance. The larger the D-index value, the better the learning performance of the predictive function of a model is. Its range is between 1.1699 and 2 assuming no underfitting, which is represented by a D-index $\geq$ 1.1699. All of our results are above this threshold, showing no evidence of underfitting. Additionally, the imbalance point of the D-index is 1.5339. As presented in Table 1, some D-index values found are close to this imbalance point. In such instances, $\approx$ 19.48% of the labels belonging to the minority class of 1, are mostly misclassified. Precision and Recall are captured in our data, but not shown as both the F1-Score and D-index reflect their effects.

5.2. Setup and Evaluation

We found that some of the results shown in Table 1 are not normally distributed by running visual inspection through a QQ-plot and numeric tests using Shapiro-Wilk and Anderson-Darling. For Shapiro-Wilk tests, 36 out of 90 p-values ($30\ trials * 3\ metrics = 90$) are <=0.05. Given that not all values fall under this threshold, we fail to reject the null hypothesis that all the samples are normally distributed. A similar case is seen in the Anderson-Darling test. For this test, the null hypothesis is rejected when the returned statistic is larger than the significance level of the chosen critical value. The results show that 13 out of 90 statistics have values lower than the 5% critical value, and 8 out of 90 are lower than the 15% critical value threshold.

Having a non-normal distribution eliminates the possibility of using statistical methods such as Student T-tests or ANOVA. Consequently, nonparametric tests — Kruskal-Wallis and Wilcoxon's signed-rank tests — were selected. The Kruskal-Wallis tests were performed for each group of models based on different inputs. We performed the test for the base and Gen1 variants of each model. The Kruskal-Wallis tests confirmed that all treatments are significantly different from each other in terms of Accuracy, F1, and D-index. It should be noted that the comparisons between base and Gen1 models yield different results. Additionally, 10 out of 45 of these comparisons (90 trials / 2 pairs) didn't pass the Kruskal-Wallis test. Specifically, the ten pairs are:

BERT Base vs. Gen1 D on F1; LegalBERT Base vs. Gen1 S on F1, D-index, and Accuracy; LegalBERT Base vs. Gen1 D + S on Accuracy; LegalBERT Base vs. Gen1 S + D on F1, D-index and Accuracy; LegalBERT Base vs. Gen1 Original on F1 and D-index

These results are likely due to those models reaching near-optimal performance, thus exhibiting high similarity. However, most comparative changes appear significant, as shown by the Wilcoxon tests. The Wilcoxon tests evaluate each model against each processed input, and compares the processed inputs against each other. These tests show that most treatments are significantly different where the $P < 0.01$. Results that do not meet this criteria are the Accuracy and D-index of BERT Gen1, as well as the Accuracy of GPT2 Gen1 in the S + D results. Comparisons between the base models and their Gen1 version yielded a similar result to Kruskal-Wallis mentioned above. The LegalBERT base and Gen1 original, S and S + D have p-values > 0.01 on Accuracy, F1 and D-index. The same is true for BERT vs. Gen1 D on F1.

Given the above analysis, we reject the null hypothesis with $P < 0.01$ for different input treatments of the Kruskal-Wallis test. The population median of all groups is equal, meaning that the treatments produce significant effects. In addition, we also reject the null hypothesis for $P < 0.01$ across most pairs of the Wilcoxon test, but with the two exceptions noted above. However, inter-generational tests show that LegalBERT base and Gen1 may not have enough differentiation, which could be attributed to the relative optimal performance ceiling. Overall, the findings support

Table 1. Results of the accuracy, F1-score, and D-index metrics for the base BERT, LegalBERT, and GPT2 models for each processed dataset, both for pre-trained and not pretrained models. Gen1 = Pretrained model; D = Defined; S = Summarized; S + D = Summarized + Defined; D + S = Defined + Summarized

