Mixed-domain Language Modeling for Processing Long Legal Documents

Wenyue Hua
Claudius Legal Intelligence
wenyue@claudius.ai

Yuchen Zhang*
University of Pennsylvania
zycalice@seas.upenn.edu

Zhe Chen
Claudius Legal Intelligence
zhe@claudius.ai

Josie Li*
UC San Diego
jol027@ucsd.edu

Melanie Weber
Claudius Legal Intelligence
Harvard University
melanie@claudius.ai

Abstract

The application of Natural Language Processing (NLP) to specialized domains, such as the law, has recently received a surge of interest. As many legal services rely on processing and analyzing large collections of documents, automating such tasks with NLP tools such as language models emerges as a key challenge since legal documents may contain specialized vocabulary from other domains, such as medical terminology in personal injury text. However, most language models are general-purpose models, which either have limited reasoning capabilities on highly specialized legal terminology and syntax, such as BERT or ROBERTA, or are expensive to run and tune, such as GPT-3.5 and Claude. Thus, in this paper, we propose a specialized language model for personal injury text, LEGALRELECTRA, which is trained on mixed-domain legal and medical corpora. We show that as a small language model, our model improves over general-domain and single-domain medical and legal language models when processing mixed-domain (personal injury) text. Our training architecture implements the ELECTRA framework but utilizes REFORMER instead of BERT for its generator and discriminator. We show that this improves the model's performance on processing long passages and results in better long-range text comprehension.

1 Introduction

Following the striking success of large language models, the development of specialized language models that are adapted to domain-specific syntax and vocabulary has received increasing attention. In many application domains, such as medicine or the law, raw data is often given in text format, creating a need for NLP tools that aid in automating data processing. General-domain models typically lack expressivity on specialized domains. Rare words and domain-specific meanings of vocabulary are difficult to process with general-domain models. Thus, domain adaption had to be addressed in downstream tasks. In contrast, recent literature proposes to integrate domain adaptation into the pre-training stage. Such domain-specific pre-trained models have been developed for a range of domains, including the law (LEGAL-BERT (Chalkidis et al., 2020), Lawyer-llama (Huang et al., 2023)), medicine (CLINICAL-BERT (Huang et al., 2019), Chatdoctor (Yunxiang et al., 2023)), biomedical sciences (BIOBERT (Lee et al., 2020)) and finance (FINBERT (Yang et al., 2020), Bloomberggpt (Wu et al., 2023)), among others.

In this work, we focus on pre-trained models for applications in the law with specialization in personal injury civil suits. Personal injury legal cases arise when an individual suffers harm from an accident or injury, and someone else might be legally responsible for that harm. This can be due to negligence, reckless behavior, or intentional misconduct. Personal injury law allows the injured person to go to civil court and get a legal remedy (damages) for all losses stemming from the accident or injury. This can include compensation for medical expenses, lost wages, and pain and suffering, among others. Personal injury claims are quite common in lawsuits. Personal injury claims constitute a substantial part of civil litigation. Therefore, having a language model that concentrates on these cases is vital for natural language processing research in the legal field.

The analysis of legal proceedings relies on access to case data, which is often given in the form of legal documents. Processing such documents presents a challenge for general-purpose language models, due to the specialized terminology and syntax conventions in the law. A natural remedy is the development of a specialized legal-domain language model that is trained on legal text (Chalkidis et al., 2020; Xiao et al., 2021). There are three key

*Work done while intern at Claudius Legal Intelligence.

Proceedings of the Natural Legal Language Processing Workshop 2023, pages 51–61

challenges in developing legal-domain models:

1. *Long document for processing:* Extracting key legal information, such as the plaintiff and defendant in a case, requires long-range text comprehension. Most legal texts are much longer than the 512 tokens, the typical limit for BERT-based models. Here, we describe an architecture that increases the maximum passage length to 8,092 tokens.
2. *Specialized terminology from other domains in legal text:* Here, we consider the example of personal injury case data, which often contains medical terminology, such as descriptions of diagnoses and treatments. In such cases, we require a language model that can process specialized text from more than one domain. Thus we train on a mixed legal and medical domain corpus.
3. *Limited access of high-quality training data:* Data on civil legal proceedings is often siloed due to privacy restrictions and there are few publicly available, curated data sets. Hence, we focus on the development of a small, specialized language model, which can be trained, tuned and run on small data sets in a cost-effective way.

LEGALRELECTRA provides a pretrained model for personal injury text with a special focus on enabling long-range text comprehension. It lends itself to a plethora of applications that involve legal case documents, including summarization or extraction of key information for civil suits, identifying patterns and trends in legal proceedings and identifying precedent in past cases, among others. In addition, legal language models may aid in summarizing and analysing legal scholarship. We demonstrate the applicability of LEGALRELECTRA on a downstream task, for which we train a Name Entity Recognition (NER) model. In practise, such an NER model may be applied to extract key legal information from case documents, such as the identities of plaintiff and defendant, medical injuries and civil case type. With that, NER enables a simple summarization of civil suits, which may serve as a basis for further case analytics.

