Legal Knowledge and Information Systems
E. Schweighofer (Ed.)

doi:10.3233/FAIA210326

An Information Retrieval Pipeline for Legislative Documents from the Brazilian Chamber of Deputies

Ellen SOUZA [a,b,1], Douglas VITÓRIO [a,d], Gyovana MORIYAMA [b], Luiz SANTOS [b], Lucas MARTINS [b], Mariana SOUZA [b], Márcio FONSECA [e], Nádia FÉLIX [b,c], André C. P. L. F. CARVALHO [b], Hidelberg O. ALBUQUERQUE [a,d], and Adriano L. I. OLIVEIRA [d]

[a] *MiningBR Research Group, Federal Rural University of Pernambuco, Brazil*
[b] *Institute of Mathematics and Computer Sciences, University of São Paulo, Brazil*
[c] *Institute of Informatics, Federal University of Goiás, Brazil*
[d] *Centro de Informática, Federal University of Pernambuco, Brazil*
[e] *Chamber of Deputies, Brasilia, Brazil*

Abstract. This work investigates information retrieval methods to address the existing difficulties on the *Preliminary Search*, part of the law making process from the Brazilian Chamber of Deputies. For such, different preprocessing approaches, stemmers, language models, and BM25 variants were compared. Two legislative corpora from Chamber were used to build and validate the pipeline. All texts were converted to lowercase and had stopwords, accentuation, and punctuation removed. Words were represented by their stem combined with word unigram and bigram language models. Retrieving the bill that was originated from a specific job request, the BM25L with Savoy stemmer reached a R@20 of 0.7356. After removing queries with inconsistencies or which made reference exclusively to attachments, to other job requests, or to bills, the R@20 increased to 0.94.

Keywords. Legal Information Retrieval, Legislative Document Retrieval, Brazilian Portuguese, BM25

1. Introduction

The Brazilian Chamber of Deputies was founded over two hundred years ago and has more than 20 thousand employees, including citizen representatives from all over the country. Since its founding, the Chamber has processed more than 144 thousand bills [1]. Each bill needs to be formalized as an initial legislative document *draft* and an optional justification document, which are submitted for discussion and voting. For a typical bill, a large number of documents is produced and aggregated in different stages of processing. This content, generated by the members of the parliament, is massive and keeps increasing. Besides, the unstructured nature of these documents makes their organization, access, and retrieval a challenging task [1].

[1] Corresponding Author: ellen.ramos@ufrpe.br.

A bill is submitted to the Legislative Consulting (CONLE), an advisory body of the House, whose main role is to provide the necessary support to the law making process. The CONLE has an internal team of specialists and researchers in 22 legal subjects, including economics, technology, and transportation. With the increasing demand for legislative production, a remarkable amount of legislative consulting requests is redundant, regarding other proposals already under analysis by the CONLE, and even existing laws. As consequence, a large deal of effort from the consulting team is devoted to this process, called *Preliminary Search*.

This work investigates the use of information retrieval (IR) methods to address these legislative production issues. Given a set of legislative documents and a query document (i.e., a job request), the system filters and ranks the documents according to their relevance to the query. The research is conducted in the context of the *Ulysses* project, an institutional set of artificial intelligence initiatives with the purpose of increasing transparency, improving the Chamber's relationship with citizens, and supporting the legislative activity with complex analysis [2]. This paper is organized as follows: Section 2 presents the major related studies. Section. 3 details the IR pipeline for Brazilian legislative documents. Section. 4 presents and discusses the obtained results. Section. 5 brings the conclusion and highlights future works.

2. Related Work

The only study found by the authors performing legislative document retrieval with data written in European Portuguese was [3]. In this study, a unsupervised document similarity algorithm is presented using sets of synonyms.The author's goal was to rank legislative documents based on their relevance to a query, regardless of the language used. Using the English, Spanish, French, and European Portuguese editions of the *JRC-Acquis* dataset they compared their unsupervised synset-based approach to a semi-supervised category-based one, reaching inferior results. The algorithm's performance was evaluated in terms of P@k (Precision at k documents): P@3, P@5, and P@10; achieving the results of 0.78, 0.75, and 0.71, respectively, for the Portuguese dataset.

