Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 1449–1458
Marseille, 11–16 May 2020

VICTOR: a dataset for Brazilian legal documents classification

Pedro H. Luz de Araujo, Teofilo E. de Campos, Fabricio A. Braz, Nilton C. da Silva
CiC and FGA, University of Brasília (UnB),
Brasília - DF, Brazil
pedrohluzaraujo@gmail.com, {teodecampos, fabraz, niltoncs}@unb.br

Abstract

This paper describes VICTOR, a novel dataset built from Brazil's Supreme Court digitalized legal documents, composed of more than 45 thousand appeals, which includes roughly 692 thousand documents—about 4.6 million pages. The dataset contains labeled text data and supports two types of tasks: document type classification; and theme assignment, a multilabel problem. We present baseline results using bag-of-words models, convolutional neural networks, recurrent neural networks and boosting algorithms. We also experiment using linear-chain Conditional Random Fields to leverage the sequential nature of the lawsuits, which we find to lead to improvements on document type classification. Finally we compare a theme classification approach where we use domain knowledge to filter out the less informative document pages to the default one where we use all pages. Contrary to the Court experts' expectations, we find that using all available data is the better method. We make the dataset available in three versions of different sizes and contents to encourage explorations of better models and techniques.

Keywords: text classification, legal domain, language resources

1. Introduction

Brazil's legal system suffers from an unreasonably large number of lawsuits (de Cássia Carvalho Lopes, 2017). To put matters into perspective, about 80 million lawsuits were awaiting judgement in 2017. That is almost one process for every three Brazilians. The period from 2009 to 2017 saw an increase of 19.4 million lawsuits (Fariello, 2018). In addition, the average processing time of lawsuits can reach more than seven years in some cases. The long waiting times impact Brazil's legal certainty and represent greater budgetary requirements—Brazil spent R$ 90.7 billion in 2017 to maintain the judiciary, approximately 22 billion dollars (Secretaria de Comunicação Social do Conselho Nacional de Justiça, 2018).

Our work aims to apply Natural Language Processing (NLP) and Machine Learning (ML) techniques to Brazil's Supreme Court (*Supremo Tribunal Federal* or STF) cases to help overturn this scenario. The STF receives roughly 42 thousand cases each semester, taking 22 thousand hours for humans to sort through. That time could be better spent at more complex stages of the judicial work flow, for instance the ones requiring legal reasoning.

Most of the cases reach the court as PDF files with raster scanned documents. Approximately 10% of these are unstructured, containing several unindexed documents ranging from petitions and orders to rulings. Therefore, as a first goal we explore and evaluate methods for automatically classifying document types. The documents originate in different Brazilian courts and often contain visual noise (handwritten annotations, stamps, stains). So the main challenges here are the intra-class diversity and the quality of the scanned documents.

In addition, lawsuits pertaining to the STF belong to one or more general repercussion (*repercussão geral*) themes that are presently checked by humans during the initial processing of the suit. As our final goal we train and evaluate a series of models that assign themes to suits. In this case, the central difficulty is the size of the suits, which can contain dozens of documents.

Our main contribution is VICTOR[1], a dataset of legal documents belonging to STF's suits labeled by a team of experts. We hope that this can help other researchers to explore NLP and ML applied to the legal field, document analysis, text classification and multilabel classification. Our second contribution is a benchmark that compares a series of models we evaluate for each goal: document type classification and lawsuit theme assignment.

The rest of this paper is organized as follows. In Section 2, we introduce related works. In Section 3, we discuss the dataset and its creation process. We present the models explored and the experiments involved and discuss the results obtained in Sections 4 and 5 regarding the first and second goals, respectively. Section 6 concludes the paper.

2. Related Works

2.1. Text classification

A traditional well-performing baseline for text classification is representing a document as a bag-of-words and give that as input to a classifier like Naïve Bayes or Support Vector Machines (Joachims, 1998). Such representation is invariant to word-order, a property that may hinder performance in applications such as sentiment classification, where word positioning can completely change the semantics of the sentence. Using n-grams instead of only 1-grams (words) can mitigate that problem. Joulin et al. (2017) propose a shallow model that uses n-gram features and hierarchical softmax to efficiently train on large datasets. Liu et al. (2016) propose a semi-supervised text classification method that combines boosting and examples that do not belong to any class, which is shown to particularly benefit problems with few labeled examples.