		BERT		Legal BERT		GPT2	
		Base	Gen1	Base	Gen1	Base	Gen1
Acc	Original	**0.809 ± 0.011**	**0.824 ± 0.008**	**0.836 ± 0.014**	**0.838 ± 0.007**	**0.782 ± 0.035**	**0.818 ± 0.008**
	D	0.801 ± 0.014	0.812 ± 0.011	0.824 ± 0.009	0.829 ± 0.009	0.754 ± 0.066	0.803 ± 0.011
	D + S	0.799 ± 0.013	0.803 ± 0.010	0.807 ± 0.008	0.807 ± 0.012	0.763 ± 0.044	0.785 ± 0.014
	S	0.805 ± 0.010	0.810 ± 0.011	0.820 ± 0.009	0.819 ± 0.010	0.769 ± 0.036	0.793 ± 0.010
	S + D	0.800 ± 0.015	0.807 ± 0.009	0.814 ± 0.009	0.814 ± 0.006	0.760 ± 0.053	0.793 ± 0.010
F1	Original	**0.374 ± 0.084**	**0.424 ± 0.045**	**0.440 ± 0.134**	**0.481 ± 0.044**	**0.264 ± 0.141**	**0.440 ± 0.031**
	D	0.323 ± 0.091	0.336 ± 0.055	0.369 ± 0.124	0.418 ± 0.051	0.242 ± 0.107	0.321 ± 0.054
	D + S	0.225 ± 0.074	0.280 ± 0.049	0.227 ± 0.092	0.292 ± 0.046	0.161 ± 0.094	0.198 ± 0.055
	S	0.310 ± 0.069	0.357 ± 0.048	0.340 ± 0.112	0.379 ± 0.041	0.208 ± 0.095	0.293 ± 0.045
	S + D	0.269 ± 0.067	0.304 ± 0.044	0.278 ± 0.122	0.322 ± 0.044	0.182 ± 0.093	0.269 ± 0.048
D-Index	Original	**1.550 ± 0.030**	**1.580 ± 0.023**	**1.601 ± 0.058**	**1.616 ± 0.025**	**1.487 ± 0.053**	**1.583 ± 0.017**
	D	1.522 ± 0.030	1.534 ± 0.025	1.561 ± 0.046	1.581 ± 0.026	1.444 ± 0.073	1.520 ± 0.024
	D + S	1.484 ± 0.019	1.506 ± 0.017	1.495 ± 0.025	1.513 ± 0.020	1.427 ± 0.036	1.459 ± 0.017
	S	1.518 ± 0.029	1.518 ± 0.029	1.546 ± 0.040	1.556 ± 0.021	1.449 ± 0.035	1.500 ± 0.017
	S + D	1.496 ± 0.023	1.517 ± 0.017	1.519 ± 0.039	1.530 ± 0.017	1.430 ± 0.049	1.491 ± 0.014

that there is a significant performance difference between data processing treatments.

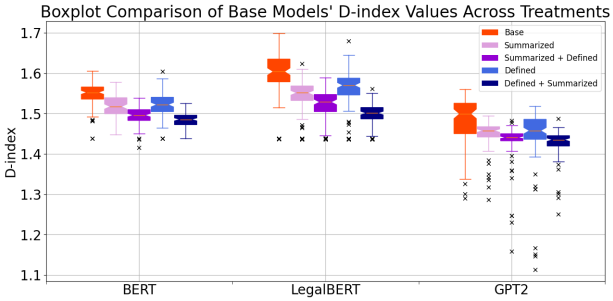

Figure 2. Performance Overview of Base Models for Each Treatment

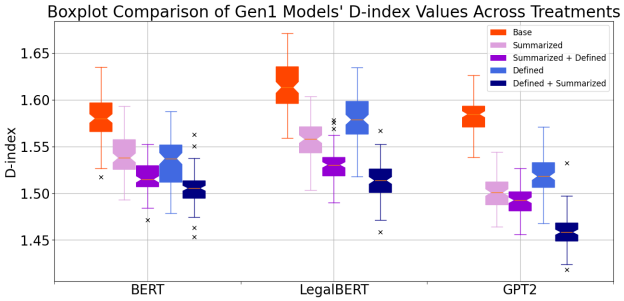

Figure 3. Performance Overview of Gen1 Models for Each Treatment

5.3. Interpretation

Table 1, Figure 2, and Figure 3 portray the results of each text classification for three LLM models and its Gen1 variance. The results shown in Table 1 are the mean and standard deviations of a 20-epoch run in a 5-fold stratified cross-validation session, totalling a 100 epochs per model per input. It is important to note that the performance of our models is not ideal. In fact, the F1-Scores are very low. There are several reasons for this, the main one being that legal text classification requires a vast amount of pretraining.

