Journal of Information and Data Management, 2024, 15:1, doi: 10.5753/jidm.2024.3368

Legal Document Segmentation and Labeling Through Named Entity Recognition Approaches

Gabriel M. C. Guimarães [**University of Brasilia** | ***gabriel.ciriatico@aluno.unb.br***]
Felipe X. B. da Silva [**University of Brasilia** | ***felipe.barbosa@aluno.unb.br***]
Lucas A. B. Macedo [**University of Brasilia** | ***almeida.bandeira@aluno.unb.br***]
Victor H. F. Lisboa [**University of Brasilia** | ***victor.lisboa@aluno.unb.br***]
Ricardo M. Marcacini [**University of São Paulo** | ***ricardo.marcacini@gmail.com***]
Andrei L. Queiroz [**University of Brasilia** | ***andreiqueiroz@unb.br***]
Vinicius R. P. Borges [**University of Brasilia** | ***viniciusrpb@unb.br***]
Thiago P. Faleiros [**University of Brasilia** | ***thiagodepaulo@unb.br***]
Luís P. F. Garcia [**University of Brasilia** | ***luis.garcia@unb.br***]

Department of Computer Science, University of Brasília, Asa Norte, Brasília, DF, 70910-900, Brazil.

Received: 29 April 2023 • **Published:** 23 February 2024

Abstract
The document segmentation task allows us to divide documents into smaller parts, known as segments, which can then be labelled within different categories. This problem can be divided in two steps: the extraction and the labeling of these segments. We tackle the problem of document segmentation and segment labeling focusing on official gazettes or legal documents. They have a structure that can benefit from token classification approaches, especially Named Entity Recognition (NER), since they are divided into labelled segments. In this study, we use word-based and sentence-based CRF, CNN-CNN-LSTM and CNN-biLSTM-CRF models to bring together text segmentation and token classification. To validate our experiments, we propose a new annotated data set named PersoSEG composed of 127 documents in Portuguese from the Official Gazette of the Federal District, published between 2001 and 2015, with a Krippendorff's alpha agreement coefficient of 0.984. As a result, we observed a better performance for word-based models, especially with the CRF architecture, that achieved an average F1-Score of 75.65% for 12 different categories of segments.

Keywords: Legal documents, Named Entity Recognition, Segmentation

1 Introduction

A Government Gazette is a comprehensive source of information on all official government acts, providing detailed data due to its high volume, publication frequency, and long-standing history. These Official Gazettes maintain periodic publications, such as announcements about the hiring and dismissing of public servants, bidding processes, and contracts between the government and private companies, to promote transparency in government actions. Public officials and professionals often rely on the Official Gazettes to confirm the official status of something and to obtain relevant details such as dates and the relevant agency involved. Such gazettes also enable tracking essential information, including the companies hired by the government, the career progression of civil servants, and more.

Since its foundation in 1960, the Federal District Government publishes its official gazette, the Official Gazette of the Federal District (OGFD) [1]. Editions published during the first seven years are not available on the Internet. The editions published between October 1967 and April 2020 can be downloaded only in PDF format, without any segmentation between the different topics in the document. Since May 2020, OGFDs can be found on-line in text format, divided into segments called acts.

The Official Gazettes present diverse themes and several segments (called acts) in the same document. This document is published in natural language, which creates a challenging scenario to extract the information in a structured way because the language used is typically from the public administration domain. Reading all the publications to extract and classify the necessary information from the various government administration departments requires an inconceivable individual effort to be performed daily. In this case, Natural Language Processing (NLP) tasks, such as text segmentation and automatic text classification, can extract information from entities related to public acts.

Document segmentation is a task that involves dividing a text into smaller, meaningful segments, and a segment labeling task automatically assigns pre-defined labels to segments of a text. Conventionally, these two problems are addressed separately: first, a model is used for document segmentation, and then a text classifier is employed to categorize the segments in acts. In order to structure data from OGFDs, we propose utilizing Machine Learning models for both act extraction and classification. This study investigates the efficacy of tackling both issues simultaneously.

We propose an approach based on token classification, especially Named Entity Recognition (NER), to address the

[1] `https://www.dodf.df.gov.br/`

problems of document segmentation and segment labeling. NER is a technique that involves identifying entities of text composed of words and categorizing them. The two main steps of NER are similar to our problem: extracting and classifying text segments (specifically in NER, words). This paper shows that document segmentation can benefit from NER techniques, although it requires changes since NER usually deals with smaller text segments. We consider a range of models, from classical techniques like Conditional Random Field (CRF) [Lafferty *et al.*, 2001] to more advanced approaches such as Long Short-Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] and Convolutional Neural Network (CNN) [LeCun *et al.*, 1989].

In this context, this work presents two main contributions. Firstly, we offer a data set called PersoSEG of human-annotated segmented documents, more extensive than most existing data sets in this field. Secondly, we evaluate and compare various adapted techniques for NER to extract text segments, contributing to the literature on document segmentation and segment labeling. In addition, it should be highlighted that this paper expands the previously published work [da Silva *et al.*, 2022].