Gomes and Ladeira [4] empirically evaluated the framework for case-law retrieval of the Brazilian Superior Court of Justice (STJ), comparing its legacy system to approaches based on text similarity: the TF-IDF traditional retrieval model, BM25, and four Word2Vec models. The STJ's system uses Boolean queries and the authors wanted to use free text as queries without any operator. The results reported, using NDCG@25 (Normalized Discounted Cumulative Gain with a cut off of 25 documents), demonstrated the superiority of BM25 based systems in this task, with a mean NDCG@25 equal to 0.752. Although the paper explored information retrieval in the real-world legal domain, it used a jurisprudence scenario, while, here, we are using legislative documents.

Another work investigating jurisprudence document retrieval and the impact of Stemming on the retrieval of real documents from the Court of Justice of the State of Sergipe (TJSE), in Brazil [5]. The authors compared four radicalization algorithms (Porter, RSLP, RSLP-S, and UniNE) to evaluate: 1) their gain in dimensionality reduction; 2) their predictive performance regarding legal document retrieval. the Okapi BM25 was used and evaluated by MAP (Mean Average Precision), MPC (Mean of Precision@10), and MRP (the average of R-Precision). RSLP obtained the largest dimen-

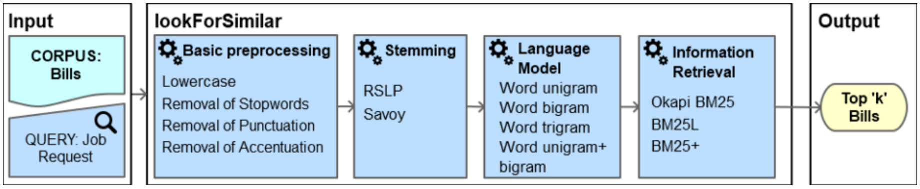

Figure 1. Brazilian Portuguese legislative information retrieval pipeline.

sionality reduction, while RSLP-S and UniNE were the best Stemming algorithms for IR, with the best MAP results, 0.87 and 0.88, respectively. According to the experimental results, the use of radicalization deteriorated the BM25 performance.

Chalkidis et al. [6] investigated regulatory compliance in EU and United Kingdom (UK) legislation using IR. They proposed a new approach, called Regulatory Information Retrieval (REG-IR), for document-to-document IR, in which a query is an entire document. The authors used two groups of legislation: EU directives and UK laws. REG-IR uses a neural IR system with a two-step pipeline: first, an IR algorithm (pre-fetcher) retrieves the top-k documents related to a query; next, a neural model re-ranks the documents. As pre-fetching algorithms, the authors evaluated Okapi BM25, W2V-CENT, BERT, S-BERT, LEGAL-BERT, C-BERT (BERT fine-tuned to predict EUROVOC concepts), and an ensemble of C-BERT and BM25; alongside six re-ranking techniques. Using R@100 (Recall at 100 documents) as metric to evaluate the pre-fetchers and R@20, NDCG@20, and R-Precision for the re-ranks, C-BERT was the best pre-fetcher for the datasets used, while the neural re-ranks failed to improve the retrieval performance.

Cantador and Sánchez [7] proposed a new approach for IR of parliamentary content, such as debate transcripts and laws proposals. The authors present a case study, in the Spanish Congress of Deputies, where they integrate their approach into *Parlamento2030*, an online platform that monitors parliamentary activity. They investigated the application of the Generalized Vector Space Model (GVSM) to the *Parlamento2030* dataset. The GVSM incorporates a semantic relatedness measure into the Vector Space Model (VSM), combined with an ontology-based document representation model. The authors used average P@5, P@10, P@15, and P@20. The results obtained (0.733, 0.683, 0.656, 0.600) were better than those obtained using just the matches of query and document key terms (0.633, 0.483, 0.422, 0.358).

3. The Method Used

Figure 1 presents the Brazilian Portuguese legislative IR pipeline[2]. The *job requests* are the queries and represent the user's input to the system. While the bills are the output answer, ranked according to a matching rate between the documents and the query (Subsection 3.1). We also evaluated basic preprocessing techniques (Subsection 3.2), two stemmers for the Portuguese language (Subsection 3.3), four word n-gram language models (Subsection 3.4), and three BM25 variants (Subsection 3.5).

