Contributions to Legal Document Summarization: Judgments from the Portuguese Supreme Court of Justice

Margarida Rebelo Dias
Iscte - Instituto Universitário de Lisboa, Lisbon, Portugal
INESC-ID Lisboa, Lisbon, Portugal

Ricardo Ribeiro
Iscte - Instituto Universitário de Lisboa, Lisbon, Portugal
INESC-ID Lisboa, Lisbon, Portugal

H. Sofia Pinto
Instituto Superior Técnico - Universidade de Lisboa, Lisbon, Portugal
INESC-ID Lisboa, Lisbon, Portugal

Abstract

Legal documents are commonly known for being lengthy and having a specific vocabulary. For professionals and non-jurists, having a summary of each document is crucial so they can use it as a reference for other cases without spending too much time reading the entire document. In the Portuguese Supreme Court of Justice, summaries are done manually, by its Judges which is very time-consuming because of the length of the legal documents. Aiming to support the Judges in this task, the goal of this work is to investigate how different techniques and methods of automated text summarization can achieve good performance on Portuguese legal documents.

2012 ACM Subject Classification Computing methodologies → Natural language processing; Applied computing → Law

Keywords and phrases automatic text summarization, legal document summarization, abstractive summarization, transformers, European Portuguese

Digital Object Identifier 10.4230/OASIcs.SLATE.2024.2

Funding This research was supported by Fundação para a Ciência e Tecnologia (FCT), through the INESC-ID multi-annual funding with reference DOI:10.54499/UIDB/50021/2020. This research is part of the IRIS project with reference PR07005.

Acknowledgements This project is a collaboration involving the Portuguese Supreme Court of Justice and INESC-ID.

1 Introduction

In the legal domain, the provision of concise summaries for legal documents is essential. Given the length and complexity of these documents, a summary will help reduce the time of finding and understanding specific legal documents in an easier way. Today, summaries are still done manually by professionals, which is a very time-consuming process, making it important to automate the summarization task [8].

When generating a summary, the main goal is to generate a shorter text that captures the context of the original text while maintaining its fluency and coherence. In the summarization task, concerning the relation to the source document, there are two main approaches: extractive summarization, which extracts the top ranked sentences of the document to be summarized, and abstractive summarization, which generates the summary by rewriting the document [1].

In the legal field, extractive summarization has already shown good performance in a variety of works [6, 15]. However, when applying an extractive approach, the generated summaries may lack grammatical correctness and fluency, due to their reliance on sentence extraction. As a result, the abstractive approach is becoming very appealing, considering that it simulates the way a human would write a summary, generating a more natural and structured version. In recent years, research in text summarization has shifted towards exploring abstractive models [9, 13] and the wide adoption of transformer-based generative pre-trained models is having a significant impact in the field.

A major challenge of legal document summarization is that the law differs from country to country, which makes it more difficult to use an already trained model from a specific legal system to summarize documents from another legal system.

In this study, we implemented different approaches to understand the effectiveness of text summarization of Portuguese legal documents from the Portuguese Supreme Court of Justice, exploring approaches that range from a simple extractive summarization approach, LexRank [3], that served as our baseline due to its simplicity and being a well-known algorithm for sentence extraction, to a hybrid approach combining LexRank and MBART [12], that allowed us to evaluate the performance of generating a summary using a transformer model that can better capture the context of a document. We also analyze the difference between the generation of summaries based on sentence-level and summary-level strategies. To evaluate the generated summaries, we use ROUGE and BERTscore metrics.

After this introduction, the structure of the paper is as follows: Section 2 provides a literature review of prior related works. Section 3 offers an overview of the dataset. Section 4 provides a brief description of methods employed and how they were implemented in our work. The results are discussed in Section 5. Finally, Section 6 refers to the conclusions and possible directions for future work.

2 Related Work

In this section we focus in some of the methods employed for legal document summarization and present the difference between generating summaries using sentence-level and summary-level approaches.

2.1 Legal Document Summarization

Extractive methods are the most common approach to text summarization. Some of them include scoring systems using graphs like LexRank, which can extract the most important sentences from the document based on their similarity, representing sentences with Term Frequency-Inverse Document Frequency (TF-IDF) vectors [3]. Other methods employ deep learning models, such as transforms like BERT [2], which can understand the context of the words in a text, better capturing its semantics [11]. One notable advantage of these model is their versatility across various languages. BERT architecture was used to create multilingual models, such as “BERT-base-multilingual-cased”, which can be fine-tuned to facilitate extractive summarization in specific languages. An example of this is BERTimbau, an adaptation of BERT designed specifically for the Brazilian Portuguese language [14].

It can be intuitively assumed that there is a preference for reading a text that is fluent and coherent, as opposed to some extracted sentences without any connectors. Consequently, abstractive summarization approaches have been taken into consideration more recently. There has been work done in this field, such as BART, which is a model created to effectively

capture the contextual information of a text in a bidirectional way, again based on the transformer architecture, which is a sequence-to-sequence model that has shown effective results in generating text [9].