In contrast to [da Silva *et al.*, 2022], this study research presents new experiments for the CNN-CNN-LSTM and CNN-biLSTM-CRF models, both for word-based and sentence-based, which were trained once again due to limitations related to character encoding in the previous work. The evaluation was further extended to include more detailed comparisons of algorithm performance. It also presents an up-to-date discussion of related works on annotated data sets, as well as it details the quality of the proposed data set, including Inter-Annotator Agreement rates such as Krippendorff's alpha.

Additionally, our work follows open science principles, making our methodology, data and materials openly accessible [Vicente-Saez and Martinez-Fuentes, 2018]. These principles allow different researchers to easily reproduce our experiments, expanding on our work or simply using it with educational purposes [Haim *et al.*, 2023]. Our annotated data set is openly available, as well as the notebooks and the Python scripts used to train and evaluate the models. The hyperparameters used to train the models are also detailed in 4.

This paper is structured as follows. Section 2 covers the existing background on text segmentation, classification, and other correlated subjects. Section 3 details the PersoSEG corpus creation and validation. Section 4 describes the methodology used in this research. Section 5 includes a presentation and discussion of the results obtained. Finally, Section 6 concludes this work and presents the final remarks.

2 Related Work

This section provides an overview on the research being made in the document segmentation and NER fields, as well as on the available data sets related to them. Sub-section 2.1 explores the document segmentation problem, formalizing it and detailing the state-of-the art of the field, especially when dealing with legal documents. Sub-section 2.2 explores related work that uses combined approaches to deal with document segmentation and NER. Finally, Sub-section 2.3 lists the existing labeled data sets available to tackle document segmentation and NER, as well as their problems.

2.1 Text Segmentation

Text segmentation, or document segmentation, divides a document into smaller parts, typically text segments [Kumar *et al.*, 2011]. Segments can be split into tokens, and each token can be categorized as a word, phrase, topic, sentence, or any unit of information that represents a subset of the document. Each segment unit has its relevant meaning, which is closely related to the sequence of tokens.

We can define the text segmentation problem formally as follows. Let $D_j = \{S_1, S_2, \ldots, S_n\}$ the set of n text segments of a document d_j, and each segment $S_i \in D_j$ is defined as sequence of tokens, $S_i = t_{1_i}, t_{2_i}, \ldots, t_{m_i}$, where the index of each token in S_i indicate the sequential position in the text document. Here, we define two sets: labels segments $L^S = \left\{l_1^S, l_2^S, \ldots, l_{c^S}^S\right\}$, and tokens labels $L^t = \{l_1^t, l_2^t, \ldots, l_{c^t}^t\}$. We also define a function $\delta : S_i \rightarrow L$ that assign labels to tokens. The segment label l_r^S of a segment S_i is the sequence of labels assigned to the sequence of tokens $t_{k_i}, t_{k+1i}, \ldots, t_{m_i}$, *i.e.*, $l_r^S = \delta(t_{k_i}), \delta(t_{k+1i}), \ldots, \delta(t_{m_i})$. Then, the goal of the text segmentation problem is to find the specific sequence of labels tokens that define a label segment.

Document segmentation models typically use coherence to detect different segments in a text [Barrow *et al.*, 2020]. Term co-occurrences were used in the TextTiling algorithm [Hearst, 1997]. Bayesian methods were used successfully within the BayesSeg algorithm [Eisenstein and Barzilay, 2008], among others [Riedl and Biemann, 2012]. Another popular approach to this problem uses unsupervised algorithms, like GraphSeg [Glavaš *et al.*, 2016], where a graph is built using sentences as nodes and semantic similarity are represented by an edge.

Glavaš and Somasundaran [2020] proposed the Coherence-Aware Text Segmentation (CATS) model, which produces state-of-the-art segmentation performance on a collection of benchmark data sets. It is a multi-task learning model, based on a neural architecture consisting of two hierarchically connected Transformer networks, which couples the sentence-level segmentation goal with the coherence goal that differentiates correct sentence sequences from corrupt ones. The model can successfully segment texts in languages not seen in training. The model has also been proven efficient in zero-shot language transfer experiments.

Aumiller *et al.* [2021] presented a segmentation approach that can predict the topical coherence of sequential text segments spanning multiple paragraphs, effectively segmenting a document and providing a more balanced representation for downstream applications. The approach is based on transformer networks and structural text segmentation, formulated as topical change detection and performing a series of independent classifications that allow efficient tuning on task-specific data.

There is an extensive literature on computer vision and document image analysis, including document segmentation from images [Eskenazi *et al.*, 2017]. Related sub-fields include: document understanding, whose task is to extract and search information from documents through Optical Character Recognition (OCR); document layout analysis (DLA), where regions of interest are identified in a document image; and handwritten text recognition [Coquenet *et al.*, 2023]. However, in this study we do not deal with computer vision: the documents are already given in a textual format. This difference of approach impacts not only in the related work we deal with, but also the data sets we explore, which are composed of documents in textual format, not images containing documents.

