# Information Extraction in the Legal Domain: Traditional Supervised Learning vs. ChatGPT

Gustavo M. C. Coelho[1][a], Alimed Celecia[1][b], Jefferson de Sousa[1][c], Melissa Lemos[1][d], Maria Julia Lima[1][e], Ana Mangeth[2][f], Isabella Frajhof[2][g] and Marco Casanova[1][h]

[1] *Tecgraf - PUC-Rio, Rio de Janeiro, Brazil*

[2] *LES - PUC-Rio, Rio de Janeiro, Brazil*

Keywords: Natural Language Processing, Information Extraction, Text Classification, Named Entity Recognition, Large Language Models, Prompt Engineering.

Abstract: Information Extraction is an important task in the legal domain. While the presence of structured and machine-processable data is scarce, unstructured data in the form of legal documents, such as legal opinions, is largely available. If properly processed, such documents can provide valuable information about past lawsuits, allowing better assessment by legal professionals and supporting data-driven applications. This paper addresses information extraction in the Brazilian legal domain by extracting structured features from legal opinions related to consumer complaints. To address this task, the paper explores two different approaches. The first is based on traditional supervised learning methods to extract information from legal opinions by essentially treating the extraction of categorical features as text classification and the extraction of numerical features as named entity recognition. The second approach takes advantage of the recent popularization of Large Language Models (LLMs) to extract categorical and numerical features using ChatGPT and prompt engineering techniques. The paper demonstrates that while both approaches reach similar overall performances in terms of traditional evaluation metrics, ChatGPT substantially reduces the complexity and time required along the process.

## 1 INTRODUCTION

The prolonged duration of a legal case within the Brazilian courts presents a challenge for legal professionals and society. The high volume of new legal cases yearly submitted to the courts, combined with the existing backlog, adds complexity to this matter, encouraging the automation of processes in this context. Over the past years, efforts have been made to address this issue using Artificial Intelligence as a tool to increase court efficiency, switching from knowledge-representation techniques to machine-learning-based approaches. Like most data-driven methods, this approach requires high-quality, structured machine-processable data, which is generally scarce in the legal domain (Surden, 2018). On the other hand, unstructured data in the form of legal documents, such as legal opinions, is largely available. If properly processed, such documents can provide valuable structured information that can be used to describe each legal case. The description of legal cases by a structured and interpretable dataset can be further used in a variety of applications, such as Similar Case Matching (Xiao et al., 2019), Legal Judgment Prediction (Zhong et al., 2018), Recommendation Systems, and other data-driven applications.

More recently, the popularization of Large Language Models (LLMs) such as GPT (Radford et al., 2018) and the introduction of instruction-following LLMs such as ChatGPT[1] have caused a significant impact on the NLP field by simplifying many of these

[a] https://orcid.org/0000-0003-2951-4972
[b] https://orcid.org/0000-0001-9889-795X
[c] https://orcid.org/0000-0002-5928-9959
[d] https://orcid.org/0000-0003-1723-9897
[e] https://orcid.org/0000-0003-3843-021X
[f] https://orcid.org/0000-0003-1624-1645
[g] https://orcid.org/0000-0002-3901-4907
[h] https://orcid.org/0000-0003-0765-9636

[1] https://openai.com/blog/ChatGPT

Coelho, G., Celecia, A., de Sousa, J., Lemos, M., Lima, M., Mangeth, A., Frajhof, I. and Casanova, M.
Information Extraction in the Legal Domain: Traditional Supervised Learning vs. ChatGPT.
DOI: 10.5220/0012499800003690

In *Proceedings of the 26th International Conference on Enterprise Information Systems (ICEIS 2024) - Volume 1*, pages 579-586
ISBN: 978-989-758-692-7; ISSN: 2184-4992

tasks with the use of prompt engineering techniques. This work addresses information extraction from text by comparing two approaches: the traditional Supervised Learning approach, including Text Classification and Named Entity Recognition (NER), and the ChatGPT Prompt Engineering approach, which uses the prompt engineering principles applied to ChatGPT to perform the same tasks.