The popularization of deep neural networks gave rise to the creation of many architectures for text categorization. Zhang et al. (2015) and Conneau et al. (2017) independently show that a character-level CNN surpasses shallow models' performances on large datasets. Johnson and

[1]Data available at `http://ailab.unb.br/victor/lrec2020/`

Zhang (2016) were able to improve the state of the art by using a word-level LSTM network with pooling. Howard and Ruder (2018) introduce a transfer learning method for any NLP task that outperforms the state-of-the-art text classifiers, in addition to requiring much less data to match the performance of a model trained from scratch.

2.2. Natural Language Processing and Machine Learning in the legal domain

Several works have explored the use of Natural Language Processing and Machine Learning techniques to analyze legal documents. Named entity recognition (NER) has been used to automatically extract relevant entities from legal text (Dozier et al., 2010; Cardellino et al., 2017; Luz de Araujo et al., 2018). Automatic summarization has been employed to help manage the great amount of information legal employees are required to process (Kanapala et al., 2017; Galgani et al., 2012; Kumar and Raghuveer, 2012; Kim et al., 2013). In addition, topic models have been used to analyze large corpora of legal documents (Carter et al., 2016; Remmits, 2017; O'Neill et al., 2016).

Text classification in the legal domain is used in a number of different applications. Katz et al. (2014) use extremely randomized trees and extensive feature engineering to predict if a decision by the Supreme Court of the United State would be affirmed or reversed, achieving an accuracy of 69.7%. Aletras et al. (2016), in a similar fashion, trained a model to predict, given the textual content of a case from the European Court of Human Rights, if there has been a violation of human rights or not. The paper employed n-grams and topics as inputs to a SVM, reaching an accuracy of 79%. Şulea et al. (2017) trained a linear SVM on text descriptions of cases from the French Supreme Court, obtaining a 90% F1 score in law area prediction (eight classes) and a 96.9% F1 score in ruling prediction (six classes). Undavia et al. (2018) evaluated a series of classifiers (CNN, RNN, SVM and logistic regression) trained on a dataset of cases from the American Supreme Court. Their best performing model, a CNN, was able to achieve an accuracy of 72.4% when classifying the cases into 15 broad categories and 31.9% when classifying over 279 finer-grained classes.

3. The Dataset

The VICTOR dataset is composed of 45,532 Extraordinary Appeals (Recursos Extraordinários) from the STF. Each suit in turn contains several different documents, ranging from the appeal itself to certificates and rulings, totaling 692,966 documents comprising 4,603,784 pages.

The Court provided the VICTOR data in the form of PDF files where each file either represents a particular document or is an unstructured volume containing several documents. In the former case, the suits were manually annotated by experts from the Court staff with labels for the document classes, totalizing 44,855 suits with 628,820 documents.

The first issue we faced was extracting the text from the PDF files. A significant part of the provided data is in the form of images obtained by scanning printed documents, which often contain handwritten annotations, stamps, stains and other sources of visual noise.

The first step is checking if a file content is purely an image scan or contains text data. If the former is true, we apply an Optical Character Recognition (OCR) system (Smith, 2007) and store the resulting text. Otherwise, we use regular expressions to verify the embedded text quality. In case the quality is deemed acceptable, we simply store the text; if not, we apply OCR and store the result. The extracted text contains some artifacts from the OCR system and PDF tagging scheme. For that reason, we employ regular expressions to clean the text. In addition, we apply to the text some preprocessing steps: stemming, removal of stop words, lower-casing, tokenization of e-mails and URLs, and specific tokenization of articles of law (e.g. Lei—law— 11.419 to LEI_11419).

The dataset contains two types of annotation for two different tasks.

1. Labels for document type classification: *Acórdão*, for lower court decisions under review; *Recurso Extraordinário* (RE), for appeal petitions; *Agravo de Recurso Extraordinário* (ARE), for motions against the appeal petition; *Despacho*, for court orders; *Sentença* for judgements; and *Others* for documents not included in the previous classes. This task has evolved from early versions evaluated in (Braz et al., 2018; da Silva et al., 2018).

2. Labels for lawsuit theme classification, which assign one or more General Repercussion (*Repercussão Geral*) themes to each Extraordinary Appeal. There are 28 theme options identified by integers (e.g. theme 810) corresponding to the most frequent ones and one class (with ID 0) for the remaining themes, summing up to 29 classes.