Recent advances in large language models, such as GPT-3 and GPT-4, have demonstrated superior ability in understanding and generating coherent texts. Their capacity to capture complex relationships between sentences and texts renders GPT models relevant to the task of summarization. These models are trained on a diverse range of datasets comprising multiple languages, which offers an advantage when dealing with non-English texts, such as Portuguese ones (see, for example, [17]).

In the case of sensitive documents that contain personal information, like Portuguese legal documents, it is essential to consider how summarization models manage the data provided as input. For instance, models such as GPT retain all the information provided, which represents a risk if a document that contains sensitive data is provided to these models. In the case of Portuguese Supreme Court of Justice judgments, it is necessary to create summaries based on judgments that have not been anonymized, as the judge may require the use of sensitive data to create a well-founded and accurate summary. While GPT models have achieved state-of-the-art results in the natural language processing field, it is essential to consider the risk of violating data privacy. Other factor to take into consideration when implementing transformer-based models, especially in the context of legal documents, is the number of tokens a model can process. When dealing with lengthy documents, like legal documents, this limitation is of significant concern in a way that challenges the model ability to capture the full context of the document. Hybrid summarization approaches can offer a solution to this problem by first extracting some sentences to create a smaller text that can actually be an input for an abstractive algorithm [7, 5].

2.2 Sentence-level *vs* Summary-level

The majority of extractive summarization models use sentence-level methods where they extract sentences based on certain criteria to generate the final summary. However, these algorithms tend to select highly generalized sentences. In contrast to them, approaches based on summary-level methods have demonstrated great results and improved the quality of extractive models. In this type of methods, the top sentences of a document are extracted and combined to generate a range of different candidate summaries. Subsequently, by using a text-matching model the best summary is chosen. For example, MatchSum is a model where it is calculated the similarity between all generated candidates and the original text. The candidate with the highest score is selected for the final summary [18]. A more recent improvement, SeburSum, introduces a contrastive learning framework to train the model. Also instead of using the original text they just compare the candidates between them instead, which improved the computational performance of the model [4].

3 Dataset

Our data is composed from 5000 legal documents from the Portuguese Supreme Court of Justice, representing the "Cível", "Criminal", and "Social" areas. For each document, there is a respective summary, which was used as a reference summary for evaluation purposes. It is also important to refer to the structure of the legal documents, which are typically organised into three, top level, distinct sections:

- Report (*relatório* in Portuguese), which outlines the main topics of the judgment, identifies the different parties involved, and specifies the decision to be made;

- Grounds (*fundamentação* in Portuguese), where a first subsection describes the facts of the case (*matéria de facto* in Portuguese), and a second part that integrates all the details that have been addressed so far that could be taken into consideration to make the final decision (*fundamentação de direito* in Portuguese);
- Decision (*decisão* in Portuguese) that corresponds to the final decision of the judges.

For the preparation of the data we start by cleaning all the documents and summaries, removing HTML tags by using "BeautifulSoup".[1] In addition, all the documents were divided into sentences. Secondly, we create two different datasets. One with the original documents and the respective summaries, containing all the 5000 entries which we called "Dataset 1" – we can see the distribution of documents per area in Table 1. And a second dataset, where instead of using the original documents, we only use the parts of the documents extracted from the sections *fundamentação de direito* and *relatório*, which we call "Dataset 2" (see Table 2). This second dataset was created with the aim to investigate whether certain sections of Portuguese legal documents were more relevant to summarization than others. Given that the original documents do not always contain a uniform structure, it was only possible to identify 3552 documents that contain the sections *fundamentação de direito* and *relatório*. After analyzing both datasets we could observe the following:

- The documents from the "Criminal" area are more lengthy than the documents from the other two areas having an average number of sentences per document of 209.4, around more 68 and 34 sentences than the "Cível" and "Social" areas, respectively (see Figure 1). Subsequently, we can see that the number of sentences in the summaries in the "Criminal" area is also bigger than the others (see Figure 2);
- It is also clear the variation of the number of sentences for the documents and summaries from "Dataset 1", where the number of outliers is very salient on both graphs (see Figures 1 and 2);
- The mean and median number of sentences for the summary range between five to seven in "Dataset 1" (see Figure 2) and between four to six sentences for "Dataset 2" (see Figure 4);
- In "Dataset 2" the number of sentences that composes the section *relatório* is more consistent, with a mean of 73.2 sentences per text, than for the section *fundamentação de direito* where there is a lot of texts with more than 100 sentences (see Figure 3).

Table 1 Total number of documents for each area.