[2] https://github.com/Convenio-Camara-dos-Deputados/BM25-Experiments

3.1. Corpora

Two legislative corpora from the Brazilian Chamber of Deputies were used to build and validate this pipeline: the *Bills* and the *Job Request* corpora. The former is available [3], while the latter has confidential information and cannot be made available.

The three most common types of bills were selected for the *Bills* Corpus: Law Project (Projeto de Lei - PL), Complementary Law Project (Projeto de Lei Complementar - PLC), and Constitutional Amendment Proposal (Proposta de Emenda Constituicional - PEC). The final corpus has 48,555 proposals. The attribute *imgArquivoTeorPDF*, which is the bill itself, was used in the experiments. It has an average of 300 words.

The *Job Request* corpus represents the user's query and contains 295 anonymized *Job Requests*. Data identifying the parliamentarian who made the request to CONLE were removed. This corpus has two attributes. The former contains the number of the bill that was originated from the *Job Request* specified in the latter attribute. Table 1 shows examples of parliamentarians' *Job Requests* (i.e. queries). Most requests have between 10 and 40 words.

Table 1. Samples from anonymized Job Request corpus.

Originated bill	**Job Request (user's query)**
PL XXXX/2019	Projeto para restabelecer na CLT a proibição de terceirização para atividade fim (Project to prohibit the outsourcing of core activity in the CLT)
PL XXXX/2019	Criação de PL, com base nos dois esboços encaminhados anexo. (Make of bill based on the two sketches sent in the attachment)
PL XXXX/2019	Solicito parecer pela aprovação de acordo com a solicitação XXXX/2019. (Request an opinion for the approval according to job request number XXXX/2019.)
PL XXXX/2019	Complementar parecer em função da apensação do PL XXXX/19 ao mesmo (Complementary opinion according to the PL XXXX/19)
PL XXXX/2019	Parlamentar solicita aprovação (Parliamentarian requests approval)

3.2. Basic Preprocessing

Both corpora presented in previous subsections had their texts converted to lowercase and had stopwords, accentuation, and punctuation removed. We evaluated each technique separately and all techniques together. The preprocessing techniques were performed using the Python NLTK. For the stopword removal, we used a Portuguese stopword list.

3.3. Stemming

The main purpose of stemming is to reduce the inflected words into its root form or stem. Thus, words can be mapped to the same concept, improving the process of IR, regarding its ability to index documents and to reduce data dimensionality [5]. RSLP and Savoy algorithm were chosen because of their effectiveness in the retrieval of documents [12,5, 13,14].

[3] https://drive.camara.leg.br/s/c3p2nLgLRcMz6eX

- RSLP (Removedor de Sufixos da Lingua Portuguesa): a rule-based algorithm developed by [9] and improved by [10]. Like Porter, it applies successive steps to remove the suffixes. As it was developed specially for Portuguese, it has more rules than Porter. It has 8 steps and a list of exception which prevents the algorithm from removing suffixes of words that have endings that are similar to suffixes.
- Savoy (UniNE): developed by Jacques Savoy in 2006, it presents stemmers for various languages, including Portuguese. The algorithm is simpler than the others, as it has less rules. It removes inflections attached to both nouns and adjectives, based on rules for the plural and feminine form. Our implementation is based on [11].

3.4. Language Model

An n-gram language model predicts the probability of a given n-gram within any sequence of words in the language. It is widely used in text mining [15,16], including in the legal domain [19]. An n-gram is a contiguous sequence of n items from a given sequence of text. These items can be phonemes, characters, words, and others. Unigram refers to n-gram of size 1, bigram refers to n-gram of size 2, and so on. In this work, we evaluated four different word n-gram combinations [17,15,18]

3.5. Information Retrieval

BM25 [20] is the most well-known scoring function for “bag of words” document retrieval [21]. It is derived from the binary independence relevance model to include within-document term frequency information and document length normalization in the probabilistic framework for IR [22]. The algorithm has also been used successfully in the retrieval of legal documents [5,4,6,23]. We implemented the variants presented in [24].