To ensure the reproducibility of our experiments we randomly divided the appeals into 70%/15%/15% splits for train/validation/test respectively, maintaining theme distribution across them.

There are three versions of VICTOR:

- Big VICTOR or BVic, used only for theme classifications, since it contains all data, including the unlabeled documents.

- Medium VICTOR or MVic (44,855 suits, 628,820 documents and 2,086,899 pages) is the result of filtering out those samples and can be employed for both theme and document type classification.

- Small VICTOR or SVic. Due to the huge size of the MVic dataset it is extremely hard to share it with the community. So we limit the number of suits for each theme to 100 samples in each set to create the SVic dataset, which contains 6,510 Extraordinary Appeals, 94,267 documents and 339,478 pages.

Table1 exhibits the class distribution for each split of the relevant versions of the dataset. Figures 1, 2 and 3 show the theme distribution for each versions of VICTOR. The presented theme IDs are the ones used originally by the Court.

Table 1: Class distribution per split.

Dataset	Category	Training set		Validation set		Test set	
		Documents	Pages	Documents	Pages	Documents	Pages
MVic	Acórdão	1,966	4,740	354	656	358	659
	ARE	2,894	34,640	760	8,373	721	7,347
	Despacho	2,415	3,952	326	457	346	490
	Others	420,494	1,323,841	92,696	280,399	93,855	283,763
	RE	4,396	77,893	902	15,753	849	15,129
	Sentença	4,065	21,210	727	3,970	696	3,627
SVic	Acórdão	301	553	201	299	199	273
	ARE	270	2,546	237	2,149	213	1,841
	Despacho	265	346	147	183	147	198
	Others	38,585	134,134	25,898	84,104	25,744	85,408
	RE	453	9,509	326	6,364	312	6,331
	Sentença	420	2,129	284	1,636	265	1,475

Table 2: F1 score of our methods for document type classification on the test sets. A baseline that always chooses the majority class yields a F1 score weighted by class frequencies of 87.06/84.41 and a average F1 score of 15.90/15.73 on MVic and SVic, respectively.

Dataset	Model	Acórdão	ARE	Despacho	Others	RE	Sentença	Weighted	Average
MVic	NB	49.20	32.08	39.82	89.38	38.06	37.80	84.77	47.72
	SVM	65.41	52.62	59.34	95.85	64.52	69.75	92.88	67.92
	BiLSTM	**72.84**	57.82	**60.07**	97.11	67.74	69.96	94.33	**70.92**
	CNN	71.06	**58.11**	56.04	**97.37**	**68.71**	**72.35**	**94.64**	70.61
SVic	NB	66.40	36.07	51.15	93.24	55.89	55.99	88.93	59.79
	SVM	81.15	**58.06**	**67.88**	96.85	74.66	**79.30**	94.25	**76.32**
	BiLSTM	85.82	52.12	51.01	97.15	74.06	76.70	94.65	72.81
	CNN	**86.43**	55.92	59.88	**97.30**	**76.23**	79.29	**94.72**	75.84

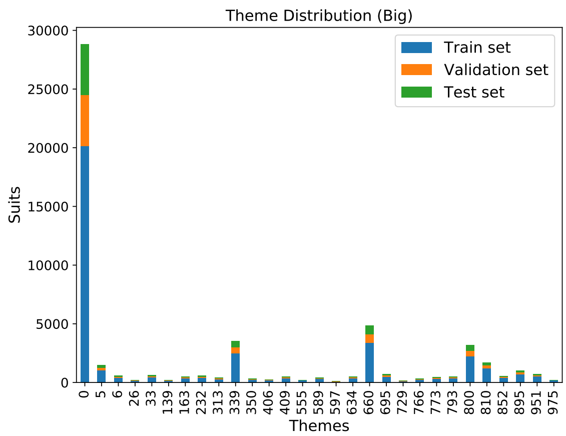


Figure 1: BVic theme distribution.

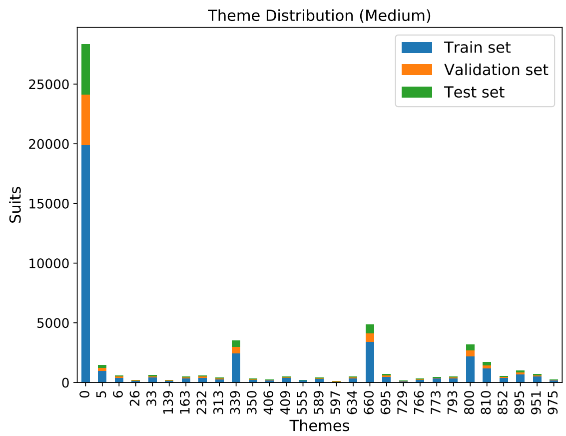


Figure 2: MVic theme distribution.