More specifically, this article is positioned in the context of the automated analysis of legal opinions related to consumer complaints. A *legal opinion* is "a written explanation by a judge or group of judges that accompanies an order or ruling in a case, laying out the rationale and legal principles for the ruling"[2]. A *consumer complaint* is "an expression of dissatisfaction on a consumer's behalf to a responsible party"[3]. In such cases, the legal opinion contains specific provisions referring to the plaintiff's claim, such as moral damage, material damage, and legal fees due by the defeated party. The term *legal opinion* is restricted to this particular context in what follows.

To evaluate the proposed approaches, we use a specially created dataset containing 959 manually annotated legal opinions (in Brazilian Portuguese) enacted by lower court judges in the State Court of Rio de Janeiro in the context of consumer complaints involving electric power companies.

The rest of the article is organized as follows. Section 2 introduces background concepts and summarizes related work. Section 3 describes information extraction from text focusing on Text Classification, NER, and ChatGPT Prompt Engineering. Section 4 describes the experiments and compares the results. Finally, Section 5 presents the conclusions and directions for future research.

## 2 RELATED WORK

### 2.1 Text Classification in the Legal Domain

Text Classification is an important task in information extraction. Sulea et al. (2017) argue that using Text Classification, as in various other domains, can benefit legal professionals by providing a decision support system or at least a sanity check system. The proposed framework uses word unigrams and word bigrams as features and an ensemble classifier as the classification model. The goal is to assign each legal document to one class from a pre-defined set of classes. The results show a 98% average F1-score in predicting a case ruling, 96% for predicting the law area of a case, and 87.07% for estimating the date of a ruling.

Minaee et al. (2021) explore the use of deep learning in text classification by listing more than 150 deep learning-based models. The list includes feed-forward networks, RNN and CNN-based models, graph neural networks, and hybrid models. The survey shows that deep learning-based models surpass classical machine learning-based approaches, improving state of the art on various Text Classification tasks.

In the Brazilian Legal Domain, De Araujo et al. (2020) introduced a dataset built from Brazil's Supreme Court digitalized legal documents, composed of more than 45 thousand appeals, which includes roughly 692 thousand documents. The documents contain labels related to the document type and lawsuit theme. The baseline adopted comprises bag-of-words models, CNNs, Recurrent Neural Networks (RNNs), and boosting algorithms. The results show that CNN and Bidirectional Long Short-Term Memory (BiLSTM) outperform the remaining models in all categories, emphasizing the potential of deep learning approaches in this task.

### 2.2 Named Entity Recognition in the Legal Domain

The extraction of named entities is a frequent approach for information extraction in the legal domain, where entities such as persons, organizations, and locations are combined with entities related to the legal context. Leitner et al. (2019) addressed this task by extracting several fine-grained semantic entities, such as company, institution, court, and regulation. Models based on BiLSTM and Conditional Random Field (CRF) are applied to the task, with character embedding. The results of both model families demonstrate that BiLSTMs models outperform CRF with an F1-score of 95.46% for the fine-grained classes and 95.95% for the coarse-grained classes.

In the Brazilian legal context, Fernandes et al. (2022) proposed a set of NER models to extract information from legal opinions enacted by lower and Appellate Courts. More specifically, three datasets were built to identify legal entities, such as the moral and material damage values, the legal fee due by the defeated party, and others. Five models were proposed based on different combinations of word and character embeddings, RNNs, and CRFs. The optimal results reached by the models range from 68.42% to 90.43%, depending on the dataset.

[2] https://en.wikipedia.org/wiki/Legal_opinion

[3] https://en.wikipedia.org/wiki/Consumer_complaint

Fernandes et al. (2020) extracted modifications proposed by the Brazilian Upper Court to Lower Court judges' decisions. The task was performed by first defining six entities that correspond to the most popular legal categories that the Appellate Court modifies in a specific legal domain. The extraction of these entities was evaluated by five models based on different combinations of RNNs and CRFs, and the best performance was reached by combining a BiLSTM and a CRF layer.