The Gen1 model results show that pretraining results in a performance boost. However, it should be noted that we are not trying to achieve the best performance for the task. Instead, we are examining the performance change of each model, relating to the question of whether model stacking and data processing affect classification performance. In other words, the metrics of the models is not the focus, but the differentiation of the metrics is. The best-performing

results for each model are highlighted in Table 1. This table shows that the original dataset outperforms all other inputs. Such results indicate that the performance of the binary classification task is affected by data processing and model stacking. This is also consistent with the basis of pretraining, as the models are pretrained on original input, but not treated input.

Figures 2 and 3 show that the D + S input tends to have the worst relative performance across all models, which shows that data processing is not symmetric. Performing summarization or definition mapping first leads to a significantly different performance. While definition mapping alone usually results in the least performance drop from original input, except in BERT Gen1, summarizing it after performs the worst across all results.

Another key finding is that while we expected greater performance drop correlating to more pretraining when model stacking, the result shows that such correlation does not always exist. We observed that pretraining the model improves the custom Gen1 models' task performance — not just on original data, but all treated input as well. Overall, we found Gen1 models perform better than base models across the broad, with an average of 1.86% Accuracy, 23.73% F1 and 2.24% D-index performance increase respectively.

We also investigated if the amount of pretraining reveals a larger performance drop when comparing treated datasets against the original. This evaluates whether the more pretraining a model receives, the more sensitive it is to a certain type of input, thus becoming less effective when given different inputs. For example, comparing how BERT base model work on summarized input compared to the original input. We found that while some Gen1 models have a larger performance decrease when compared to the base model on treated inputs, others behave differently. On average, the metric change from original to treated input is -9.59% for BERT base, -10.08% for BERT Gen1, -12.45% for LegalBERT, -11.15% for LegalBERT Gen1, -10.26% for GPT2, and -15.78% for GPT2 Gen1. For most results, the metric difference of Gen1 is higher than base, meaning that there exists correlation between pretraining and performance drop in treated inputs. In other words, the more pretrained a model, the more adapted it is to a type of input, and therefore the higher the performance drop when given differently treated inputs. However, some Gen1 models perform better on treated input than their base model counterparts. For BERT, Gen1 Accuracy over D has a lower performance drop than base. For GPT2, the same goes for F1 in D + S and S. For LegalBERT, Accuracy in D, F1 in D, D + S, S, and S + D, and D-index in D and D + S all have lower Gen1 performance drop than the base model. Such irregularity shows that while model stacking may generally lead to larger performance decreases when given unfamiliar input, there are exceptions. Thus, it is possible to use just one model to tackle both similar and different inputs, as long as the performance lost is evaluated to be acceptable. Please refer to *Figure A1* in Appendix A for a visual representation of the percentage changes.

6. Discussion

BERT-based models are pretrained using Wikipedia. GPT2 was pretrained using WebText, a result of web scrapping outbound links from Reddit (Radford et al., 2019). Due to the domain-specific nature of legal text, such models perform poorly on legal case classification. Our results show that there is a performance improvement both in the use of LegalBERT and using model pretraining on all models. This supports that performance does change with pretraining. Because of the domain-specific nature of the task, legal experts are necessary to collaborate in the building of a pretraining dataset. However, model pretraining is expensive both environmentally and financially for the legal domain. As mentioned in Zheng et al., the cost of hiring legal attorneys to check for legal holding classification is expensive on itself, and training a model with 15GB of data can cost over $1M.