Okapi BM25 [20] scoring function estimates the relevance of a document d to a query q, based on the query terms appearing in d, regardless of their proximity within d: where q_i is the *i-th* query term, with $idf(q_i)$ inverse document frequency and $tf(q_i,d)$ term frequency. The formula for the Okapi BM25 is presented below:

$$score(q_i,d) = \frac{IDF(q_i) \cdot TF(q_i,d)(k_1+1)}{TF(q_i,d) + k_1(1-b+b \cdot \frac{|d|}{L})} \tag{1}$$

where $TF(q_i,d)$ is the frequency of term q_i in document d, $IDF(q_i)$ is the inverse document frequency of term q_i, $\mid d \mid$ is the number of terms in document d and L is the average number of terms per document. The effectiveness of BM25 is highly dependent on properly selecting the values of k_1 and b. In traditional *ad hoc* IR, k_1 is typically evaluated in the range [0, 3] (usually $k_1 \in [0.5, 2.0]$); b needs to be in [0, 1] (usually $b \in [0.3, 0.9]$) [24]. We defined the following parameters in our experiments: $k_1 = 1.5$, $b = 0.75$, and $\varepsilon = 0.25$.

BM25L [25] is built on the observation that Okapi penalizes more longer documents compared to shorter ones. It *shifts* the term frequency normalization formula to boost scores of very long documents. Finally, BM25+ encodes a general approach for dealing with the issue that ranking functions unfairly prefer shorter documents over longer ones. The proposal is to add a lower-bound bonus when a term appears at least one time in a document [26]. The difference with BM25L is a constant δ to the TF component.

3.6. Evaluation

We have only one relevant document for each query(see Table 1, because of this, we are evaluating the results in terms of *Recall* (R), which is the fraction of relevant documents that are retrieved. We are analyzing the results with R@20 (Recall at 20 documents).

4. Experimental Results

Table 2 presents the experimental results. We checked if the bill which was originated by a specific job request appears in the top-20 relevant documents retrieved by the BM25 algorithms. BM25L achieved the best results in almost all experiments, outperforming the Okapi variant which has been widely used and performed better in previous works [5, 4,6]. This may be due to the size of the documents used in our experiments.

Table 2. Experimental results with R@20 (Recall at 20 documents).

No.	Originated Bill	R@20		
	basic preprocessing	**Okapi**	**BM25L**	**BM25+**
1	no preprocessing	0,6441	0,6678	0,6542
2	lowercase	0,6542	0,6983	0,6814
3	lowercase + punctuation removal	0,6678	**0,7153**	0,6847
4	lowercase + punctuation and acetuation removal	0,6780	**0,7153**	0,6847
5	lowercase + punctuation, acetuation, and stopword removal	0,7085	**0,7153**	0,6847
	stemming	**Okapi**	**BM25L**	**BM25+**
6	stemming (RSLP)	0,6271	0,6847	0,6508
7	stemming (Savoy)	0,6203	0,6712	0,6441
8	lowercase + punctuation, acetuation, and stopword removal + stemming (RSLP)	0,7085	**0,7288**	0,6915
9	lowercase + punctuation, acetuation, and stopword removal + stemming (Savoy)	0,6949	0,7186	0,6881
	word n-gram	**Okapi**	**BM25L**	**BM25+**
10	bigram	0,5898	0,5864	0,5729
11	trigram	0,4881	0,4881	0,4983
12	unigram + bigram	0,6542	**0,6712**	0,6441
	word n-gram + basic preprocessing	**Okapi**	**BM25L**	**BM25+**
13	lowercase + punctuation, acetuation, and stopword removal + bigram	0,5932	0,5898	0,5932
14	lowercase + punctuation, acetuation, and stopword removal + trigram	0,4712	0,4712	0,4712
15	lowercase + punctuation, acetuation, and stopword removal + unigram and bigram	**0,7085**	0,7051	0,6983
	word n-gram + basic preprocessing + RSLP	**Okapi**	**BM25L**	**BM25+**
16	lowercase + punctuation, acetuation, and stopword removal + stemming (RSLP) + bigram	0,6373	0,6305	0,6305
17	lowercase + punctuation, acetuation, and stopword removal + stemming (RSLP) + trigram	0,4881	0,4847	0,4847
18	lowercase + punctuation, acetuation, and stopword removal + stemming (RSLP) + unigram and bigram	0,7220	**0,7322**	0,7017
	word n-gram + basic preprocessing + Savoy	**Okapi**	**BM25L**	**BM25+**
19	lowercase + punctuation, acetuation, and stopword removal + stemming (Savoy) + bigram	0,6237	0,6237	0,6237
20	lowercase + punctuation, acetuation, and stopword removal + stemming (Savoy) + trigram	0,4780	0,4780	0,4746
21	lowercase + punctuation, acetuation, and stopword removal + stemming (Savoy) + unigram and bigram	0,7288	**0,7356**	0,7051