## 2.3 LLM Prompt Engineering

With the recent advances in instruction-following LLMs, prompt engineering techniques have been extensively explored as a new paradigm in NLP tasks. Dong et al. (2022) provides a survey of advanced in-context learning techniques, exploring approaches for the description of clear instructions, the selection of examples to be demonstrated, and the prompts formatting.

To automate the construction of LLM prompts, Zhou et al. (2022) proposes a method that optimizes the prompt construction by searching over a pool of instruction candidates that are created by an LLM. The optimal prompts are chosen based on the maximization of a certain score function, resulting in an efficient approach for reaching human-level performance on various tasks with minimum human inputs.

In addition to prompt engineering techniques focused on the creation of clear instructions for specific tasks, LLMs present a high ability for reasoning. Yao et al. (2022) explores this ability by creating an approach named ReAct. This approach is based on the concept of following a sequence of steps that involves reasoning over each step and acting accordingly until the task is finalized. The method enables LLMs to recover from mistakes along the process and decreases the chances of hallucinations, which is a common and known issue related to language models.

# 3 MODELS

## 3.1 The Supervised Learning Approach

The Supervised Learning approach refers to the use of traditional machine-learning techniques which involve the optimization of model parameters based on an annotated dataset. The need for extracting both numerical and categorical provisions from legal opinions leads to the use of two different model categories, according to their tasks: Text Classification and Named Entity Recognition.

### 3.1.1 Text Classification

Four different Text Classification models were implemented during the experiments, based on two frameworks. The first three models are based on Kowsari et al. (2019), which summarizes most text classification systems as a three-step procedure. The first step converts textual units into fixed-length numerical vectors by using a feature extraction model. The second step covers an optional dimensionality reduction over the results of the first step, which is potentially high dimensional, depending on the feature extraction model applied. The third step consists of a classification model, such as Naïve Bayes, support vector machines (SVM), and random forests. In this step, each reduced feature vector referred to a document is classified in one of the pre-defined classes.

The three models applied based on this framework differ from each other according to the feature extraction method applied in the first step. The TF-IDF Classifier, SIF Classifier, and Doc2vec Classifier are based on respectively TF-IDF (Jones, 1972), SIF (Arora et al., 2017) and Doc2vec (Le and Mikolov, 2014) as the feature extraction methods and Logistic Regression as the classifier. The dimensionality reduction step didn't result in significant advantages during our experiments and was thus bypassed in all models.

The fourth model, the C-LSTM Classifier, is based on the C-LSTM framework (Zhou et al., 2015), which is a neural network approach for text representation and classification. The strategy used by this model combines CNN and LSTM layers. Since CNNs and LSTMs adopt different ways of understanding natural language, they work in different roles inside this framework. While the CNN layer is used to capture a sequence of higher-level phrase representations, the LSTM layer captures global and temporal semantics. Thus, C-LSTM can map both word semantics (with the use of word embeddings) and local and global contextual information from text instances.

### 3.1.2 Named Entity Recognition

The NER model used during the experiments is based on a framework described by Souza et al. (2019), which proposes a BERT model for a Portuguese NER task. The model's training process can vary in two main approaches. The fine-tuning approach uses a linear layer as the classifier and all weights are optimized jointly during training, including BERT, classifier, and CRF weights. The feature-based approach uses a 1-layer BiLSTM model as the classifier. This approach freezes the BERT weights during training, while the classifier and CRF are optimized.

Using this framework as a basis, the experiments assessed four models with two key differences: whether they employed a CRF layer or not; and the training approach (fine-tuning or feature-based).