The summarization and "legalese" definition mapping are also constrained to domain specific expertise. The corpus available for summarization training is quite limited in the legal domain. To our knowledge, the Billsum dataset is the most comprehensive summarization dataset available for legal text. However, this dataset is constrained to 22,218 US Congresional bills. As shown in Zheng et al., the amount of pretraining that a model needs to perform optimally requires a large amount of data. Therefore, the summarization method is limited by the relatively small data resources available to train the FLAN-T5 Model. In order to create reference summaries to train the model with, a legal domain expertise must produce it. This task is costly and time consuming, also restricting the capabilities of model fine-tuning. Additionally, the "legalese" definition mapping is limited both by the amount of data available and the lack of benchmark datasets to do so. To our knowledge, there is no comprehensive publicly available dictionary from "legalese" to its informal definition. This limits the amount of terms that can be used in the definition mapping task. Domain expertise could also play a role in creating concise definitions, as the ones we gathered from the U.S. Courts' Glossary of Legal Terms tend to be lengthy (See Table A1). Another constraint of the definition mapping task is that it is context-based. This leads to a single word having several meanings. For example, the registered definition for "Defendant" is "In a civil case, the person or organization against whom the plaintiff brings suit; in a criminal case, the person accused

of the crime" and also "An individual (or business) against whom a lawsuit is filed". Additional context checking within a legal case would have to be explored to appropriately assign a term to its definition. This example also shows that legal definitions often reference other legal terms within it. Within the "defendant" definitions, the words "plaintiff" and "lawsuit" are included.

As mentioned in Section 5, there is a performance drop, both in base and Gen1 models, between the original dataset and those with data processing and model stacking. This result was expected given that the Gen1 models were pretrained with the original data. Therefore, they acknowledge the legal terms as part of their classification given that this is what they were trained on. Such terms may be replaced or excluded in the definition mapping and summarization processes. Different results may have arisen if the model pretraining had been personalized to each treated dataset.

7. Conclusions

The structure of legal text makes its interpretation difficult for individuals missing legal expertise. Such constraints make legal text inaccessible to the general public. Additionally, as mentioned by Zheng et al., the number of benchmark legal datasets is limited by the cost and resources needed to produce them. To produce legal data, help from individuals in the legal industry is needed. Legal documents are extensive and therefore, expensive to train a model with. These limitations have made the number of available legal studies in the machine-learning realm quite scarce. In terms of our study, we tested the possibility of making legal text simple and concise through the inclusion of explicit definitions and text summarization. We evaluated each input processing through different LLM models and examined the effect of model stacking on binary text classification. The results show a minor performance impact on the D-index and Accuracy. However, the large negative impact on F1 score shows that we should not be too optimistic about model stacking. We also showed that with pretraining, even in relatively small amounts, model performance can be improved.

Acknowledgements

This work is partially supported by NASA Grant 80NSSC22K1015, NSF 2229138, and the McCollum endowed chair startup fund.

References

Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., and Androutsopoulos, I. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 2898–2904, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.261.

Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., Webson, A., Gu, S. S., Dai, Z., Suzgun, M., Chen, X., Chowdhery, A., Castro-Ros, A., Pellat, M., Robinson, K., Valter, D., Narang, S., Mishra, G., Yu, A., Zhao, V., Huang, Y., Dai, A., Yu, H., Petrov, S., Chi, E. H., Dean, J., Devlin, J., Roberts, A., Zhou, D., Le, Q. V., and Wei, J. Scaling instruction-finetuned language models, 2022.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Gallegos, I. and George, K. The right to remain plain: Summarization and simplification of legal documents, 2019. URL https://web.stanford.edu/class/cs224n/reports/custom_116652906.pdf.

Kornilova, A. and Eidelman, V. Billsum: A corpus for automatic summarization of US legislation. *CoRR*, abs/1910.00523, 2019. URL http://arxiv.org/abs/1910.00523.

Manor, L. and Li, J. J. Plain english summarization of contracts. *ArXiv*, abs/1906.00424, 2019.

