A summary on the use of NLP Based Latent Semantic Analysis for Legal Text Summarization

Neha Devi Shakya
gusshakne@student.gu.se

Abstract

The research offers an automatic text summary method that provides short and relevant summaries from lengthy judgments by capturing concepts inside a single document using a natural language processing technique called latent semantic analysis (LSA). Depending on the type of input instance(criminal or civil), a single document untrained approach or a multi-document trained technique was used.

1 Introduction

Legal text processing is a critical issue in today's world due to the large amount of information available. Extracting only the valuable parts from a large amount of data proves of great importance to lawyers and ordinary citizens. Legal editors are often hired to prepare human-generated summaries, which takes a long time and does not guarantee a dependable outcome. To solve this issue, the paper (Merchant and Pande, 2018) suggests using a system that can automatically generate summaries from the given cases, i.e. through automated text summarization. This process aims to create short summaries from the lengthy text without changing the main idea of the original document and also ensures that all the essential points are included in the summary.

Over the years, many text summarizing methods have been developed. Deep learning is used in the most recent and successful ways. It is dependent on previous data and does not examine semantics inside a single new document. Because each case appears to be unique in terms of legal text summarization, depending on previously encountered data is not an intelligent technique. As a result, the paper's suggested approach is to employ latent semantic analysis, which considers similar concepts inside a piece of text. Another issue with legal data is that criminal and civil judgments differ. While each civil case is unique, criminal proceedings tend to follow a similar pattern. As a result, we employ both a single document method and a multi-document strategy.

2 Dataset

The data used consists of legal judgements issued by the Indian judiciary system from official websites with extensions such as .nic and .gov. The data used consists of legal judgements issued by the Indian judiciary system from official websites, including Supreme Court, high court and district court cases, with extensions such as .nic and .gov. A web scraper built-in Python utilizing the Selenium and Beautiful Soup modules was used to retrieve this data. The data had more than 100 thousand documents, from which a group of 50 were chosen, which included records from various courts and sources and included both civil and criminal judgments.

3 Proposed System

3.1 Pre Processing

The input document is an XML file that contains the petitioner's name, the respondent's name, the cases referenced, the headnote, and the judgement. However, only the judgement portion of the input document was taken into account. It was further cleaned to remove the punctuation part of abbreviations so that the model does not consider wrong ends of statements, and stemming was used to convert words into their root words to be evaluated as identical.

3.2 Latent Semantic Analysis(LSA) Model

Latent Semantic Analysis (LSA), also known as Latent Semantic Indexing (LSI), is an automated, unsupervised statistical-algebraic summarising technique that uses an extractive ap-

proach to scan documents and discovers latent semantic relationships between words and phrases. It generates a collection of concepts by analysing the relationship between a set of terms and the texts. The values in the input matrix A represent the relevance of terms for each phrase in Singular Value Decomposition (SVD). Except for the diagonal, it has zeroes everywhere. Eigenvalues of A*A are the squared values of diagonal entries of Σ, and the eigenvectors are the columns of V. To identify concepts, any input matrix of a document is divided into a unique set of matrices U, V, and Σ.

They improved the system by using two different approaches, one for civil and the other for criminal cases. Criminal judgements have a similar flow and often use identical terms, so they implemented a multi-document trained approach, as scores previously generated from SVD of multiple documents impacted newly generated ones. On the other hand, civil judgements rarely had repeated terms and ideas. Therefore an untrained LSA model approach was implemented where the word score of every civil judgement input matrix after SVD was not preserved. For all civil documents, it was independent of any previously produced scores.

3.3 Sentence Selection

After completing the LSA model's analysis, the top sentences were chosen for the final summary. The summary length was limited to 5% of the original length, sufficient for people to read. Since the last sentence communicates the ultimate decision, it was included unconditionally and the sentences chosen by the LSA model. Including this statement increased the system's overall accuracy.

4 Results

Our summarization model was successfully implemented, and the results were generated.

1. Experimental Setup: Python was used to create two summarization models. One of them used training by permitting multi-document criminal summaries to be sent to a single program for execution. The alternative model provided civil summaries to the program iteratively, with a fresh execution for each document.

2. System Evaluation: The system evaluation is depicted by an example sentence System generated: "The Trial Court found all the accused guilty of the charges and convicted and sentenced them." Reference: "Accused found guilty and sentenced to jail."

The Recall-Oriented Understudy for Gisting Evaluation (ROUGE), which compares system-generated summaries against reference or human-generated summaries, was used to analyse the results. They found that the words in the machine-generated summaries do not always match those in the human-generated ones. There is a significant length difference between the algorithm and human-generated summaries due to the former being extractive and taking entire sentences into account. When evaluating the system, the model achieved an average ROGUE-1 (overlap of a single word (1-gram) between the system and reference summaries) score of 0.58, which was more than adequate than ROUGE-2 (overlap of bigrams) and ROUGE-L (using the longest common subsequence). Furthermore, when thoroughly analysed by prominent lawyers, the ideas in both summaries were nearly identical, demonstrating the model's effectiveness.

5 Conclusion

They successfully constructed a legal text summariser based on our proposed model, which employs latent semantic analysis, a natural language processing technique. The original document's essential ideas were preserved in the summary generated by the system. Since the ROUGE evaluation method is not entirely effective, they intend to analyse the model in the future based on entire concepts rather than just similar words. Furthermore, utilising the LSA technique occasionally results in a break in general continuity, and they intend to adopt the suggested system to address this issue. Finally, they intend to implement their system on a mobile platform to improve its use.

References

Kaiz Merchant and Yash Pande. 2018. https://doi.org/10.1109/ICACCI.2018.8554831 NLP Based Latent Semantic Analysis for Legal Text Summarization. In *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1803–1807.