### 3.2 The ChatGPT Prompt Engineering Approach

The ChatGPT Prompt Engineering approach for extracting information from text documents leverages ChatGPT capabilities with a simple setup. By providing clear and fairly simple instructions to an instruction-following LLM, specific information can be extracted from a piece of text without the need for time-consuming model optimizations. In summary, given a set of text documents and specific information required to be extracted from them, a prompt is manually constructed containing mainly three items:

1. Details of the information to be extracted.
2. The output format.
3. Some optional examples.

The first item refers to the explanation of the information to be extracted. The level of detail involved in this explanation depends on how domain-specific the information is.

In the second item, ChatGPT is instructed on how to respond. When extracting categorical features, a set of possible values is described, and when extracting numerical features, the format of the numerical values is specified. In addition, since instruction-following LLMs are fine-tuned for conversational responses, this item usually involves instructions for direct answers, avoiding unnecessary post-processing of the LLM's response. As a strategy for lowering the probability of model hallucination, which is known to be a common issue for instruction-following LLMs, it turned out to be a good strategy to instruct ChatGPT to return not only the result but also a text segment where the result was based on. This part of the output is later discarded.

Finally, an optional third item can be specified with the description of examples, where the desired information is correctly extracted from examples of text instances. Similarly to the first step, the need for describing examples is related to how domain-specific the information to be extracted is. The use of this step is what typically differentiates a zero-shot prediction (where no examples are provided in the prompt) and a few-shot prediction (where some examples are described).

Following this approach, two types of models were implemented. The extraction of categorical provisions was addressed by the ChatGPT Classifier, which uses a specific prompt for each categorical provision present in legal opinions, following the steps described above. Similarly, the extraction of the numerical provision is addressed by the ChatGPT Entity Extractor, with one relevant difference regarding the prompt creation: in this case, the output was not restricted to a list of possible values since it should reflect the exact number associated with the numerical provision.

The experiments used GPT-3.5-turbo for both types of models.

## 4 EXPERIMENTS

### 4.1 Experimental Setup

Following the Supervised Learning approach for Text Classification, the four models adopted (TF-IDF, SIF, Doc2vec, and C-LSTM Classifiers) were trained and evaluated in a 10-fold cross-validation setup. To establish the main hyperparameters on each model, such as Logistic Regression penalties, vector dimensions, and others, a Bayesian Optimizer (Snoek et al., 2012) was used, where each of these hyperparameters is defined as input search dimensions for the optimization of the objective function, which is defined as the average 10-fold cross validation F1-Score.

This setup is not valid for the ChatGPT Prompt Engineering approach, since no training is required. Instead, the entire dataset was simply used for evaluation.

As a prior step for each model, common preprocessing routines are applied. This is an especially important step considering that legal opinions are structured differently from other domains. When considering the supervised models, this step includes lowercasing the text and removing the punctuation, line breakers, and excessive spaces. In addition, to minimize the task's complexity, the document is filtered to contain only the operative part of the judgment, where the lower or Appellate Court judge presents the judicial solution to the lawsuit. To identify this part, which is located at the end of the document, a list of regular expressions is used.

After the identification of the operative part and the removal of the remaining document, the stopwords are removed and the words are tokenized. Lastly, the tokens are stemmed to decrease the variety of expressions with different suffixes, resulting in a more simple representation of the text. The preprocessing step is therefore highly dependent on the type of legal document in question and must be adjusted accordingly for other contexts.

## 4.2 Dataset

The experiments used a dataset containing 959 manually annotated legal opinions (in Brazilian Portuguese) enacted by lower court judges in the State Court of Rio de Janeiro in the context of consumer complaints involving electric power companies. Each legal opinion in the dataset was manually analyzed by legal professionals to locate four types of provisions. Three of the provisions are categorical and are described as follows: *case ruling* refers to "a court's decision on a matter presented in a lawsuit"[4]. *Restoration of supply* indicates if the electric company should reestablish the plaintiff's supply. *Restitution* establishes if the electric company should refund excessive monetary charges paid by the plaintiff, and whether the refund value should be doubled or not.