Martínez, E., Mollica, F., and Gibson, E. Poor writing, not specialized concepts, drives processing difficulty in legal language. *Cognition*, 224:105070, Jul 2022. doi: 10.1016/j.cognition.2022.105070. URL https://www.sciencedirect.com/science/article/pii/S0010027722000580.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. 2019.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683, 2019. URL http://arxiv.org/abs/1910.10683.

Simon, J. Juliensimon/t5-base-billsum, 2023. URL https://huggingface.co/juliensimon/t5-base-billsum/.

Sun, C., Qiu, X., Xu, Y., and Huang, X. How to fine-tune bert for text classification?, Feb 2020. URL https://arxiv.org/abs/1905.05583.

Widyassari, A. P., Rustad, S., Shidik, G. F., Noersasongko, E., Syukur, A., Affandy, A., and Setiadi, D. R. Review of automatic text summarization techniques and methods. *Journal of King Saud University - Computer and Information Sciences*, 34(4):1029–1046, 2022. doi: 10.1016/j.jksuci.2020.05.006.

Zheng, L., Guha, N., Anderson, B. R., Henderson, P., and Ho, D. E. When does pretraining help? assessing self-supervised learning for law and the casehold dataset. *CoRR*, abs/2104.08671, 2021. URL https://arxiv.org/abs/2104.08671.

A. Appendix

Table A1. Example observation from CaseHOLD dataset across different treatments. D = Defined; S = Summarized; D + S = Defined + Summarized; S + D = Summarized + Defined

Dataset	Text
Original	NSF lease transaction rendered him a 'real estate salesperson' in Virginia," and therefore required him to be licensed as such. Appel-lee's Br. at 17. 2 . The Supreme Court of Virginia first addressed the issue of commission payments to unlicensed brokers and salespersons in Massie v. Dudley, refusing to enforce an agreement "made by an unlicensed person" because "its substance [was] unlawful." 173 Va. 42, 3 S.E.2d 176, 180-81 (1939). The court has consistently reiterated this principle following Massie. In Harrison & Bates, Inc. v. LSR Corp., for example, the court held unenforceable a contract to split commissions made between a licensed corporation and an unlicensed firm. 238 Va. 741, 385 S.E.2d 624 (1989); see also Hancock, Co. v. Stephens, 177 Va. 349, 14 S.E.2d 332, 334 (1941) (<HOLDING>); State Realty Co. v. Wood, 190 Va. 321, 57
D	nsf lease transaction rendered him a 'real estate salesperson' in virginia," and therefore required him to be licensed as such. appel-lee's br. at 17. 2 . the supreme government entity authorized to resolve legal disputes of virginia first addressed the 1. the disputed point between parties in a a legal action started by a a person or business that files a formal complaint with the court against a defendant based on a complaint that the defendant failed to perform a legal duty which resulted in harm to the a person or business that files a formal complaint with the court; 2. to send out officially, as in a court issuing an order of commission payments to unlicensed brokers and salespersons in massie v. dudley, refusing to enforce an agreement "made by an unlicensed person" because "its substance [was] unlawful." 173 va. 42, 3 s.e.2d 176, 180-81 (1939). the government entity authorized to resolve legal disputes has consistently reiterated this principle following massie. in harrison & bates, inc. v. lsr corp., for example, the government entity authorized to resolve legal disputes held unenforceable a an agreement between two or more people that creates an obligation to do or not to do a particular thing to split commissions made between a licensed corporation and an unlicensed firm. 238 va. 741, 385 s.e.2d 624 (1989); see also hancock, co. v. stephens, 177 va. 349, 14 s.e.2d 332, 334 (1941) (<holding>); state realty co. v. wood, 190 va. 321, 57
S	Appel-lee's Br. at 17. The Supreme Court of Virginia first addressed the issue of commission payments to unlicensed brokers and salespersons in Massie v. Dudley, refusing to enforce an agreement "made by an unregistered person" because "its substance [was] unlawful."
D + S	nsf lease transaction rendered him a "real estate salesperson" in virginia, and therefore required him to be licensed as such. The supreme government entity authorized to resolve legal disputes of virgina first addressed the disputed point between parties in a legal action started by a person or business that files a formal complaint with the court against a defendant based on a complaint that the defendant failed to perform a duty which resulted in harm to the plaintiff. The government entity has consistently reiterated this principle following massie
S + D	appel-lee's br. at 17. the supreme government entity authorized to resolve legal disputes of virginia first addressed the 1. the disputed point between parties in a a legal action started by a a person or business that files a formal complaint with the court against a defendant based on a complaint that the defendant failed to perform a legal duty which resulted in harm to the a person or business that files a formal complaint with the court; 2. to send out officially, as in a court issuing an order of commission payments to unlicensed brokers and salespersons in massie v. dudley, refusing to enforce an agreement "made by an unregistered person" because "its substance [was] unlawful."