2.2 Segmentation as a NER Problem

The NER task is a sequence labeling problem where each word in a sentence must be classified into one of many predefined named entities. It is a token classification problem, where the tokens to be classified are the named entities. Therefore, NER data sets come with word-level annotations, assigning each word its true label. This annotation is usually done with Inside-Outside-Beginning (IOB) tagging, where the O tag indicates a token that does not belong to a chunk, and I-X indicates that the word is within a specific chunk (X might be any classification). This chunk is initialized with a B-X tag and ends with an E-X tag. Since O is not a part of an entity, O tags are not labeled together with an X classification, such as B, I, and E.

Segmentation can be seen as a NER problem when it is considered to have two steps: finding segments in a text and labeling them. Those two steps are the steps required to tackle the NER problem. Successful attempts have been made to combine NER and document segmentation algorithms to improve the segmentation task results and combine document segmentation and segment labeling.

Arnold *et al.* [2019] proposed a model capable of tackling extraction and classification tasks, motivated by how writers make texts: segments are naturally separated, bearing in mind common topics. To translate this idea into architecture, they proposed SECTOR, consisting of sentence encoding, topic embedding, classification, and segmentation. The topic embedding step uses two layers of LSTM to read the documents in both directions, followed by the topic classification, where the class labels are decoded. The topic segmentation step leverages the information in the first steps, outputting the classified segments. SECTOR presented an improvement of 29.5 points F1 when comparing state-of-the-art text classifiers combined with segmentation models [Arnold *et al.*, 2019].

Barrow *et al.* [2020] presented a model capable of learning segmentation boundaries and segmentation level labels together at training time. Segment Pooling LSTM (S-LSTM) is a supervised model based on an LSTM architecture trained to predict segment boundaries and pool over and classify segments. In support of joint training, an approach was developed to teach the model to recover from errors by aligning predicted segments and ground truth [Barrow *et al.*, 2020]. This approach segments the document and annotates its segments without using NER models.

Inan *et al.* [2022] presented a set of models to deal with text segmentation and segment labeling. The authors tackle the task which they call *structured summarization*, where they segment a document and label the segments with generative techniques. They differ from discriminative techniques, where a given set of labels is given in the training step, with the output restricted to these same labels. In generative labeling, the texts receive generated labels that summarize their content. The authors conducted experiments combining discriminative segmentation and generative labeling, as well as both generative segmentation and labeling. The authors improved the performance of state-of-the-art models, achieving F1-Score higher up to 8 points compared to the S-LSTM and SECTOR models [Inan *et al.*, 2022]. The presented models are also focused on mid- to low-computational power, providing models that can be trained without the need for high-resource machines.

2.3 Existing data sets

When dealing with ML architectures, it is important to have high-quality annotated data sets, since the models learn the patterns present within this data. Different data sets have been proposed in the document segmentation field, but there is still a lack of diversity in languages and topics. Document segmentation data sets have been available mainly in English (38% of the works) or in Chinese (33%), regarding mostly web documents, such as web pages, web blogs, social media comments, and reviews. Most of the studies also use words as the segment units (47%), followed by textual topics (18%) [Pak and Teh, 2017].

Koshorek *et al.* [2018] proposed the Wiki-727K, composed of $727,746$ documents from the English Wikipedia, as well as the Wiki-50, a set of 50 randomly sampled documents from the Wiki-727k. This data set can be explored based on topics and it is composed of natural and open-domain text. However, it is still an automatically annotated data set. Aumiller *et al.* [2021] also proposed an automatically annotated data set composed of $74,000$ documents. Their data set, however, is focused on legal documents, being composed of Terms of Service extracted through web crawling. The authors highlighted the lack of legal documents data sets available to train document segmentation models [Aumiller *et al.*, 2021].

Tackling both the document segmentation and the segment labeling problems, Arnold *et al.* [2019] proposed the WikiSection, consisting of $38,000$ documents from English and German Wikipedia. It encompasses up to 30 topics from domains related to diseases and cities, with the first ones having more restricted topics and the last one broader and more ambiguous topics. This data set can be used to train text classification models, document segmentation, or even both of these tasks together, such as was done by Arnold *et al.* [2019].

The existing data sets on document segmentation show the lack of annotated legal documents [Aumiller *et al.*, 2021], as well as the lack of data sets in languages other than English and Chinese Pak and Teh [2017]. There is also a lack of data sets that deal with document segmentation and segment labeling simultaneously, with the most important ones tackling one of these problems, with the exception of WikiSection [Arnold *et al.*, 2019]. To tackle these problems and propose a high-quality annotated data set, one may resort to human labeling, as opposed to the automated labeled data sets seen here. A manually annotated collection of texts with high Inter-Annotator Agreement [Wissler *et al.*, 2014] capable of dealing with these problems would be a huge contribution to the legal document segmentation field.

3 PersoSEG corpus

The OGFDs texts are composed of 3 sections: Section I, where laws, government decrees, and ordinances are published; Section II, where there is information about civil services, such as retirement and allowance; and Section III, where financial information can be found. The corpus used in this paper focuses only on Section II.