Contributions. Our main contributions are:

1. We describe a novel model architecture (RELECTRA) that adapts the popular ELECTRA model to the processing of long passages. For this, we replace the BERT generators and discriminators with REFORMER.
2. We describe a training procedure for mixed-domain language models that are adapted to processing text from more than one domain.
3. We demonstrate the benefits of training a domain-specific tokenizer as opposed to pre-training with the general-domain tokenizer.

The resulting model, LEGALRELECTRA, is well equipped to process long passages of mixed-domain text (here, personal injury cases, i.e., mixed legal and medical domain), as we demonstrate in a range of benchmark experiments against state of the art general and domain-specific models.

2 Related Work

Recently, there has been growing interest in utilizing Machine Learning in the legal domain (Legal Artificial Intelligence), including for judgment prediction (Chalkidis et al., 2019; Medvedeva et al., 2020), the analysis of fairness in legal proceedings (Kleinberg et al., 2020; Ciocanel et al., 2020; Avery and Cooper, 2020; Sargent and Weber, 2021), as well as legal document analysis (Zhong et al., 2020; Grover et al., 2003; Sulea et al., 2017). The development of specialized legal language models can aid in the latter. Many variations of transformers that are adapted to specialized domains have been proposed in the literature (e.g., (Huang et al., 2019; Chalkidis et al., 2020; Yang et al., 2020; Xiao et al., 2021; Rasmy et al., 2021)) and demonstrated to be more efficient and accurate than BERT on their specialized domains. However, most of these pre-trained domain-specific models do not adopt new frameworks but rely on the classical BERT architecture. To the best of our knowledge, only LAWFORMER (Xiao et al., 2021), which is a Chinese legal domain pre-trained model, utilizes LONGFORMER (Beltagy et al., 2020), instead of BERT. This architecture changes allows LAWFORMER to process longer passages (up to 4,096 tokens). Here, we adopt the ELECTRA (Clark et al., 2019) framework, which has been shown to be more data- and parameter-efficient than BERT. In addition, our model utilizes REFORMER as generator and discriminator, which significantly improves over the maximum text length of LONGFORMER (up to 8,092 tokens). In this paper, we show that the resulting model (LEGALRELECTRA) outperforms BERT, as well as state-of-the-art single-domain adapted models on a downstream task (Named Entity Recognition, sec. 5.3; Legal Case Retrieval,

sec. 5.4;). Moreover, some current state-of-the-art domain-specific language models are pretrained using a general-domain tokenizer (e.g., the BERT tokenizer in LEGAL-BERT) to preprocess the input data. Here, we train a *domain-specific* tokenizer to pre-train LEGALRELECTRA. Our results indicate that this improves the training process and downstream tasks (sec. 5.1 and 5.2).

Lastly, we note that all current domain-specific models focus on adaption to a *single* domain. In contrast, our model is trained on both legal and medical domains motivated by applications in processing personal injury text. Our experimental results demonstrate that LEGALRELECTRA performs competitively against general-purpose and single-domain models (sec. 5.3).

3 Language Modeling for Legal Text

In this section, we outline the architecture of our model, describe its pretraining process, and detail the tasks on which it is assessed.

3.1 Legal Language Modeling

We utilize the ELECTRA framework as basic model architecture. However, in contrast to the classical ELECTRA structure, we replace the BERT generator and discriminator with REFORMER models. This subsection briefly describes the structure of ELECTRA and REFORMER and how they relate to LEGALRELECTRA.

ELECTRA. ELECTRA (Clark et al., 2019) is a sample-efficient model, which learns from all input tokens instead of just the small masked-out subset (as in BERT). It is shown to excel on the GLUE benchmark and multiple downstream NLP tasks, such as question answering. ELECTRA consists of two sub-models, a generator and a discriminator. Given texts with 15% of the tokens masked, the generator is trained to generate the original non-masked text. Then, given the generated text, the discriminator is trained to decide whether any generated token is identical to the original token. Thus, the discriminator loss is calculated over all input tokens as it performs prediction on each token.