Dataset 1	Number of Documents
Cível	2862
Criminal	1444
Social	694

4 Experiments

The goal of this study is to investigate the performance of different models in summarizing Portuguese legal documents. This section outlines the experimental setup and process implemented to evaluate the selected algorithms. As previously described in Section 3,

[1] `https://pypi.org/project/beautifulsoup4/`

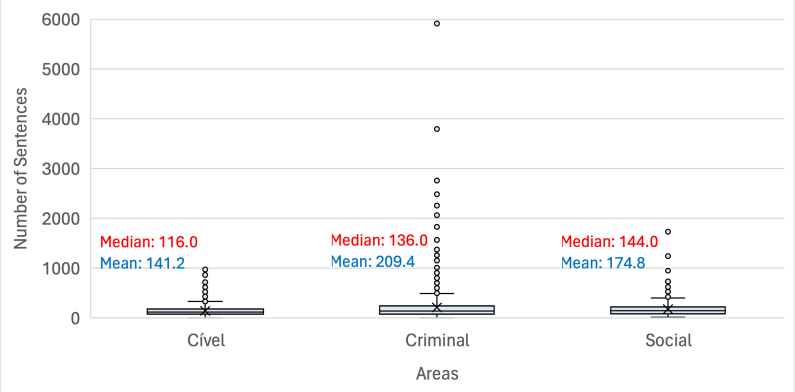


Figure 1 Number of sentences in legal documents – Dataset 1.

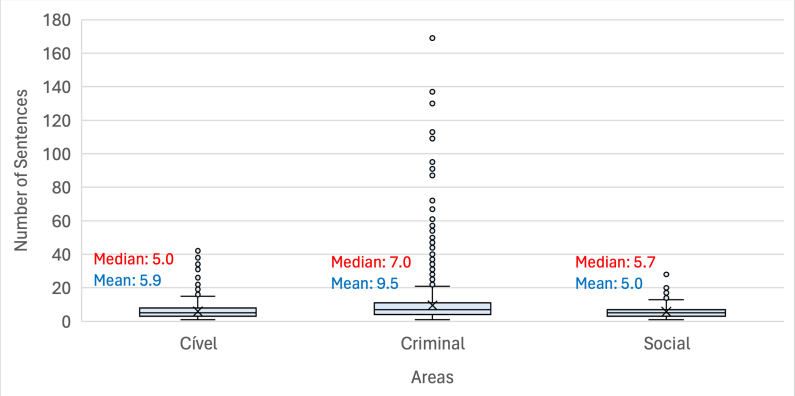


Figure 2 Number of sentences in reference summaries – Dataset 1.

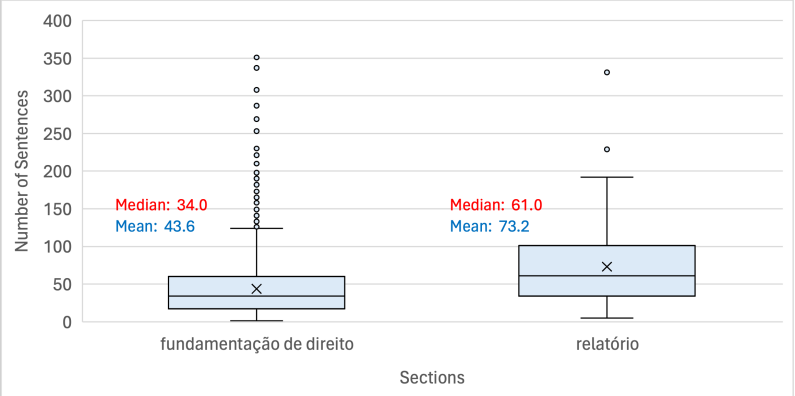


Figure 3 Number of sentences in sections from the legal documents – Dataset 2.

Table 2 Total number of documents for each section.

Dataset 2	Number of Documents
fundamentção de direito	3393
relatório	159

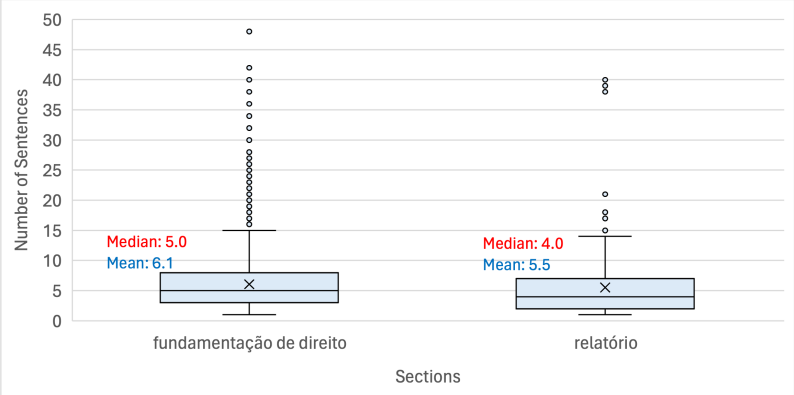


Figure 4 Number of sentences reference summaries – Dataset 2.

we start by preparing the data for input into the summarization models. Secondly, three distinct models were selected, including both extractive and abstractive methods, in order to evaluate the different approaches. The details and justification for each model are provided in Subsection 4.1. The next phase focused on setting up the models' parameters, such as the maximum length of the summaries, and adapting each model to our necessities. Finally, the outcomes of each model were analyzed and compared. To evaluate the generated summaries, we use two metrics: ROUGE and BERTScore, describe in Subsection 4.2.