The corpus creation was done through 6 steps: the sampling of OGFD editions; the annotation tool selection; the annotators training; the annotation process; the corpus construction through the tool in a structured format; and the corpus validation, calculating the inter-annotator agreement rate. The sampling was done using the DODFMiner tool to download and extract text from PDF files ranging from 2001 to 2019. With the files downloaded and their texts extracted, we selected randomly 100 editions. The selected tool was chosen so as to fulfill some requirements, such as support to the token classification task, collaborative annotation, centralized management and free and open source code, with the *TeamTat* platform being chosen.

These 100 gazettes had the second section extracted, with their subsections, also called acts, being then manually labeled. The acts usually begin with specific words that allow the annotator to identify where it starts. Subsections of the documents that were not part of any act type were removed, such as the beginning and the footer of the documents. The resulting data set comprises the PersoSEG corpus.

The annotation was performed by 21 non-specialist volunteers who were trained based on specialists' guidelines with a set of materials: tutorials in text and video format, as well as official documents detailing each document and entities within it. The process was divided in two phases comprising 4 batches of documents: tagging and validation. In the tagging phase, 90% of the documents were selected and divided into 3 batches, annotated in two steps. In the first step, each annotator A_n received a distinct set of documents D_n; and in the second step, a peer review was made through the shift of the documents, where each annotator would review the next $modn$ set of documents ($D_{n+1 \bmod n}$), where n is the number of annotators. Figure 1 exemplifies the tagging phase, with documents being peer-reviewed.

The validation phase used the last batch, comprising 10% of the remaining documents, where every annotator received the same set of documents D. The validation phase made possible the calculation of the Krippendorff's alpha metric. It was chosen because the data was annotated by multiple annotators and comprised multi-class annotation [Antoine *et al.*, 2014]. The closer the α is to 1, the more the annotators agree with each other's labels; the closer to 0, the more they diverge; and if smaller than 0, then the disagreements are more frequent than they would be by chance. Table 1 shows the Krippendorff's α for personal acts.

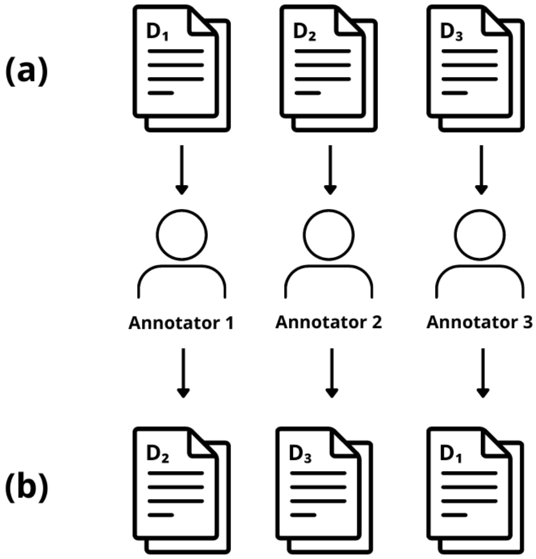

Figure 1. Document distribution done in the tagging phase: (a) each annotator receives a document; (b) the documents are switched and annotators do a peer review.

Krippendorff Score	Support
0.98406	983

Table 1. Krippendorff's α for personal acts.

Table 2 details the final composition of the PersoSEG corpus. A total of $9,058$ acts were annotated, divided into 12 types.

Act type	Number of Acts
Permanence Allowance	134
Cession	265
Dismissal of Commissioned Position	2,009
Dismissal of Effective Position	241
Nomination of Commissioned Position	2,313
Nomination of Effective Position	46
Rectification of Commissioned Appointment	198
Rectification of Effective Appointment	1,214
Reversal	58
Substitution	2,312
Rendered Ineffective Retirement Acts	20
Rendered Ineffective Dismissal or Nomination Acts	248
Total	9,058

Table 2. Total labeled acts available in the data set.

4 Methodology

The experiments performed in this research had the main task of correctly extracting segments from a group of OGFDs. To accomplish this, a new data set was made, called PersoSEG, as explained in 3 resulting in 100 manually annotated documents, publicly available. Word-based and sentence-based

[1] `https://github.com/UnB-KnEDLe/DODFMiner`

[1] `https://github.com/UnB-KnEDLe/persoseg-corpus`

models' capacity for extracting these segments was evaluated. In word-based models, each word from the document has an individual label in IOB format, while in sentence-based models, a single label was assigned to each sentence from the document (also using the IOB format).

We propose a baseline of experiments comprising 6 different models: CRF, CNN-CNN-LSTM and CNN-biLSTM-CRF, each one with a word-based and sentence-based version. This baseline was chosen considering that we are proposing solutions that require less computational power and that use already publicly available models, with changes made to fit into our problem of token classification. Recent models such as the SECTOR [Arnold *et al.*, 2019] are dependent on heavy computation power, since they use transformers, while others, such as the S-LSTM [Barrow *et al.*, 2020] and the models proposed by Inan *et al.* [2022], do not have implementations publicly available.