REFORMER. In order to be able to process long passages, we leverage REFORMER (Kitaev et al., 2019), which can process text length up to 8,192. This is in stark contrast to BERT, which can only process up to 512 tokens. Traditional transformer models incur computational and memory cost of $O(L^2)$ when computing full attention over a text of length L, creating a significant computational bottleneck. However, computing full attention is unnecessary: The weighted average of attention weights and values involves $\mathrm{softmax}(QK^T)$, which is dominated by the largest elements in a sparse matrix. Thus for each query q, the model only needs to pay attention to the keys k that are closest to q. In contrast, the REFORMER model utilizes locality-sensitive-hashing (LSH) to reduce the complexity of attending over long sequences. LSH is an efficient approach for approximate nearest neighbors search in high dimensions. When using LSH to hash the Q and K matrices, similar q and k vectors are divided into the same buckets. Then standard attention is only computed for the q and k vectors within the same hash buckets.

In our model, the generator and discriminator are two REFORMER models instead of BERT models as in the original ELECTRA framework. We name this new model RELECTRA.

3.2 Pre-Training

Our domain-specific language model specializes in processing personal injury legal text. Personal injury refers to harm to the body, mind, emotions, or reputation, distinct from property damage [1]. Such lawsuits are brought against those responsible due to negligence, misconduct, or intentional harm. The injuries often encompass medical bills, pain and suffering, and reduced quality of life. Consequently, personal injury texts intertwine legal and medical terminologies. Our pre-training corpus draws from public databases, such as CourtListener from the FreeLaw project (The Free Law Project, 2021), and fully anonymized civil case descriptions from proprietary sources. We adopt standard preprocessing methods such as string matching and regular expressions to eliminate special characters, special punctuation, foreign languages, and headers. While a vast collection of personal injury texts would be ideal, accessibility challenges arise due to licensing. As a solution, our primary personal injury text corpus is supplemented with texts from other legal areas and medical content. The inclusion of medical text equips the model with a deeper understanding of personal injuries in a legal context. For a detailed breakdown of our training data sources, refer to Tab. 1. The data set comprises medical (3GB), legal (6GB), and mixed legal-medical (3GB) content, totaling 12GB.

[1] adopted from https://www.law.cornell.edu/wex/personal_injury

Data Set	Size	Description
Legal	6G	Case descriptions from different legal branches; sources include COURTLISTENER (The Free Law Project, 2021).
Medical	3G	Doctor's notes and letters from MIMIC and MIMIC-CXR databases.
Mixed	3G	Personal injury case descriptions from Supreme Court opinions, academic literature, COURTLISTENER, BYU LAW, case descriptions from attorneys.

Table 1: Training data, consisting of a 12GB corpus consisting of legal text, medical text, mixed-domain text with legal and medical terminology, as well as general English text.

3.3 Named Entity Recognition

We validate LEGALRELECTRA by training legal and mixed-domain NER models. Training is performed using automatically (in-house) annotated legal text. In addition, we benchmark LEGALRELECTRA on general and medical domain NER tasks, for which we used publicly available annotated training data conll2003 (Tjong Kim Sang and De Meulder, 2003) (general domain), MIMIC III (Johnson et al., 2016) (medical domain)).

We chose the following labels, which are representative of an NER task that one may encounter in practise: *plaintiff*, *defendant* and *case type*. Here, *case type* categorizes the branch of civil law, which applies to the case. We distinguish among *motorvehicle accidents*, *slip-and-fall accidents* and *work-related cases*, such as illegal termination of a work contract and negligence by a professional. For the latter, the NER task consists of identifying words that help to determine the type of a case. Case types are usually not explicitly mentioned in-text; hence, it can be challenging to categorize civil legal documents. The sources for our test data are described in Table 6 in supplemental A. We will make the validation and test data available after publication, but are not able to publish the training data for license reasons.

Annotation. The manual annotation of the training and testing data was performed as follows: For our legal text sources, we have access to ground truth plaintiff and defendant information from case headers. This allows us to annotate relevant phrases in the text via string matching. For the case type annotations, we create a word list for each case type, which contains frequently occurring phrases that are indicative of the respective case type. We then enrich the lists with synonyms of these phrases. After creating the initial word lists, we removed ambiguous phrases or phrases that appeared in the wordlists of more than one case type. Annotations were done via inexact string matching at first to reduce the workload for annotators, where phrases that were similar to an entry in a word list were labeled with the respective case type. After performing automatic annotations based on header information and case type word list, all cases in training, validation and testing are manually checked by three expert annotators.

3.4 Legal Case Retrieval

The second evaluation task for LEGALRELECTRA involves legal case retrieval, a process that seeks to identify similar legal cases from a given collection based on a provided case description, without any fine-tuning. This task is crucial for legal analysis and education as past cases can profoundly influence the analysis and legal judgment over new cases. However, annotating legal case similarity are challenging due to (1) its reliance on specific expertise and domain knowledge, making crowdsourced annotations difficult (2) law firms often confidentially collect similar cases for future reference, which are typically inaccessible. Consequently, we perform legal case retrieval using the pretrained model, without any fine-tuning on annotated datasets. The dataset for this task comprises 500 cases, selected by law firms from Kentucky (cases from 1998 to 2018) and Louisiana (cases from 2002). Each case encapsulates various information such as claim type, injuries, claim county, plaintiff and defendant names, claim date, medical expenses, future medical expenses, lost salaries, among others. The dataset covers nine general claim types: Parked/Parking Accident, Accident in Intersection, Property Owned by Individual, Medical Malpractice, Truck-Involved Accident, Rear-End Accident, Head-On Accident, Business Negligence, and Retail Establishment.