4.1 Approaches

For our experiments, we used different models and algorithms that showed good performance on summarization tasks. In the following, we describe the overall characteristics of each of them.

LexRank The LexRank algorithm is an unsupervised, graph-based method for summary generation. LexRank has been selected for this study due to is effectiveness in identifying relevant sentences within a text by evaluating the relation between sentences. In this method, sentences are represented as nodes and linked based on their similarity, calculated using the cosine similarity. Each sentence is assigned a score, representing its relevance to the document. The final summary is created by selecting the top N sentences with the highest scores [3]. The simplicity of LexRank makes it a solid algorithm to serve as our baseline model. Consequently, this method allows for the comparison of results from more complex implementations, and provides a starting point for the improvement of the other implementations where LexRank is used to rank sentences by relevance.

SeburSum SeburSum is a summary ranking strategy also used for extractive summarization. Similar to LexRank, SeburSum extracts the top N sentences and generates combinations of the sentences to create different candidate summaries. It compares the similarity between all the candidates and selects the most representative one [4]. Given that SeburSum demonstrated improvements in summary-level strategies relative to sentence-level strategies, we examined how it performed with documents from the legal domain.

The main goal was to ascertain if the generated summaries improved when compared with the summaries generated with LexRank. This was achieved by comparing the results obtained in both implementations. Another reason for selecting SeburSum was to determine if the top sentences selected by a sentence-level approach, such as the LexRank algorithm, were indeed the most suitable ones to be included in the final summary. To this end, we analyzed whether the selected sentences for both generated summaries differed.

MBART The MBART model is a transformer-based architecture capable of handling multilingual texts: it learns to comprehend and generate text in different languages. When given a text as input in a specific language and the target language, the model has shown good performance in generating accurately translated text [12]. Although commonly employed for translation purposes, we use MBART for summarization due to its pre-training with Portuguese data and its capacity to generate fluent and coherent texts by understanding the context and relationship between the sentences in a document. Furthermore, it should be noted that MBART is not an online abstractive model that could potentially compromise sensitive data from legal documents.

4.2 Evaluation Metrics

One of the most important processes in summarization is the way we evaluate the degree of similarity between two summaries. This technique can be employed to evaluate how well a model performs by comparing generated and reference summaries. Bellow we describe two well-known methods that can provide scores expressing the degree of similarity between two summaries:

ROUGE is one of the most used metrics for evaluating text summarization. ROUGE measures the overlap of N-grams between the generated summary and the reference one, where N indicates the contiguous sequences of $gramN$ words [10] (see Eq. 1).

$$\text{ROUGE} = \frac{\sum_{S \in Reference\ Summaries} \sum_{gramN \in S} count\_match(gramN)}{\sum_{S \in Reference\ Summaries} \sum_{gramN \in S} count(gramN)} \tag{1}$$

ROUGE-L and ROUGE-Lsum are other metrics from ROUGE family that do not require a predefined N-gram length. ROUGE-L refers to the longest common sub-sequence between the texts. ROUGE-Lsum refers to the longest common sub-sequence between sentences from two texts.

BERTscore differs from other metrics because it uses the contextual embeddings of tokens to compute the cosine similarity, capturing the semantic context of words in sentences and providing a more accurate evaluation of text generation tasks. It takes into account recall, that measures the proportion of the reference summary that is covered by the generated summary; precision which evaluates how well the generated summary represents the content of the reference summary; and F1 score which combines recall and precision, as shown in the following equations (Eqs. 2, 3, and 4), where x is a reference sentence, $\hat{x}$ is a candidate sentence, and $x_i^T \hat{x}_j$ is a cosine similarity calculation [16].

$$R_{\text{BERT}} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x_j} \in \hat{x}} x_i^T \hat{x}_j \tag{2}$$

$$P_{\text{BERT}} = \frac{1}{|\hat{x}|} \sum_{\hat{x_j} \in \hat{x}} \max_{x_i \in x} x_i^T \hat{x}_j \tag{3}$$

$$F1_{\text{BERT}} = \frac{2 \cdot P_{\text{BERT}} \cdot R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}} \tag{4}$$