For the BM25L, analyzing the basic preprocessing techniques, there was no difference between the removal of punctuation, accentuation, and stopwords. In order to reduce data dimensionality, two Stemming algorithms were evaluated, improving the pipeline result. RSLP performed better with basic preprocessing tecniques (Table 2, line 8), but Savoy performed slightly better in combination with unigram and bigram (Table 2, line 21). This was not observed by Oliveira and C. Junior [5], in whose study radicalization deteriorated the Okapi BM25 performance. Although Savoy showed a slightly better result than RSLP when combined with unigram and bigram, RSLP obtained the largest dimensionality reduction in the retrieval of legal documents [5]. Therefore, the use of the word n-gram alone did not improve the results, but in combination with basic prepossessing (Table 2, line 5) and stemming (Table 2, line 21) the technique improved the pipeline result.

Considering our best result (Table 2, line 21), the algorithm failed to retrieve 55 queries from a total of 295 job requests (queries). The analysis of these queries showed the following problems with our *Job Request* corpus: 7 queries made reference only to attachments; (Table. 1, line 2); 6 queries made reference only to other job requests (Table. 1, line 3); 10 queries made reference only to a bill name (Table. 1, line 4); and 11 queries did not refer to any subject (Table. 1, line 5). For those 34 job requests, the BM25L needs more information in addition to the text presented in the query. Therefore, analyzing the remaining 21 failed job requests, it was possible to observe also that, for seven requests, the text presented in the query did not refer to the bill associated to it, increasing the BM25L R@20 to 0.94.

5. Conclusion and Future Work

This paper explored IR for the legislative domain in a real-world scenario. Our preprocessing approach converts text to lowercase, removes stopwords, accentuation, and punctuation. We evaluated RSLP and Savoy Stemming algorithm to reduce dimensionality, improving the performance of the IR pipeline. A combination of unigram and bigram also improved BM25 results. We compared different BM25 algorithms and the L outperformed the Okapi and Plus variants.

We plan to use word embedding language models to capture semantic knowledge. As highly relevant documents are more valuable than marginally [28], we parented to perform a rank evaluation in our pipeline as in [6], which have applied neural models to improve ranking ordering. Currently, we are evaluating Named Entity Recognition to expand those queries, as well as considering the user relevance feedback to improve the performance of the whole IR pipeline.

References

[1] Brandt MB. Modelagem da informação legislativa: arquitetura da informação para o processo legislativo brasileiro. Faculdade de Filosofia e Ciências da Universidade Estadual Paulista (UNESP); 2020.

[2] Almeida PGR. Uma jornada para um Parlamento inteligente: Câmara dos Deputados do Brasil. Red Información. 2021;24. Available from: https://www.redinnovacion.org/revista/red-información-edición-nº-24-marzo-2021.

[3] Badenes-Olmedo C, García JLR, Corcho Ó. Legal document retrieval across languages: topic hierarchies based on synsets. CoRR. 2019;abs/1911.12637.

[4] Gomes T, Ladeira M. A New Conceptual Framework for Enhancing Legal Information Retrieval at the Brazilian Superior Court of Justice. In: Proceedings of the 12th International Conference on Management of Digital EcoSystems; 2020. p. 26-29.

[5] Oliveira RA, C Junior M. Experimental Analysis of Stemming on Jurisprudential Documents Retrieval. Information. 2018;9(2).

[6] Chalkidis I, Fergadiotis M, Manginas N, Katakalou E, Malakasiotis P. Regulatory Compliance through Doc2Doc Information Retrieval: A case study in EU/UK legislation where text similarity has limitations. arXiv preprint arXiv:210110726. 2021.