4 Experiment Setup

This section delineates the experimental configurations for both the pretraining phase and the finetuning for named entity recognition. Since the legal

case retrieval task is assessed without finetuning, we do not provide detailed setup specifics for it.

4.1 Tokenizer

Tokenization refers to the process of splitting a stream of characters into words (Grefenstette and Tapanainen, 1994). While often seen as part of preprocessing, good tokenization is a crucial prerequisite for good downstream performance. Tokenization is particularly important for processing domain-specific text that contains a large amount of specialized terminology. In contrast to most of the published domain-specific language models, which use the standard BERT tokenizer, we train a custom tokenizer. Our tokenizer is trained via standard Byte-Pair Encoding (Sennrich and Birch, 2016), which replaces the most common pair of consecutive bytes of data with a byte that does not occur in that data until the vocabulary size is reached. Since our pre-training data set contains text with specialized terminology, our custom tokenizer generates more sensible tokenization of domain-specific text. An example is given in Tab. 2 (see supplemental).

4.2 Pre-Training

We pre-train LEGALRELECTRA on the collected legal, medical and mixed-domain text corpora described above, in batches of four samples each. Our combined corpora contain up to 30,522 sentence piece tokens. There are 120k training steps and 20k warm-up steps (linear warm-up with linear decay scheduler). We use the ADAMW optimizer with learning rate of 1e-5 for the first 80k training steps and 1e-6 for the remaining 40k training steps. LEGALRELECTRA was trained on one NVIDIA P100 GPU and 16 GB memory, for a total of 16 days, 6 hours and 30 minutess.

4.3 Downstream tasks

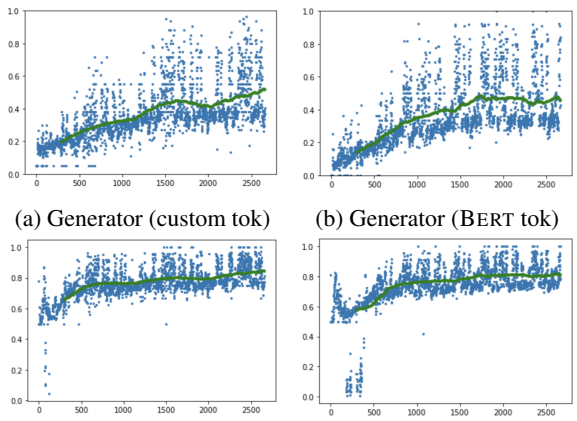
We evaluate LEGALRELECTRA on two downstream tasks, *Named Entity Recognition* (NER) and *Legal case retrival*. We use BERT, CLINICALBERT, LEGAL-BERT and REFORMER as baseline models. Each NER model is trained for 10 epochs with batch size 1, using the AdamW optimizer with learning rate 3e-5.

5 Results

5.1 Tokenizer

The performance of tokenizers is in general difficult to evaluate, as it is highly dependent on the downstream application. To the best of our knowledge, no established convention exists in the literature. Here, we employ quantitative metrics for comparing the performance of our custom LEGALRELECTRA tokenizer with that of the standard BERT tokenizer. For our evaluation, we align the output of the BERT and LEGALRELECTRA tokenizers for ten text segments with personal injury case descriptions. All text segments contain both medical and legal terminology. We analyze the number of recognized words and the number of total unique errors, as well as errors in medical and legal phrases. A similar validation scheme was suggested in (Habert et al., 1998). Our results (see Tab. 7) show that the custom LEGALRELECTRA tokenizer performs better at recognizing words, as evident in the lower number of words detected. We further notice that LEGALRELECTRA tokenizer has a smaller number of errors involving legal and medical terminology, suggesting a superior performance on domain-specific text. Errors that do not involve medical or legal terminology are often (non-English) names for both tokenizers.

5.2 Pre-training



(a) Generator (custom tok) (b) Generator (BERT tok)

(c) Discriminator (custom) (d) Discriminator (BERT)

Figure 1: Masked Language Modeling (MLM) accuracy of generator and discriminator of LEGALRELECTRA with our custom tokenizer and the standard BERT tokenizer. Here, one dot represents the accuracy score of one datapoint (evaluated every 100 datapoints). The line is a smoothed function of averaged accuracy scores over intervals of length 200 aggregating the scores of the 100 data points before and after respectively.