In this work, the moral damage compensation is the only numerical provision addressed. Along with the value associated with the given moral damage, each document was POS tagged, denoting the position in the text where the values are expressed.

## 4.3 Results

### 4.3.1 Extraction of Categorical Provisions

After hyperparameters optimization, the supervised learning models were applied to the dataset to evaluate their main average metrics in a 10-fold stratified cross-validation setup. For comparison, the precision, recall, F1-score, and accuracy were extracted.

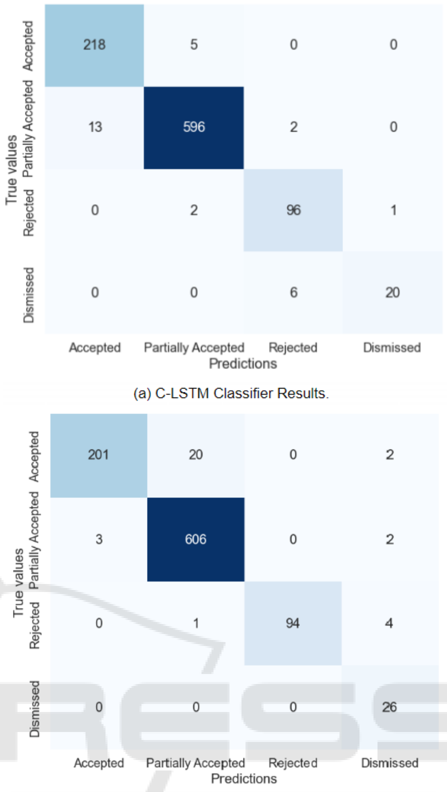
By contrast, each ChatGPT Classifier result is based on one iteration over the entire dataset, since no training data is required.

Table 1 shows the mean results for each provision, where the highlighted lines represent the best models per provision by their mean F1-Score.

The overall results indicate that the Doc2vec, C-LSTM and ChatGPT Classifiers reached the best performances, while the TF-IDF and SIF Classifiers generally had the worst performances. This is an expected result, given the corresponding models' complexities.

Interestingly, the ChatGPT Classifier showed very competitive results compared to the remaining models. This is especially relevant considering the convenient nature of the implementation of the ChatGPT models, the main argument this paper supports.

For this reason, for each task, the rest of this section presents confusion matrices for the best-supervised learning model and the ChatGPT Prompt Engineering approach. The confusion matrix associated with a supervised model is based on the sum of validation sets of the 10-fold cross-validation results.



(a) C-LSTM Classifier Results.

(b) ChatGPT Classifier Results.

Figure 1: Comparison between C-LSTM and ChatGPT Classifiers for case ruling.

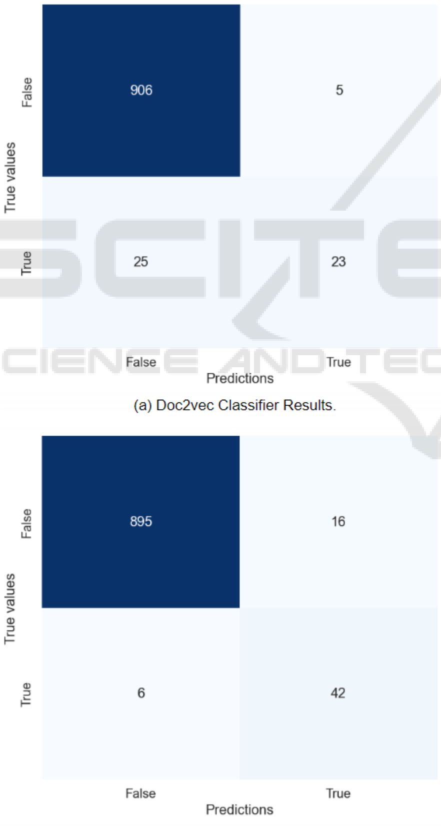
For *case ruling*, Figure 1 shows the confusion matrices related to the C-LSTM and the ChatGPT classifiers. The concentration of errors between "Partially Accepted" and "Accepted" can be explained by a particularity of this provision in the text. Case rulings can be expressed as a summary of each plaintiff's claim. For instance, a rejection of the moral damage claim does not necessarily translate into the entire case being rejected, since the remaining claims might have been accepted, that is, the case ruling was partially accepted. The sequential and more complex representation of these expressions in the C-LSTM model proved to better capture the correct context in this case. The ChatGPT classifier, however, reached very similar metrics, demonstrating to be competitive, compared to the best supervised learning model.