[7] Cantador I, Sánchez LQ. Semantic Annotation and Retrieval of Parliamentary Content: A Case Study on the Spanish Congress of Deputies. In: Proc. of the Joint Conference of the Information Retrieval Communities in Europe. vol. 2621; 2020.

[8] Hotho A, Nürnberger A, Paaß G. A Brief Survey of Text Mining. Journal for Computational Linguistics and Language Technology. 2005:1-37.

[9] Orengo VM, Huyck C. A stemming algorithm for the Portuguese language. Proceedings Eighth Symposium on String Processing and Information Retrieval. 2001:186-193.

[10] Orengo VM, Buriol LS, Coelho AR. A study on the use of stemming for monolingual ad-hoc Portuguese information retrieval. In: Workshop of the Cross-Language Evaluation Forum for European Languages. Springer; 2006. p. 91–98.

[11] Savoy J. Light Stemming Approaches for the French, Portuguese, German and Hungarian Languages. In: SAC '06: Proceedings of the 2006 ACM symposium on Applied computing. New York, NY, USA: Association for Computing Machinery; 2006. p.1031–1035.

[12] de Oliveira RA, Colaço Júnior M. Assessing the Impact of Stemming Algorithms Applied to Judicial Jurisprudence-An Experimental Analysis. In: International Conference on Enterprise Information Systems. vol. 2. SCITEPRESS; 2017. p. 99–105.

[13] Flores FN, Moreira VP, Heuser CA. Assessing the impact of stemming accuracy on information retrieval. In: International Conference on Computational Processing of the Portuguese Language. Springer; 2010. p. 11–20.

[14] Flores FN, Moreira VP. Assessing the impact of Stemming Accuracy on Information Retrieval – A multilingual perspective. Information Processing & Management. 2016;52(5):840–854.

[15] Ravi K, Ravi V. A survey on opinion mining and sentiment analysis: Tasks, approaches and applications. Knowledge-Based Systems. 2015;89.

[16] Castro DW, Souza E, Vitório D, Santos D, Oliveira ALI. Smoothed n-gram based models for tweet language identification: A case study of the Brazilian and European Portuguese national varieties. Applied Soft Computing. 2017;61:1160–1172.

[17] Pang B, Lee L. Opinion Mining and Sentiment Analysis. Foundations and Trends® in Information Retrieval. 2008;2(1–2):1–135.

[18] Tripathy A, Agrawal A, Rath SK. Classification of sentiment reviews using n-gram machine learning approach. Exp Sys with App. 2016;57:117 – 126.

[19] Katz DM, Bommarito MJ, Seaman J, Agichtein E. Legal n-grams? A simple approach to track the evolution of legal language. Frontiers in Artificial Intelligence and Applications. 2011;235(Vienna):167–168.

[20] Robertson S, Walker S, Jones S, Hancock-Beaulieu M, Gatford M. Okapi at TREC-3. In: TREC; 1994.

[21] Kamphuis C, de Vries AP, Boytsov L, Lin J. Which BM25 Do You Mean? A Large-Scale Reproducibility Study of Scoring Variants. In: Advances in Information Retrieval; 2020. p. 28–34.

[22] Robertson S, Zaragoza H. The Probabilistic Relevance Framework: BM25 and Beyond. Foundations and Trends in Information Retrieval. 2009;3:333–389.

[23] Bansal A, Bu Z, Mishra B, Wang S, Ashley KD, Grabmair M. Document Ranking with Citation Information and Oversampling Sentence Classification in the LUIMA Framework. In: JURIX; 2016.

[24] Trotman A, Puurula A, Burgess B. Improvements to BM25 and language models examined. ACM International Conference Proceeding Series. 2014;27-28-Nove:58–65.

[25] Lv Y, Zhai C. When Documents Are Very Long, BM25 Fails! In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval; 2011. p. 1103–1104.

[26] Robertson S, Zaragoza H, Taylor M. Simple BM25 Extension to Multiple Weighted Fields. In: Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management. CIKM '04; 2004. p. 42–49.

[27] Manning CD, Raghavan P, Hinrich S. An Introduction to Information Retrieval Draft. c; 2009. Available from: http://www-nlp.stanford.edu/IR-book/.

[28] Järvelin K, Kekäläinen J. IR Evaluation Methods for Retrieving Highly Relevant Documents. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval; 2000. p. 41–48.