We evaluate LEGALRELECTRA trained with our custom (domain-specific) tokenizer in comparison with a second LEGALRELECTRA model that was trained using the standard (general-domain) BERT tokenizer. We report both generator and discrimi-

Medical	Tokenization result
text	gastrointestinal complaints, neurologic changes, rashes, palpitations, orthopnea
BERT (uncased)	gas, tro, int, estinal, complaints, ne, uro, logic, changes, rash, es, pal, pit, ations, or, th, op, nea
custom (uncased)	gastrointestinal, complaints, neurologic, changes, rashes, palpitations, orthopnea
Legal	Tokenization result
text	the nature of adjudications upon which erroneous subsequent proceedings rest
BERT (uncased)	the, nature, of, ad, ju, dication, s, upon, which, er, rone, ous, subsequent, proceedings, rest
custom (uncased)	the, nature, of, adjudications, upon, which, erroneous, subsequent, proceedings, rest

Table 2: Tokenization example for the BERT tokenizer and our custom tokenizer.

nator accuracy after 120k training steps. The pretraining task for the generator is Masked language modeling (MLM). In MLM, the input is corrupted by replacing some tokens with "[MASK]". Then we train a model to reconstruct the original tokens. The pretraining task for the discriminator is to predict whether each token in the corrupted input was replaced by a generator sample or not. The results for both models are shown in Fig. 1. We observe that the pretraining performance of the generator and discriminator is comparable. Notably, both the generator and discriminator model trained with our custom tokenizer improve over the model trained with the BERT tokenizer towards the end of the training process (Fig. 2 shows the generator-discriminator difference in both models).

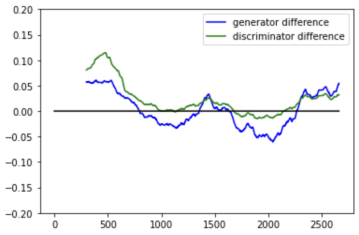


Figure 2: Differences of Generator and Discriminator accuracy between LEGALRELECTRA models pretrained with custom tokenizer and BERT tokenizer.

5.3 Named Entity Recognition

To evaluate LEGALRELECTRA on a downstream task, we analyze the performance of legal domain and mixed legal-medical domain NER models trained with LEGALRELECTRA in comparison with benchmark NER models trained with BERT, LEGAL-BERT, CLINICAL-BERT, REFORMER and LEGALRELECTRA with BERT tokenizer. We consider three labels for the legal domain (case type, plaintiff, defendant) and four labels for the mixed medical-legal domain (case type, plaintiff, defendant, injury). For experiments on BERT, LEGAL-BERT, CLINICAL-BERT which have a maximum token length of 512, we stride and chunk the training and test passage. For experiments on REFORMER, LEGALRELECTRA and LEGALRELECTRA with BERT tokenizer, we use a maximum token length of 1536 evaluated on validation data.

The results are given in Tables 3 (legal domain) and 4 (mixed domain). We observe that LEGALRELECTRA outperforms both the general-domain BERT, as well as the specialized LEGAL-BERT and CLINICAL-BERT models on both domains (with respect to the overall f1 score). It also performs better than REFORMER which is pretrained on the same corpus, demonstrating the benefits of ELECTRA training framework. Notably, both LEGALRELECTRA and REFORMER trained with our custom tokenizer outperform LEGALRELECTRA trained with the BERT tokenizer, which demonstrates the benefits of a domain-specific tokenizer.

5.4 Legal Case Retrieval

To quantitatively assess retrieval performance, we manually review 50 cases, randomly selected from the dataset, and benchmark the top-1 retrieval legal case matching against BERT, LEGAL-BERT, CLINICAL-BERT, and REFORMER. The evaluation is based on five criteria:

1. *Claim type:* Considered a match, if the cases share the same claim type.
2. *Injury categories:* Consider a match, if over half of the injuries in the given case match the retrieved case.
3. *Gender of plaintiff:* Counted as a match, if plaintiffs in both cases share the same gender.
4. *Age of plaintiff:* Considered a match, if the age difference between plaintiffs in the given and retrieved cases is within ±10 years.
5. *Medical expenses:* Counted as a match if the difference in medical expenses between the given and retrieved case falls within a range of ±50

	precision			recall			f1			f1
Legal domain	TYPE	PLT	DEF	TYPE	PLT	DEF	TYPE	PLT	DEF	all
BERT	95.00	82.75	82.86	76.00	64.86	87.88	84.44	72.73	85.29	80.45
CLINICAL-BERT	95.00	86.95	87.88	79.17	58.89	90.63	86.36	70.17	89.23	81.93
LEGAL-BERT	84.70	82.31	88.56	77.11	65.34	80.59	80.73	72.85	84.39	79.32
REFORMER	87.51	73.69	92.33	76.25	87.97	89.31	81.48	80.20	90.79	84.16
LEGALRELECTRA	89.66	84.31	100.00	74.28	84.31	83.33	81.25	84.31	90.91	**85.93**
LEGALRELECTRA (BERT tok)	95.23	85.37	97.56	55.56	67.31	86.96	70.17	75.27	91.95	80.12