For *restoration of supply*, Figure 2 shows the confusion matrices for the Doc2vec and the ChatGPT classifiers. Note that the errors of the Doc2vec classifier are concentrated on the minority class "True". Indeed, the extraction of the provision *restoration of*

[4] https://www.law.cornell.edu/wex/ruling

Table 1: Results for the classification of categorical provisions.

| Provision | Classifier | Precision | Recall | F1-score | Accuracy |
|---|---|---|---|---|---|
| *Case ruling* | TF-IDF | 0.707 | 0.691 | 0.604 | 0.691 |
| | SIF | 0.872 | 0.870 | 0.863 | 0.870 |
| | Doc2vec | 0.893 | 0.893 | 0.890 | 0.893 |
| | **C-LSTM** | **0.969** | **0.970** | **0.968** | **0.970** |
| | ChatGPT | **0.969** | 0.967 | 0.967 | 0.967 |
| *Restoration of supply* | TF-IDF | 0.946 | 0.961 | 0.949 | 0.961 |
| | SIF | 0.959 | 0.963 | 0.955 | 0.963 |
| | Doc2vec | 0.964 | 0.969 | 0.963 | 0.969 |
| | C-LSTM | 0.953 | 0.961 | 0.954 | 0.961 |
| | **ChatGPT** | **0.980** | **0.977** | **0.978** | **0.977** |
| *Restitution* | TF-IDF | 0.618 | 0.556 | 0.492 | 0.556 |
| | SIF | 0.887 | 0.875 | 0.874 | 0.875 |
| | Doc2vec | 0.929 | 0.925 | 0.925 | 0.925 |
| | **C-LSTM** | **0.974** | **0.973** | **0.973** | **0.973** |
| | ChatGPT | 0.934 | 0.930 | 0.930 | 0.930 |



(a) Doc2vec Classifier Results.

(b) ChatGPT Classifier Results.

Figure 2: Comparison between Doc2vec and ChatGPT Classifiers for restoration of supply.

*supply* is challenging due to its extreme imbalance. Although the Doc2vec model reached the highest F1-score among the supervised models, the F1-Score of 96.3% can be misleading since the model classified over 50% of the instances in class “True” as “ False”, indicating the need for a large number of training instances for effective implementation. Interestingly, the ChatGPT classifier, which reached the best performance among all models, seems to be less affected by the class imbalance issue. This can be explained by the fact that an LLM prompt engineering approach does not rely on a training dataset, and thus it is not affected by an imbalanced training set. In addition, restoration of supply is a fairly simple concept, which implies that examples or a more complex prompt description are not required.