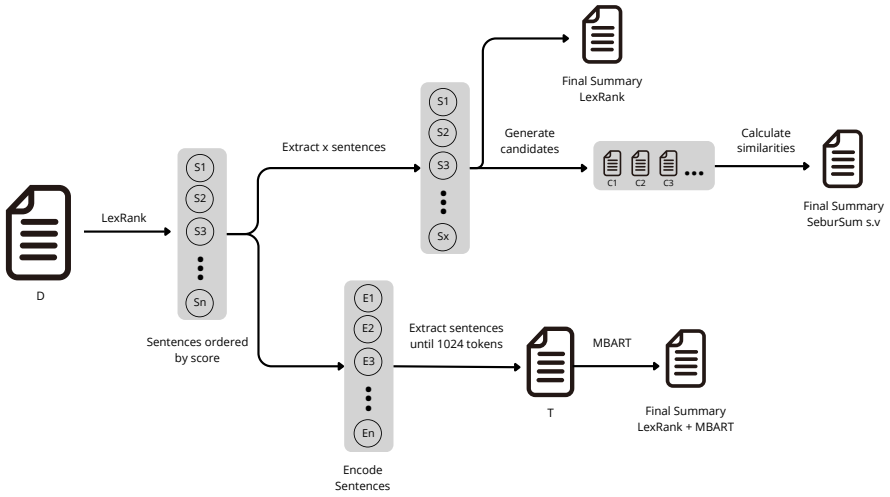


Figure 5 Architecture design for Portuguese legal documents summary generation using LexRank, SeburSum simplified version, and LexRank+MBART.

4.3 Implementations

In this study, three distinct approaches were implemented, which we will detail in this section.

Figure 5 illustrates the overall architecture designed, showing three different branches, each representing a different implementation.

- We start by employing the LexRank algorithm for our extractive method. The goal was to understand the impact of different summary sizes in comparison with the reference summaries. We use the classical LexRank algorithm, where the number of sentences to generate the summaries was set at five, six, and seven and the threshold parameter to 0.1. The selected values were chosen on the basis of the mean of the reference summaries, as seen before (see Figures 2 and 4).
- Additionally, we aimed to investigate how an abstractive summarization approach performs in this domain. For that, we selected the MBART model because it can handle Portuguese texts. As we employ this model for summarization rather than translation, both the original text and target languages were set to Portuguese. Also, it was necessary to add the "*max_length*" parameter to ensure that the generated text was smaller than the input text.

  Another issue we needed to address was the 1024 tokens input limitation that the model can handle. One possible solution for reducing the texts, would be to truncate the input texts to the token limit. However, this would result in the first sentences of the texts being the only ones to be processed by MBART model. Consequently, it can not be guaranteed that the first sentences were the most relevant ones to be present in the final summary. To address this issue, we thought of a hybrid approach that combines the LexRank scoring system, to order the sentences from input text by relevance, and the MBART model for generating the text. Our implementation follows this structure: given a document D that contains n sentences, $D = (S_1, S_2, S_3, \ldots, S_n)$, we first use the LexRank algorithm to rank the sentences by relevance; secondly, we create the input text

T by selecting the sentences that obtain the highest LexRank scores until T reaches the limit of 1024 tokens. In order to determine the number of tokens present in each sentence, we encoded each sentence with "MBART50TokenizerFast";[2] finally, we rearrange the sentence order according to the original position in D and pass T into the MBART model to generate the final summary.
By applying this strategy, the objective was to provide the highest amount of content to the MBART model, as well as the most informative.

- To verify the efficacy of a summary-level approach, we start by implementing a simplified version of SeburSum without using the contrastive learning framework. In our version of SeburSum, we start by ordering the sentences from the original document D by relevance, applying the LexRank algorithm, where $D = (S_1, S_2, S_3, \ldots, S_n)$. The next step involved creating a set of candidate summaries, $C = (C_1, C_2, C_3, \ldots, C_m)$. To create the candidates, it was necessary to select the top N sentences and the minimum, min, and maximum, max, number of sentences a candidate can have. The candidate summaries are then created by generating all n-sentence combinations for all possible candidate sizes. For each candidate, we create the embedding for the full text using the pretrained model "neuralmind/bert-base-portuguese-cased"[3] and calculate the cosine similarity between all candidates that did not contain equal sentences in them. Finally, the candidate summary that achieved the highest score is chosen as the final summary.
  For each dataset we use the following parameters:
  - $k = 12$, min $= 4$, max $= 6$ for "Dataset 1";
  - $k = 10$, min $= 3$, max $= 5$ for "Dataset 2".

5 Results and Discussion

For all implementations, we evaluate the generated summaries comparing them to the reference summaries by calculating the ROUGE and BERTscore scores. From the ROUGE metrics we show results specifically for: ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-Lsum. And for BERTscore we show recall, precision, and F1-score.