Table 3: **NER-legal:** Performance of LEGALRELECTRA on legal text in comparison with BERT, CLINICAL-BERT, LEGAL-BERT and LEGALRELECTRA (BERT tokenizer) for case type (TYPE), defendant (DEF) and plaintiff (PLT).

	precision				recall				f1				f1
Mixed domain	TYPE	PLT	DEF	PROB	TYPE	PLT	DEF	PROB	TYPE	PLT	DEF	PROB	all
BERT	70.59	82.76	71.43	82.61	57.14	64.86	80.65	76.00	63.15	72.72	75.76	79.17	73.39
CLINICAL-BERT	84.00	82.91	75.68	88.46	69.23	58.82	87.32	92.00	75.90	80.08	68.11	90.19	78.55
LEGAL-BERT	88.88	80.00	83.87	85.00	66.67	60.61	86.67	66.67	76.19	68.97	85.25	77.27	77.07
REFORMER	89.92	83.44	86.95	83.21	63.56	62.17	84.38	78.92	74.48	71.25	85.65	81.01	78.10
LEGALRELECTRA	91.30	85.71	95.12	86.96	58.33	57.69	92.86	76.92	71.18	68.97	93.98	81.63	**78.57**
LEGALRELECTRA (BERT tok)	90.47	81.58	95.00	73.91	54.28	59.62	79.17	65.38	67.86	68.89	86.36	69.39	74.20

Table 4: **NER-mixed:** Performance of LEGALRELECTRA in mixed domain with labels case type (TYPE), defendant (DEF) plaintiff (PLT), medical problem (PROB).

Model	Claim	Injury	Gender	Age	Expenses
BERT	80.00	66.00	58.00	48.00	18.00
CLINICAL-BERT	76.00	84.00	64.00	32.00	22.00
LEGAL-BERT	84.00	72.00	68.00	36.00	**32.00**
REFORMER	82.00	78.00	**70.00**	28.00	24.00
LEGALRELECTRA	**90.00**	**92.00**	62.00	**58.00**	30.00
LEGALRELECTRA (BERT tok)	84.00	88.00	60.00	54.00	22.00

Table 5: **Legal Case Retrieval:** Performance of LEGALRELECTRA on legal case retrieval in comparison with BERT, CLINICAL-BERT, LEGAL-BERT and LEGALRELECTRA (BERT tokenizer)

Results are presented in Table 5. We observe that again, LEGALRELECTRA outperforms both the general-domain BERT, specialized LEGAL-BERT and CLINICAL-BERT models, especially on the claim type and injury category matching. LEGALRELECTRA further outperforms LEGALRELECTRA trained with the BERT tokenizer. However, none of the models perform well on numerical information matching, *i.e.* plaintiff age and medical expenses.

5.5 Result Analysis

For both the NER task and the legal case retrieval task, a key observation in our experimental results is the superior performance of LEGALRELECTRA in comparison with LEGAL-BERT. We will provide a more detailed analysis of the two tasks below.

NER. We notice that, surprisingly, LEGAL-BERT does not perform as well as expected on legal case terminology, specifically the recognition of plaintiff and defendant. For example, it may confuse other entities, such as the defendant's attorney, as defendant or plaintiff. While the extracted information is relevant to the case, it is not precise enough to be useful for further downstream analysis. Secondly, LEGAL-BERT does not perform well on case type recognition in legal text (LEGALRELECTRA does not perform well on case type recognition in mixed text either, see below). Thirdly, LEGAL-BERT misses medical terminology, as it is not pretrained on medical data. Examples include TMJ (an abbreviation for for Temporomandibular joint), Myofascial Pain Syndrome and psychological injuries such as emotional distress. CLINICAL-BERT does not perform well on legal case terminology, such as plaintiff and defendant recognition. This is expected, as it is not pretrained on legal text. However, it indeed outperforms all other models on medical terminology recognition, including LEGALRELECTRA, which is also pre-trained on some medical data. It is unexpectedly good at the recognition of case types. BERT sometimes misses plaintiff or defendant throughout the whole case text, indicating its limited ability to recognize legal terminology. In summary, all three models show a limited ability to recognize legal terminology in *long* texts. Errors arise due to the limited number of tokens that the models can process. In some texts the plaintiff and defendant information is only given in the beginning of the text and can therefore