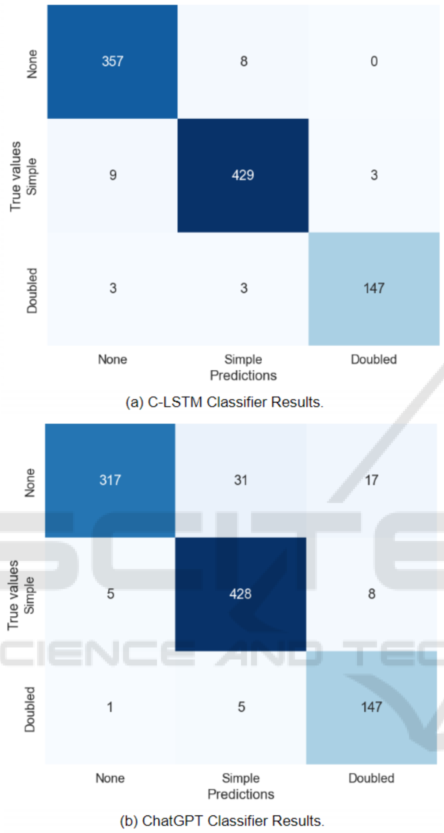
For *restitution*, Figure 3 shows the confusion matrices related to the Doc2vec and the ChatGPT classifiers. In opposition to *restoration of supply*, the provision *restitution* presents a balanced class distribution, which results in better error distribution. The C-LSTM model had the best performance for the extraction of this provision. Although the restitution of monetary values is a simple concept, the ChatGPT classifier was roughly 4% behind in terms of F1-Score. The main reason for this lower performance is the frequent misleading ”interpretation” of moral damage compensation as restitution, which are different concepts in this context. Indeed, Figure 3(b) clearly shows that the majority of errors of the ChatGPT classifier confusion matrix are located where the true value is “None”, meaning that no restitution was determined, but the predicted value was “Simple” or “Doubled”.

### 4.3.2 Extraction of Numerical Provision

The extraction of the numerical provision (moral damage compensation) adopted the BERT model for NER, implemented using four different approaches. The first, simply named BERT, used a fine-tuning ap-

Table 2: Results for the extraction of moral damage compensations.

| Model | Training approach | Accuracy | RMSE |
|---|---|---|---|
| BERT | Fine-tuning | 0.962 | 825.8 |
| BERT-CRF | Fine-tuning | 0.982 | 639.8 |
| BERT-LSTM | Feature-based | 0.954 | 738.6 |
| **BERT-LSTM-CRF** | **Feature-based** | **0.989** | **277.2** |
| ChatGPT Entity Extractor | NA | 0.984 | 1230.9 |



(a) C-LSTM Classifier Results.

(b) ChatGPT Classifier Results.

Figure 3: Comparison between Doc2vec and ChatGPT Classifiers for restitution.

proach (i.e., updating all weights jointly and using a linear layer as a classifier) without a CRF layer. The second, named BERT-CRF, is similar to the first approach, with the addition of a CRF layer. The third, named BERT-LSTM, used a feature-based approach (i.e freezing the BERT weights and using a BiLSTM as the classifier) without a CRF layer. The fourth approach, named BERT-LSTM-CRF, added a CRF layer to the third approach.

The ChatGPT Entity Extractor was implemented as described on Section 3.2.

Table 2 shows the accuracy of each model and the corresponding Root Mean Squared Errors (RMSEs).

The results demonstrate the effectiveness of the BERT model for NER and the ChatGPT entity extractor for the extraction of the moral damage value from legal opinions. The mean accuracy ranges from 96.2% and 98.9% within the different models. As a natural result, the CRF layer enhances the performance by around 2%, indicating the effectiveness of the contextual information captured by the CRF. Interestingly, the feature-based approach outperforms the Fine-tuning approach by 0.75% on average. This result possibly indicates the quality of BERT embeddings, achieving better results when the weights are frozen, which is unexpected, given the specificity of the Legal Domain context.

The ChatGPT Entity Extractor results are similar to the best-supervised model in terms of accuracy but significantly worse regarding RMSE. The high accuracy is closely related to the simplicity of the provision's concept. Moral damage compensation is a fairly known provision, and its description in a legal opinion is rarely ambiguous. The rare cases where the ChatGPT Entity Extractor fails to extract the correct value are due to the wrong identification of other types of compensations, such as material damages, legal fees, or even the moral damage requested by the plaintiff, and not given by the judge. This results in higher RMSE values when compared to the supervised model, which was exposed to these ambiguities during training.