4.1 Word-based and sentence-based models

The word-based models were CRF, CNN-CNN-LSTM [Shen *et al.*, 2017], and CNN-biLSTM-CRF [Ma and Hovy, 2016]. The CRF model was generated using the *sklearn-crfsuite* library and the *lbfgs* algorithm, with the input features of each word being: (i) the word itself, in lower case; (ii) whether or not the word is a title or in upper case; (iii) amount of digits in the word; and, (iv) all the previous items repeated for nearby words. The CNN-CNN-LSTM used 50 channels in the first CNN to create character-level embeddings, 800 channels in the second CNN to create word-level embeddings, and 200 as the size of the hidden layer of the LSTM. It also used a pre-trained GloVe [Pennington *et al.*, 2014] embedding with 50 dimensions, a fixed learning rate of 10^{-3}, and the *Adam* optimization algorithm. The CNN-biLSTM-CRF had a similar configuration, with the main difference being that it did not create word-level embeddings. Table 3 summarizes the hyperparameters used in the CNN-CNN-LSTM and CNN-biLSTM-CRF models.

For the sentence-based models, the word-based models mentioned in the previous section were adapted to work with sentence labels rather than word labels. For the CNN-CNN-LSTM model, this adaption was done just before the LSTM layer. For the CNN-biLSTM-CRF model, it was done after the biLSTM layer. This embedding for words was then converted to an embedding for the sentence using a single LSTM and taking the final cell state. For the CRF architecture, the adaptation was done by combining the features of only the first four words and the last three words of each sentence, discarding the rest. The features obtained for each of these words were the same as in the word-based version of the model.

The data set used for training was the OGFD corpus, with 5-fold cross-validation. For the deep learning models, 20% of the training set was separated and used as a validation set. Early stopping was also used for both CNN-CNN-LSTM and CNN-biLSTM-CRF, stopping the training iteration after 40 epochs without a reduction of loss in the validation set.

Model	Hyperparameter	Value
CNN-CNN-LSTM	Channels in 1st CNN	50
	Channels in 2nd CNN	800
	Size of hidden layer	200
	Optimizer	Adam
	Learning rate	10^{-3}
	Stopping criteria	Early stopping after 40 epochs without loss reduction
	n-folds	5
CNN-biLSTM-CRF	Channels in CNN	50
	Size of hidden layer	200
	Optimizer	Adam
	Learning rate	10^{-3}
	Stopping criteria	Early stopping after 40 epochs without loss reduction
	n-folds	5

Table 3. Hyperparameters used in the deep learning models.

4.2 Model Evaluation

Following the training process, every word (by the word-based models) or every sentence (by the sentence-based models) was labeled during the model evaluation on the testing set.

The final score of a model was defined as the weighted average of the F1-Score for the types B, I, and E, with the weight being the frequency of each type in the set. The label O was excluded to avoid skewing the final score, as this label is much more frequent than the others, and a high score on this label does not indicate that the model has learned anything.

Additionally, each experiment was conducted using a 5-fold split, alternating the train and test sets. As a result, each experiment has 5 final scores, from which a mean and a standard deviation were calculated. The mean score is the metric used to compare these models, while the standard deviation indicates the impact of the train/test set selection on the results. A Friedman-Nemenyi test was also performed to compare the evaluation metrics of different model architectures.

5 Experimental Results

This section comprises the experiments made using the PersoSEG corpus. Sub-ection 5.1 encompasses the experiments using word-based models, while sub-section 5.2 encompasses the experiments using sentence-based models. CRF, CNN-CNN-LSTM and CNN-biLSTM-CRF models were evaluated in both sections, adapted to deal with words and sentences.

5.1 Word-based models

The F1-Scores obtained by the word-based models for each act type can be seen in Table 4, with the mean and standard deviation of the 5 folds. The CRF model performed

[1] `https://github.com/TeamHG-Memex/sklearn-crfsuite`

better in almost every act type, with the only exceptions being Permanence Allowance, Rectification of Effective Appointment, and Rendered Ineffective Dismissal or Nomination acts. CNN-CNN-LSTM and CNN-biLSTM-CRF performed slightly worse. Acts trained with few data have big standard deviations, sometimes even bigger than the mean. This happened because some acts are more frequent than others, with models trained with more data sharing more similar performance. This explains why Reversal (with 58 acts) and Rendered Ineffective Retirement Acts (with 20 acts) had bigger standard deviation than the mean in the CRF experiments.

The low overall performance for the act types Rendered Ineffective Retirement Acts, Reversal, and Nomination of Effective Position can be partially explained by the fact that they have very few examples in the data set. Act types with the highest amount of documents (more than a thousand) all achieved more than 90% accuracy in all models. These results indicate that having exposure to more examples might be particularly important in this task.

Figure 2 shows the average precision and recall for each model and act type combination. The black dotted lines represent F1-Score isolines: the leftmost one represents an F1-Score of 0.1, the next one represents an F1-Score of 0.2, and so on. The F1-Score value of 1 would be represented by a single dot in the point (1.0, 1.0) of the graph. Dots above the main diagonal indicate acts with more false negatives than false positives, while dots below the main diagonal indicate acts with more false positives than false negatives. Each color represents a model, and each one has 12 points in total in the graph, one for each act type.