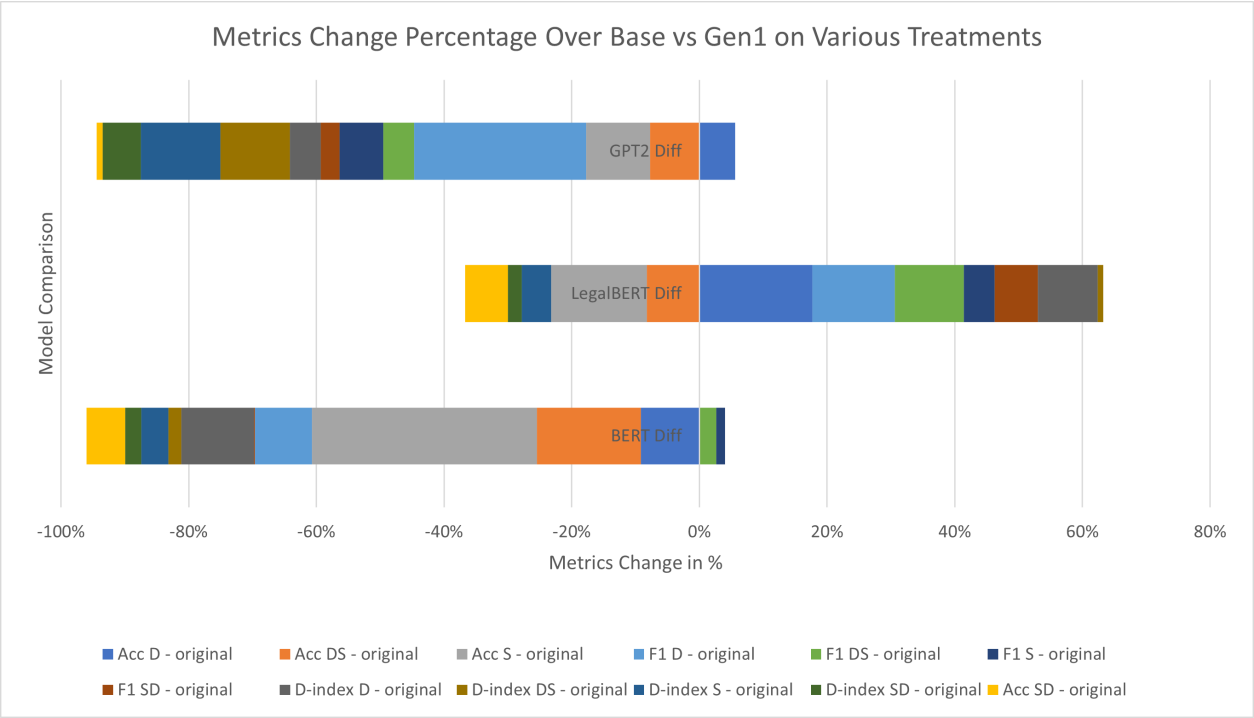

Figure A1. Percentage of metrics change over Base vs. Gen1 Models given various inputs
Positive percentage means Gen1 metrics decreases less than base model when comparing treated inputs against original. In other words, the stacked bars on the right side of 0% line represent Gen1 models that performed better when given treated inputs than the base model