## 5 CONCLUSIONS

The most direct contribution of this work is the development of a highly accurate tool for extracting provisions from legal opinions in the given context. In addition, it offered a practical comparison between the traditional supervised learning approach and the LLM prompt engineering approach.

The best models found during evaluation achieved a mean accuracy higher than 96% for the extraction of each provision. Despite this high accuracy, It is important to note the large imbalance in the dataset when categorized by the restoration of supply.

Although traditional supervised learning methods achieved the best results, the ChatGPT Prompt Engineering approach reached competitive results, with the advantage of requiring a significantly less complex implementation setup. Indeed, while traditional methods required extensive work for model definition, training, hyperparameter optimization, etc., the ChatGPT Prompt Engineering approach required an adequate prompt definition, which reduced the implementation time from months to days.

## ACKNOWLEDGEMENTS

This work was partly funded by FAPERJ under grant E-26/202.818/2017; by CAPES under grants 88881.310592-2018/01, 88881.134081/2016-01, and 88882.164913/2010-01; and by CNPq under grant 302303/2017-0.

## REFERENCES

Arora, S., Liang, Y., and Ma, T. (2017). A simple but tough-to-beat baseline for sentence embeddings. In *International conference on learning representations*.

De Araujo, P. H. L., de Campos, T. E., Braz, F. A., and da Silva, N. C. (2020). Victor: a dataset for brazilian legal documents classification. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1449–1458.

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., and Sui, Z. (2022). A survey for in-context learning. *arXiv preprint arXiv:2301.00234*.

Fernandes, W. P. D., Frajhof, I. Z., Rodrigues, A. M. B., Barbosa, S. D. J., Konder, C. N., Nasser, R. B., de Carvalho, G. R., Lopes, H. C. V., et al. (2022). Extracting value from brazilian court decisions. *Information Systems*, 106:101965.

Fernandes, W. P. D., Silva, L. J. S., Frajhof, I. Z., de Almeida, G. d. F. C. F., Konder, C. N., Nasser, R. B., de Carvalho, G. R., Barbosa, S. D. J., Lopes, H. C. V., et al. (2020). Appellate court modifications extraction for portuguese. *Artificial Intelligence and Law*, 28(3):327–360.

Jones, K. S. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*.

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., and Brown, D. (2019). Text classification algorithms: A survey. *Information*, 10(4):150.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR.

Leitner, E., Rehm, G., and Moreno-Schneider, J. (2019). Fine-grained named entity recognition in legal documents. In *International Conference on Semantic Systems*, pages 272–287. Springer.

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., and Gao, J. (2021). Deep learning–based text classification: a comprehensive review. *ACM Computing Surveys (CSUR)*, 54(3):1–40.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25.

Souza, F., Nogueira, R., and Lotufo, R. (2019). Portuguese named entity recognition using bert-crf. *arXiv preprint arXiv:1909.10649*.

Sulea, O.-M., Zampieri, M., Malmasi, S., Vela, M., Dinu, L. P., and Van Genabith, J. (2017). Exploring the use of text classification in the legal domain. *arXiv preprint arXiv:1710.09306*.

Surden, H. (2018). Artificial intelligence and law: An overview. *Ga. St. UL Rev.*, 35:1305.

Xiao, C., Zhong, H., Guo, Z., Tu, C., Liu, Z., Sun, M., Zhang, T., Han, X., Hu, Z., Wang, H., et al. (2019). Cail2019-scm: A dataset of similar case matching in legal domain. *arXiv preprint arXiv:1911.08962*.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. (2022). React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Zhong, H., Guo, Z., Tu, C., Xiao, C., Liu, Z., and Sun, M. (2018). Legal judgment prediction via topological learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3540–3549.

Zhou, C., Sun, C., Liu, Z., and Lau, F. (2015). A c-lstm neural network for text classification. *arXiv preprint arXiv:1511.08630*.

Zhou, Y., Muresanu, A. I., Han, Z., Paster, K., Pitis, S., Chan, H., and Ba, J. (2022). Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*.