not be obtained from partial text segments. In contrast, LEGALRELECTRA's ability to process larger text segments results in a superior performance in plaintiff and defendant recognition. Its performance on medical problem recognition is better than all other models except CLINICAL-BERT, indicating that the mixed pre-trained data does help on both legal and medical feature learning. REFORMER performs slightly worse on precision but better on recall, but an overall worse result on F1, indicating that the ELECTRA training framework indeed improves model performance. Comparing LEGALRELECTRA trained with our custom tokenizer and the standard BERT tokenizer, we notice that the performance on legal feature recognition is comparable. However, the recognition of medical terminology is negatively affected by the BERT tokenizer. This indicates that wrong tokenization of medical terminology affects the model performance more negatively than wrong tokenization of legal terminology.

Legal Case Retrieval. The LEGALRELECTRA model demonstrates robust performance in claim type and injury category matching, excelling particularly in traffic-related cases like Rear-End and Truck-Involved Accidents with a 96.00 accuracy score. Its ability to match Business Negligence and Property Owned by Individual is less effective, likely due to insufficient training on business-related cases and the relative rarity of these claim types. The model's skill in injury matching surpasses that of CLINICAL-BERT, identifying both singular injuries such as 'Headache' and 'Disc: Herniated or Ruptured', and composite injuries like 'Nonfracture Injury, Strain: Lumbar Only, Aggravation of Preexisting Back or Neck Strain Disc: Bulging'. However, the model fails to adequately consider key factors like gender, age, and medical expenses, despite their relevance in legal case analysis. This omission could be due to their brief textual representations, thus being overlooked by the pretrained language model unless specifically tuned to prioritize such information.

6 Conclusions

In this paper, we introduced LEGALRELECTRA, a language model that is specialized to process mixed legal and medical domain (personal injury) text. This was achieved by pre-training with a corpus consisting of legal, medical and mixed domain (personal injury) text. We demonstrate in validation experiments that LEGALRELECTRA outperforms general-purpose language models (e.g., BERT), as well as specialized legal-domain models (e.g., LEGAL-BERT) on legal and mixed-domain NER. As technical contributions, we proposed a novel model architecture that allows for improved performance on long-range text comprehension.

LEGALRELECTRA provides a pretrained model for personal injury text with a special focus on enabling long-range text comprehension. It lends itself to a plethora of applications that involve legal case documents, including summarization or extraction of key information for civil suits, identifying patterns and trends in legal proceedings, identifying precedent in past cases, among others. In addition, legal language models may aid in summarizing and analysing legal scholarship.

We demonstrate the applicability of LEGALRELECTRA in downstream tasks by training a Named Entity Recognition (NER) model and utilizing the pretrained model directly as a Legal Case Retriever. In practice, the NER model can extract crucial legal information from case documents, including plaintiff and defendant identities, medical injuries, and civil case types. This enables a basic summary of civil suits, serving as a foundation for further case analysis. The Legal Case Retriever, on the other hand, can identify past cases similar to new ones, providing valuable reference points.

7 Limitations

There are several limitations in our training and validation setup, addressing of which may lead to significant improvements. As discussed above, the ideal pre-training corpus for a personal injury language model would consist of large collections of personal injury text. However, due to the restricted access to such data, it is difficult to collect a sufficiently large text corpus. Thus, we supplement our pre-training data with text from other legal branches and (pure) medical text, which may have decreased the model's performance. Further limitations arise in the performance evaluation presented here. Testing and validation against benchmarks could have been more extensive, for instance by evaluating LEGALRELECTRA on a larger and more comprehensive dataset on both NER and retrieval. Moreover, the downstream training data annotation (see sec. 3.3) was partially automated. Instead of string matching, we could have annotated the NER training data manually, which would have been more accurate.

8 Ethical Considerations and Broader Impacts

The language model proposed in this work is designed for processing legal documents in personal injury cases. As such, the model has the potential to streamline personal injury attorney's work, including but not limited to significant time savings during legal proceedings. At the same time, ethical and privacy considerations are crucial when deploying AI technology in the legal space. In particular, if applied in practise, the user should be conscious of potential biases in legal documents, such as court opinions, that were part of the training data and how this may impact the predictions of the model. All legal case documents and medical documents used to train and test the models in this paper were fully anonymized. In the interest of transparency, we worked with publicly available data whenever possible. The size, composition and preprocessing of our data sets is documented in detail in the main text. We document and cite the source of all publicly available data.

References

Joseph Avery and Joel Cooper. 2020. Technology in the legal system. *Bias in the Law: A Definitive Look at Racial Prejudice in the US Criminal Justice System*, 161.