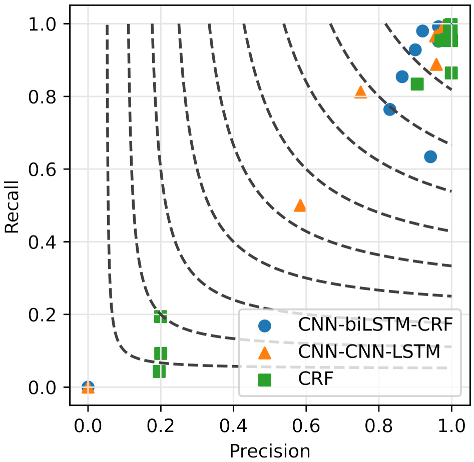

Figure 2. Results obtained by word-based models in terms of their precision and recall.

For the most part, the models had similar values of precision and recall. The largest difference was for the CNN-biLSTM-CRF model with the act type Permanence Allowance, which had a precision score of 0.942 and a recall score of 0.634. All significant differences (above 0.1) in precision and recall were caused by models that had higher precision than recall, indicating a comparatively high number of false positives.

5.2 Sentence-based models

Table 5 shows the F1-Scores obtained by the sentence-based version of the models. Both CNN-CNN-LSTM and CNN-biLSTM-CRF models performed worse than their word-based counterparts overall. It might be that the algorithm used for segmentation increased the task's difficulty rather than lowering it due to separating segments that were important for proper classification. The CRF model, however, still achieved high scores and had an average F1-Score significantly higher than any of the word-based models. Once again, it is likely that the CRF models had an advantage due to their access to capitalization features.

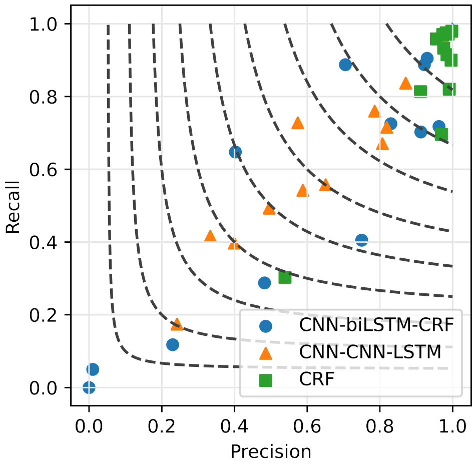

Figure 3. Results obtained by sentence-based models in terms of their precision and recall.

Figure 3 shows the average precision and recall for each model and act type combination. The most significant difference was for the CNN-biLSTM-CRF model with the Rendered Ineffective Dismissal or Nomination act type, which achieved a precision of 0.750 and a recall of 0.405, indicating a large number of false negatives. The differences in precision and recall were overall higher for the sentence-based models than for the word-based models.

The predictive performances of the word-based and sentence-based models were compared to evaluate the statistical significance of the experimental results using the Friedman and Nemenyi post-hoc statistical tests with a confidence level of 95%.

Figure 4 shows the relative performance of different word-based architectures according to the Nemenyi test. The horizontal axis in the image indicates the rank of each architecture. A lower rank represents better performance. In the image, CD is the Critical Difference. A model is considered to have performed better than another only if it has a lower ranking and the rank difference from the other model is higher than CD.

Act type\Model	CNN-CNN-LSTM (%)	CNN-biLSTM-CRF (%)	CRF (%)
Permanence Allowance	57.84 ± 47.23	**79.27 ± 7.21**	19.67 ± 39.35
Cession	96.76 ± 2.07	96.17 ± 1.07	**99.10 ± 0.68**
Dismissal of Commissioned Position	98.46 ± 0.43	97.28 ± 1.10	**99.50 ± 0.10**
Dismissal of Effective Position	91.91 ± 3.44	84.85 ± 8.92	**97.58 ± 1.40**
Nomination of Commissioned Position	99.25 ± 0.49	97.29 ± 1.43	**99.72 ± 0.11**
Nomination of Effective Position	0.00 ± 0.00	0.00 ± 0.00	**92.26 ± 7.05**
Rectification of Commissioned Appointment	76.38 ± 8.92	78.11 ± 4.73	**86.38 ± 2.92**
Rectification of Effective Appointment	95.89 ± 1.04	90.49 ± 4.43	**96.19 ± 0.58**
Reversal	0.00 ± 0.00	0.00 ± 0.00	**7.01 ± 14.03**
Substitution	98.97 ± 0.62	97.83 ± 1.22	**99.78 ± 0.26**
Rendered Ineffective Retirement Acts	0.00 ± 0.00	0.00 ± 0.00	**12.66 ± 25.31**
Rendered Ineffective Dismissal or Nomination Acts	**98.83 ± 0.59**	92.94 ± 1.65	97.90 ± 1.30
Average	67.86 ± 5.40	67.85 ± 2.65	**75.65 ± 7.76**

Table 4. Results of word-based models.