Table 3 shows the results of the experience with the LexRank algorithm for both datasets. When analyzing the results from LexRank we can observe that ROUGE only achieves the best results of 0.359 for "Dataset 1" and 0.401 for "Dataset 2" in the ROUGE-1 metric. This can indicate that the generated summaries do not contain the exact same words (or sequence of words) as the reference summaries, and as a consequence ROUGE-L and ROUGE-Lsum could not have achieved higher results. For BERTscore it is visible a balanced range of scores between 0.676 to 0.710 for "Dataset 1" and 0.695 to 0.733 for "Dataset 2".

We can observe the limitations of the LexRank algorithm in selecting the exact top sentences from the document, due to the ROUGE scores being very low, suggesting the algorithm could be missing some important key words. However, when analyzing the BERTscore we can say that the algorithm can extract sentences that are semantically similar to the ones in the reference summaries, demonstrating that the algorithm can capture the essence for the original documents (note, however, that this behaviour of BERTscore is well-known and might also have an impact on the achieved results). When analyzing the difference between the number of extracted sentences, we verify that it does not have a

[2] https://huggingface.co/docs/transformers/model_doc/mbart
[3] https://huggingface.co/neuralmind/bert-base-portuguese-cased

significant impact on the performance since the results do not show a significant difference. Also, we can see a sightly better performance on "Dataset 2", suggesting that the algorithm my perform better when using a smaller section from the original document.

Table 3 Results from LexRank implementation.

Extracted Sentences	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE (Lsum)	R_{BERT}	P_{BERT}	F_{BERT}
5 (Dataset 1)	**0.359**	0.146	**0.195**	**0.275**	0.676	0.703	0.689
6 (Dataset 1)	0.354	**0.148**	0.192	0.274	0.676	0.708	**0.691**
7 (Dataset 1)	0.348	**0.148**	0.189	0.270	0.676	**0.710**	**0.691**
5 (Dataset 2)	**0.401**	0.207	**0.232**	**0.348**	**0.697**	0.725	0.710
6 (Dataset 2)	0.394	**0.209**	0.227	0.345	0.696	0.729	0.711
7 (Dataset 2)	0.386	**0.209**	0.221	0.340	0.695	**0.733**	**0.713**

Table 4 represents the results for the hybrid implementation, using LexRank and MBART. With this implementation, we can observe that the algorithm obtained low results for the ROUGE metrics. This is understandable given that the summaries are generated by the application of an abstractive approach and may not contain the same words as the original document. In this scenario, the most important metric to take into consideration for ROUGE metrics is ROUGE-1 because it can tell us how many essential key words the summaries have in common. BERTscore is a more informative metric, as it can capture the similarity between the reference and generated summaries even when they have different terms. When comparing the BERTscore results from LexRank to this hybrid implementation we see that they are very close: for "Dataset 1" the results only differ in 0.070, 0.074, and 0.072 for recall, precision and f1-score respectively; and for "Dataset 2" the difference in recall is 0.092, in precision is 0.064, and in f1-score is 0.06.

One important note that was identified when analyzing the summaries manually was that for some documents the model could not generate an actually text and only repeats words or letters as in the following example:

```
78 S. C. S. S. S. S. S. C. - S. S. S. S. S. S. - S. S. S. S. S. S. - S.
S. S. S. S. - S. S. S. S. S. - S. S. S. S. S. S. - S. S. S., [...], S. S.
S. S. S. S
```

This suggests that this malformed generated summaries contributed to lower scores, and that accurate generated summaries may have achieved higher scores than the ones shown in Table 4. Additionally, as observed before in the LexRank implementation, the results for ROUGE and BERTscore are better for "Dataset 2" than for "Dataset 1". However, in this case, we cannot say that it may be because of the length of the input text. In this experience, the input text for MBART was limited to the same length of 1024 tokens. Due to this fact we may conclude that sections *relatório* and *fundamentação de direito* can be more relevant to the summary then the others sections.

Table 5 displays the results for the simplified version of SeburSum algorithm for both datasets. It is important to have in consideration that this results exclude the documents for which this method was unable to generate summaries due to a lack of sentences that could be employed in calculating the similarity between candidates without overlap sentences. The main goal of this implementation is to understand the differences in the generation of summaries based on a sentences-level method, such as LexRank, and a summary-level method, such as SeburSum.

Table 4 Results from hybrid implementation.