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

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

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

Maria-Veronica Ciocanel, Chad M. Topaz, Rebecca Santorella, Shilad Sen, Christian Michael Smith, and Adam Hufstetler. 2020. Justfair: Judicial system transparency through federal archive inferred records. *PLOS ONE*, 15(10):1–20.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Gregory Grefenstette and Pasi Tapanainen. 1994. What is a word, what is a sentence? problems of tokenization.

Claire Grover, Ben Hachey, and Chris Korycinski. 2003. Summarising legal texts: Sentential tense and argumentative roles. In *Proceedings of the HLT-NAACL 03 Text Summarization Workshop*, pages 33–40.

Benoit Habert, Gilles Adda, Martine Adda-Decker, P Boula de Marëuil, Serge Ferrari, Olivier Ferret, Gabriel Illouz, and Patrick Paroubek. 1998. Towards tokenization evaluation. In *Proceedings of LREC*, volume 98, pages 427–431.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. Clinicalbert: Modeling clinical notes and predicting hospital readmission.

Quzhe Huang, Mingxu Tao, Zhenwei An, Chen Zhang, Cong Jiang, Zhibin Chen, Zirui Wu, and Yansong Feng. 2023. Lawyer llama technical report. *arXiv preprint arXiv:2305.15062*.

A Johnson, L Bulgarelli, T Pollard, S Horng, LA Celi, and R Mark. 2020. Mimic-iv (version 1.0).

Alistair E W Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):1–8.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li wei H. Lehman, Mengling Feng, Mohammad Mahdi Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3.

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. 2019. Reformer: The efficient transformer. In *International Conference on Learning Representations*.

Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Cass R. Sunstein. 2020. Algorithms as discrimination detectors. *Proceedings of the National Academy of Sciences*, 117(48):30096–30100.

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. volume 36, pages 1234–1240. Oxford University Press.

Masha Medvedeva, Michel Vols, and Martijn Wieling. 2020. Using machine learning to predict decisions of the european court of human rights. *Artificial Intelligence and Law*, 28(2):237–266.

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):1–13.

Jackson Sargent and Melanie Weber. 2021. Identifying biases in legal data: An algorithmic fairness perspective. *arXiv preprint arXiv:2109.09946*.

Barry Haddow Sennrich, Rico and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *2016 Association of Computational Linguistics*.

Octavia-Maria Sulea, Marcos Zampieri, Shervin Malmasi, Mihaela Vela, Liviu P Dinu, and Josef Van Genabith. 2017. Exploring the use of text classification in the legal domain. *arXiv preprint arXiv:1710.09306*.

The Free Law Project. 2021. Courtlistener.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

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*.

Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. 2021. Lawformer: A pre-trained language model for chinese legal long documents. volume 2, pages 79–84. Elsevier.

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

Li Yunxiang, Li Zihan, Zhang Kai, Dan Ruilong, and Zhang You. 2023. Chatdoctor: A medical chat model fine-tuned on llama model using medical domain knowledge. *arXiv preprint arXiv:2303.14070*.

Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. How does NLP benefit legal system: A summary of legal artificial intelligence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics.

A Test data

Data Set	Description
Legal	Case descriptions from different legal branches; sources include COURTLISTENER (The Free Law Project, 2021).
Medical	MIMIC (Johnson et al., 2020) MIMIC-CXR (Johnson et al., 2019)
Mixed	Open-source personal injury case descriptions, sources include COURTLISTENER and BYU LAW LIBRARY[2].

Table 6: Test data source

Our test data for the general and medical domains is from the same source as the training data (see Tab. 6). For the legal domain training data, we collected text from public data bases (including COURTLISTENER from the FreeLaw project (The Free Law Project, 2021)).

B Additional experimental results for tokenizer evaluation

We analyze the number of recognized words and the number of total unique errors, as well as errors in medical and legal phrases. A similar validation scheme was suggested in (Habert et al., 1998). Errors in abbreviations are excluded from the error count. The results can be found in table 7.

	BERT tokenizer				Custom tokenizer			
	words	total errors	legal	medical	words	total errors	legal	medical
# 0	153	5	0	1	139	0	0	0
# 1	411	3	0	2	417	3	0	1
# 2	285	12	5	2	277	7	1	0
# 3	418	9	0	7	412	6	0	2
# 4	313	9	3	2	289	4	0	1
# 5	405	9	1	3	385	8	0	1
# 6	216	9	3	4	210	4	0	3
# 7	560	12	5	4	539	11	0	7
# 8	400	13	4	3	407	7	1	0
# 9	340	12	1	5	323	7	0	1

Table 7: **Tokenizer:** Evaluation of custom LEGALRELECTRA tokenizer against BERT tokenzier for ten text segments of personal injury case descriptions. We report the number of words recognized by the tokenizers, as well as the number of unique errors. In addition to the total number of errors, we report the number of errors for medical and legal terminology separately.