Act type\Model	CNN-CNN-LSTM (%)	CNN-biLSTM-CRF (%)	CRF (%)
Permanence Allowance	71.51 ± 17.18	66.86 ± 20.69	**94.77 ± 1.28**
Cession	96.31 ± 1.94	86.15 ± 6.39	**97.92 ± 0.47**
Dismissal of Commissioned Position	39.73 ± 48.65	14.37 ± 24.89	**96.69 ± 1.12**
Dismissal of Effective Position	42.10 ± 42.05	0.0 ± 0.0	**94.51 ± 1.36**
Nomination of Commissioned Position	43.44 ± 32.59	74.90 ± 17.61	**97.68 ± 0.75**
Nomination of Effective Position	55.40 ± 21.88	37.53 ± 16.47	**89.46 ± 4.22**
Rectification of Commissioned Appointment	39.24 ± 32.44	26.57 ± 31.53	**84.65 ± 1.93**
Rectification of Effective Appointment	84.63 ± 8.10	81.59 ± 10.49	**95.26 ± 0.21**
Reversal	52.63 ± 26.91	30.63 ± 35.27	**80.31 ± 4.42**
Substitution	77.15 ± 38.62	53.71 ± 38.97	**98.67 ± 0.22**
Rendered Ineffective Retirement Acts	19.88 ± 26.46	0.0 ± 0.0	**36.02 ± 13.68**
Rendered Ineffective Dismissal or Nomination Acts	56.17 ± 45.96	21.14 ± 12.39	**95.66 ± 1.44**
Average	56.52 ± 28.56	41.12 ± 17.89	**88.47 ± 2.59**

Table 5. Results of sentence-based models.

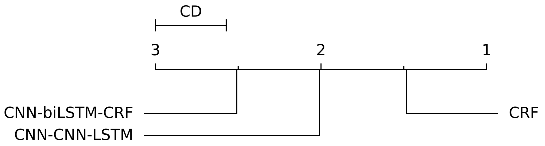

Figure 4. Relative performance of word-based models according to the Nemenyi test.

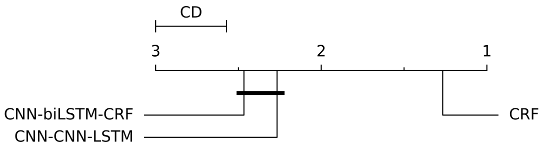

Figure 5. Relative performance of sentence-based models according to the Nemenyi test.

Figure 5 shows the relative performance of different sentence-based architectures according to the Nemenyi test. The CRF architecture had the best score, followed by CNN-CNN-LSTM and then CNN-biLSTM-CRF for the sentence-based models. Since the rank difference between the CNN-biLSTM-CRF model and the CNN-CNN-LSTM was smaller than the critical difference (CD), their performance cannot be considered significantly different according to this test.

6 Conclusions

The document segmentation task can still benefit from different techniques and approaches. This paper reinforced the usefulness of tackling document segmentation and segment labeling problems in the same architecture. We showed that NER architectures can be used in this context, combining these two traditional NLP fields.

We explored these problems by focusing on legal documents in Portuguese. Due to the lack of document segmentation data sets related to legal documents or data sets in Portuguese, we presented an annotated corpus. This corpus comprises 127 documents annotated by trained non-specialists, composed of 9,058 segments, called acts, separated into 12 different categories. With this, we contributed to the research on the segmentation of legal documents and publicly available data sets.

Three models were compared to address document segmentation and segment labeling: a traditional CRF architecture, CNN-CNN-LSTM, and CNN-biLSTM-CRF. While these models were made for taking words as inputs, they were adapted to sentence-based versions and compared to their word-based counterparts.

For the word-based models, the CRF model performed significantly better than the remaining two, achieving an average F1-Score of 75.65. Although the CNN-CNN-LSTM and CNN-biLSTM-CRF had similar averages (67.86 and 67.85, respectively), the Nemenyi test indicates that the CNN-CNN-LSTM performed better. For the sentence-based models, the CNN-CNN-LSTM architecture achieved an average of 56.52, better than the CNN-biLSTM-CRF model, which had 41.12. However, their difference was not statistically significant according to the Friedman and the Nemenyi post-hoc tests. Nonetheless, it was the sentence-based CRF that achieved the best score among all models, with an average F1-Score of 88.5. Access to capitalization features may have given the CRF architecture an advantage over the other models.

One of the limitations of this work is the scarcity of labeled data for some categories. The total number of documents used was less than a thousand, and there were variations in the topics of the segments, with specific categories having more annotations than others. This limitation is due to the expensive cost of human annotation.

There is room for improvement in experiments conducted using a baseline of architectures and commonly used data sets. This would benefit the comparison of our proposed architectures with others already established in the literature. Additionally, we could enhance the utilized architectures by leveraging different pre-trained embeddings.

In the future, it would be worthwhile to expand the proposed data set to enhance each act type's balance. Additionally, incorporating more than one type of word embedding could lead to significant improvements. Furthermore, expanding our baseline to include state-of-the-art architectures and data sets, such as those mentioned in sub-section 2.2, would be pertinent to future work.

Finally, it is worth mentioning that our work can be easily expanded due to the open science principles we followed, making all the data and materials openly available. Especially, the annotated data set is a valuable contribution to the field and future works that combine document segmentation and classification.

Funding

This research was funded by FAPDF (grant 07/2019).