Model	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE (Lsum)	R_{BERT}	P_{BERT}	F_{BERT}
LexRank +MBART (Dataset 1)	0.252	0.764	0.144	0.199	0.606	0.636	0.619
LexRank +MBART (Dataset 2)	0.302	0.122	0.183	0.242	0.641	0.669	0.653

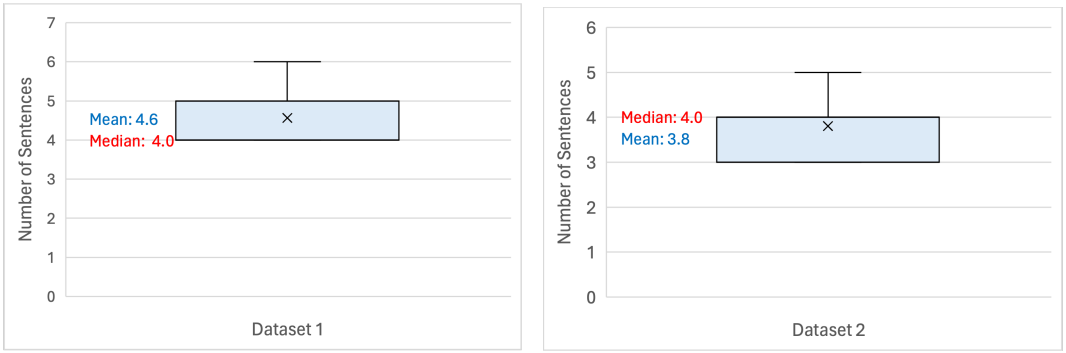
Comparing ROUGE and BERTscore scores when using the LexRank algorithm we can see that there is only a slight margin between them, for both datasets. In "Dataset1', the SeburSum results never overcome the LexRank ones, however, the highest difference was for R_{BERT} of 0.019 and it actually equaled the value for ROUGE-L. For "Dataset2', the largest difference in scores is observed in ROUGE-2 with 0.018, where LexRank is better than SeburSum. There are also some metrics where SeburSum overcomes the LexRank base model: our simplified version obtained the same result for ROUGE-1 score and a higher one on ROUGE-L and R_{BERT}. We can conclude that we can obtain identical information even when using less sentences in the final summaries.

Table 5 Results from SeburSum simplified version implementation.

Model	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE (Lsum)	R_{BERT}	P_{BERT}	F_{BERT}
SeburSum s.v (Dataset 1)	0.347	0.135	0.195	0.265	0.670	0.691	0.679
SeburSum s.v (Dataset 2)	0.401	0.191	0.242	0.340	0.699	0.718	0.706

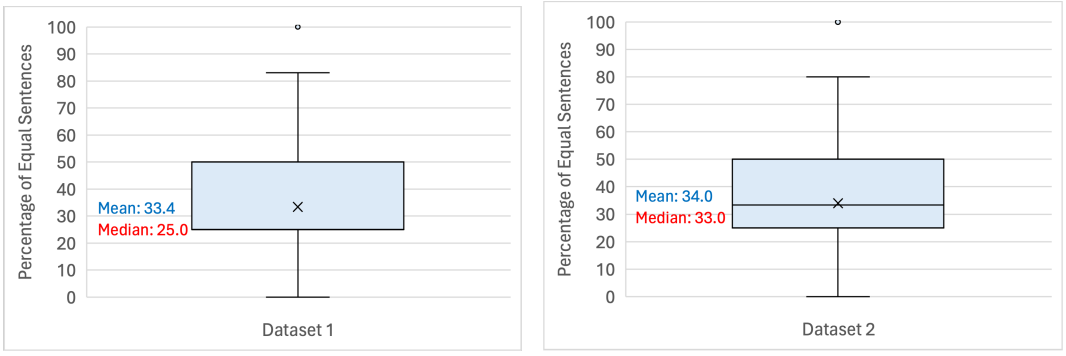
After generating the summaries for both datasets we also analysed two different aspects. Firstly, we verify the length of the generated summaries by examining the number of sentences included in the final summaries. For "Dataset 1", most of the generated summaries typically comprise four sentences (see Figure 6a). For "Dataset 2", we can see that the median number of sentences that were used to generate the final summaries is also four sentences (see Figure 6b).

Secondly, we compared the content of the final summary with the content of a final summary generated with LexRank. More concretely we analyse the percentage of the sentences that were included in the final summaries, using SeburSum simplified version, that matched those present in the final summaries generated by LexRank. In "Dataset 1", only 25% to 50% of the sentences in the final summary were equal to those expected if the final summaries were generated with the same number of sentences with LexRank (see Figure 7a). For "Dataset 2" we can see that number of equal sentences mainly varies between 33% to 50% (see Figure 7b). Based on these results, there is visible a difference in the sentences chosen to generate the summaries with SeburSum simplified version and LexRank, where less then 50% of the sentences are equal.



(a) Dataset 1 **(b)** Dataset2

Figure 6 SeburSum - Distribution of the number of sentences.



(a) Dataset 1 **(b)** Dataset 2

Figure 7 SeburSum - equal sentence percentage.

6 Conclusion

In this paper, we experimented with different summarization techniques to better understand how to overcome the limitations of summarizing long legal documents in the Portuguese language.

First, an extractive approach was used, LexRank, where the lack of performance when choosing the exact sentences for the final summary could be perceived. However, LexRank could still choose sentences that contained relevant information to the summary. One potential solution to overcome this problem could involve fine tuning the LexRank hyper-parameters to extract more representative keywords from the original document, which can be crucial to the context of the generated summary. We also experimented a hybrid approach that obtained scores very close to the LexRank ones and showed efficiency in generating abstractive summaries. An important factor to note is that some of the summaries generated did not represent actual texts. Based on this we can say that there is still room for improving and that fine tuning the MBART model for the specific summarization task, since this approach was created for translation, can be a suitable solution to this problem.

Implementing a simplified version of Sebursum, allowed us to understand that it is possible to obtain similar level of information in summaries with fewer sentences and that the selected sentences included in the final summaries mainly correspond to 30-50% of the top sentences ranked as the most informative with LexRank base implementation.

We could conclude that each model explored can offer different strengths and benefits to the legal document summarization task. For instance, LexRank allows for ranking sentences in a document by relevance; a simple version of SeburSum can create a better candidate summary; and MBART can generate a more fluent and coherent summary than the extractive approaches.

For future work, we propose the development of a model that combines the three summarization approaches. This model would have three phases:

- the initial step where the sentences would be ranked by LexRank based on their importance. Improvements to the LexRank ranking strategy could be made, such as using embeddings to represent sentences instead of using the normal representation sentences used in LexRank, TF-IDF.
- In a second phase, the best sentences would be selected using a version of SeburSum. It could be interest to study how different methods to select the best candidate summary would work, instead of using the maximun score.
- Finally, the selected candidate from the previous phase would be the input to the MBART model, with the goal of obtaining a structured summary that maintains the context of the original document.

References

1 Wubetu Barud Demilie. Comparative analysis of automated text summarization techniques: The case of ethiopian languages. *Wireless Communications & Mobile Computing (Online)*, 2022, 2022.

2 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics, 2019. doi:10.18653/V1/N19-1423.

3 Günes Erkan and Dragomir R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.*, 22:457–479, 2004. doi:10.1613/JAIR.1523.

4 Shuai Gong, Zhenfang Zhu, Jiangtao Qi, Wenqing Wu, and Chunling Tong. Sebursum: a novel set-based summary ranking strategy for summary-level extractive summarization. *J. Supercomput.*, 79(12):12949–12977, 2023. doi:10.1007/S11227-023-05165-8.

5 Yue Huang, Lijuan Sun, Chong Han, and Jian Guo. A high-precision two-stage legal judgment summarization. *Mathematics*, 11(6):1320, 2023.

6 Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. A sentence is known by the company it keeps: Improving legal document summarization using deep clustering. *Artificial Intelligence and Law*, pages 1–36, 2023.

7 Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. Summarization of lengthy legal documents via abstractive dataset building: An extract-then-assign approach. *Expert Syst. Appl.*, 237(Part B):121571, 2024. doi:10.1016/J.ESWA.2023.121571.

8 Ambedkar Kanapala, Sukomal Pal, and Rajendra Pamula. Text summarization from legal documents: a survey. *Artif. Intell. Rev.*, 51(3):371–402, 2019. doi:10.1007/S10462-017-9566-2.

9 Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, pages 7871–7880. Association for Computational Linguistics, 2020. doi:10.18653/V1/2020.ACL-MAIN.703.

10 Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

11 Yang Liu. Fine-tune BERT for extractive summarization. *CoRR*, abs/1903.10318, 2019. arXiv:1903.10318.

12 Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Trans. Assoc. Comput. Linguistics*, 8:726–742, 2020. doi:10.1162/TACL_A_00343.

13 Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020. URL: https://jmlr.org/papers/v21/20-074.html.

14 Fábio Souza, Rodrigo Frassetto Nogueira, and Roberto de Alencar Lotufo. Bertimbau: Pretrained BERT models for brazilian portuguese. In *BRACIS*, volume 12319 of *Lecture Notes in Computer Science*, pages 403–417. Springer, 2020. doi:10.1007/978-3-030-61377-8_28.

15 Yufeng Sun, Fengbao Yang, Xiaoxia Wang, and Hongsong Dong. Automatic generation of the draft procuratorial suggestions based on an extractive summarization method: Bertslca. *Mathematical Problems in Engineering*, 2021:1–12, 2021.

16 Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with BERT. In *ICLR*. OpenReview.net, 2020. URL: https://openreview.net/forum?id=SkeHuCVFDr.

17 Justin Zhao, Timothy Wang, Wael Abid, Geoffrey Angus, Arnav Garg, Jeffery Kinnison, Alex Sherstinsky, Piero Molino, Travis Addair, and Devvret Rishi. Lora land: 310 fine-tuned llms that rival gpt-4, A technical report. *CoRR*, abs/2405.00732, 2024. doi:10.48550/arXiv.2405.00732.

18 Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. Extractive summarization as text matching. In *ACL*, pages 6197–6208. Association for Computational Linguistics, 2020. doi:10.18653/V1/2020.ACL-MAIN.552.