Authors' Contributions

Gabriel M. C. Guimarães and Felipe X. B. da Silva are the main writers of this manuscript and performed the experiments and related works research. Lucas A. B. Macedo and Victor H. F. Lisboa contributed to the writing, especially in the data sets sections (both in related works and the methodology). Ricardo M. Marcacini, Andrei L. Queiroz, Vinicius R. P. Borges, and Thiago P. Faleiros reviewed the manuscript. Luís P. F. Garcia supervised the writing and the experiments.

Competing interests

The authors declare that they have financial competing interests, as the research was funded by FAPDF.

Availability of data and materials

The code for the experiments and the generated data sets can be accessed on `https://github.com/UnB-KnEDLe/kdmile-2022-segmentation`.

Acknowledgment

The authors would like to thank Fundação de Apoio à Pesquisa do Distrito Federal (FAPDF), project KnEDLe (grant 07/2019).

References

Antoine, J.-Y., Villaneau, J., and Lefeuvre, A. (2014). Weighted krippendorff's alpha is a more reliable metrics for multi-coders ordinal annotations: experimental studies on emotion, opinion and coreference annotation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 550–559. DOI: 10.3115/v1/E14-1058.

Arnold, S., Schneider, R., Cudré-Mauroux, P., Gers, F. A., and Löser, A. (2019). SECTOR: A neural model for coherent topic segmentation and classification. *Transactions of the Association for Computational Linguistics*, 7.

Aumiller, D., Almasian, S., Lackner, S., and Gertz, M. (2021). Structural text segmentation of legal documents. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, page 2–11. DOI: 10.1145/3462757.3466085.

Barrow, J., Jain, R., Morariu, V., Manjunatha, V., Oard, D., and Resnik, P. (2020). A joint model for document segmentation and segment labeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 313–322. DOI: 10.18653/v1/2020.acl-main.29.

Coquenet, D., Chatelain, C., and Paquet, T. (2023). Dan: a segmentation-free document attention network for handwritten document recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):8227–8243. DOI: 10.1109/TPAMI.2023.3235826.

da Silva, F., Guimarães, G., Marcacini, R., Queiroz, A., Borges, V. R. P., Faleiros, T., and Garcia, L. (2022). Named entity recognition approaches applied to legal document segmentation. In *Anais do X Symposium on Knowledge Discovery, Mining and Learning*, pages 210–217.

Eisenstein, J. and Barzilay, R. (2008). Bayesian unsupervised topic segmentation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 334–343.

Eskenazi, S., Gomez-Krämer, P., and Ogier, J.-M. (2017). A comprehensive survey of mostly textual document segmentation algorithms since 2008. *Pattern Recognition*, 64:1–14.

Glavaš, G., Nanni, F., and Ponzetto, S. P. (2016). Unsupervised text segmentation using semantic relatedness graphs. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 125–130. DOI: 10.18653/v1/S16-2016.

Glavaš, G. and Somasundaran, S. (2020). Two-level transformer and auxiliary coherence modeling for improved text segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7797–7804. DOI: 10.1609/aaai.v34i05.6284.

Haim, A., Shaw, S., and Heffernan, N. (2023). How to open science: A principle and reproducibility review of the learning analytics and knowledge conference. In *LAK23: 13th International Learning Analytics and Knowledge Conference*, pages 156–164.

Hearst, M. A. (1997). Text tiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9:1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

Inan, H., Rungta, R., and Mehdad, Y. (2022). Structured summarization: Unified text segmentation and segment labeling as a generation task. *CoRR*, 2209.13759. DOI: 10.48550/arXiv.2209.13759.

Koshorek, O., Cohen, A., Mor, N., Rotman, M., and Berant, J. (2018). Text segmentation as a supervised learning task. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 469–473. DOI: 10.18653/v1/N18-2075.

Kumar, M., Sharma, R. K., and Jindal, M. K. (2011). Segmentation of lines and words in handwritten gurmukhi script documents. In *Proceedings of the First International Conference on Intelligent Interactive Technologies and Multimedia*, page 25–28. DOI: 10.1145/1963564.1963568.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, page 282–289.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551.

Ma, X. and Hovy, E. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. *CoRR*, 1603.01354. DOI: 10.48550/arXiv.1603.01354.

Pak, I. and Teh, P. L. (2017). Text segmentation techniques: A critical review. In *Innovative Computing, Optimization and Its Applications*, pages 167–181.

Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. DOI: 10.3115/v1/D14-1162.

Riedl, M. and Biemann, C. (2012). TopicTiling: A text segmentation algorithm based on LDA. In *Proceedings of ACL 2012 Student Research Workshop*, pages 37–42.

Shen, Y., Yun, H., Lipton, Z., Kronrod, Y., and Anandkumar, A. (2017). Deep active learning for named entity recognition. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 252–256. DOI: 10.18653/v1/W17-2630.

Vicente-Saez, R. and Martinez-Fuentes, C. (2018). Open science now: A systematic literature review for an integrated definition. *Journal of business research*, 88:428–436.

Wissler, L., Almashraee, M., Díaz, D. M., and Paschke, A. (2014). The gold standard in corpus annotation. In *IEEE Germany Student Conference*, pages 1–4.