4. Document Type Classification

In this section we compare different methods we explored to classify the document types. All results, unless stated otherwise, are reported on the test set and refer to page prediction accuracy. For a baseline, we select the most frequent class (*others*), which gives a F1 score weighted by class frequencies of 87.06/84.41 and a average F1 score of 15.90/15.73 on M/SVic test set.

4.1. Bag-of-words Methods

We represent the documents as bag-of-words with tf-idf features. We experiment with two different classifiers: Naïve Bayes and SVM.

Feature extraction: We search for the best hyperparameters using the validation set. The best approach uses unigrams and bigrams, and includes only terms with a minimum document frequency of two pages and a maximum frequency of 50% of the pages. We restrict our vocabulary to the 70,000 most frequent words in the training set.

Naïve Bayes: We train a Naïve Bayes classifier with a additive Laplace smoothing parameter $\alpha = 0.001$ and class prior fitting due to the category imbalance.

SVM: We employ a SVM with linear kernel and apply weights inversely proportional to class frequencies to com-

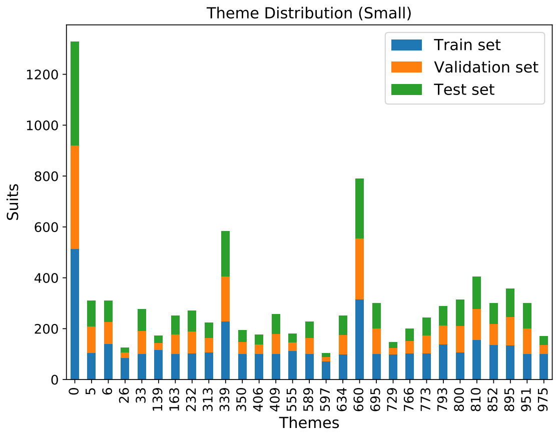


Figure 3: SVic theme distribution.

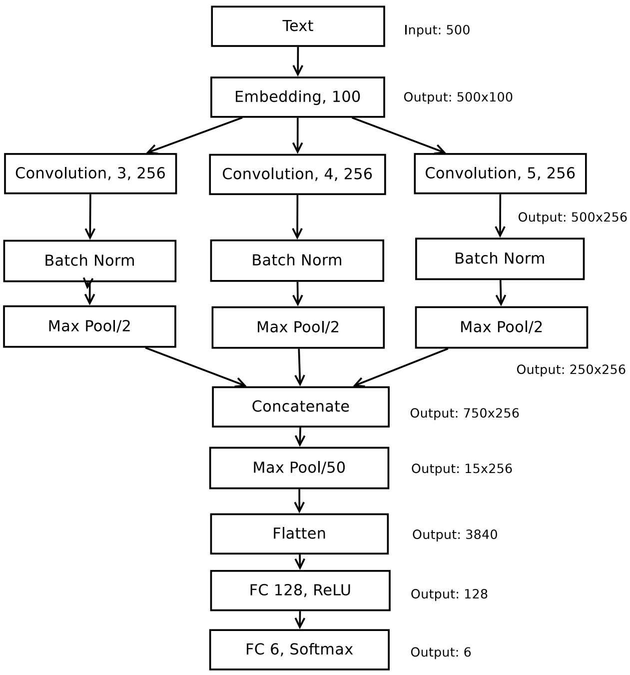


Figure 4: CNN architecture for document type classification.

pensate the imbalance.

4.2. Convolutional Neural Network

We based our CNN architecture on the one proposed in (Conneau et al., 2017). Our network is shallower though, as we found that stripping several layers improved the accuracy of the model. As a result, the network trains faster and requires less GPU memory. We also work on the word level instead of on the character level.

Our architecture is shown in Figure 4. The network takes as input the first 500 tokens from the input and embed them into 100 dimensional vectors. The remaining tokens are discarded, with the intuition that those first tokens are sufficient to discriminate between classes, which was confirmed in early experiments. Next, we concatenate the output of three convolutional blocks formed by a convolutional layer with 256 filters and varied sizes (3, 4 and 5) followed by

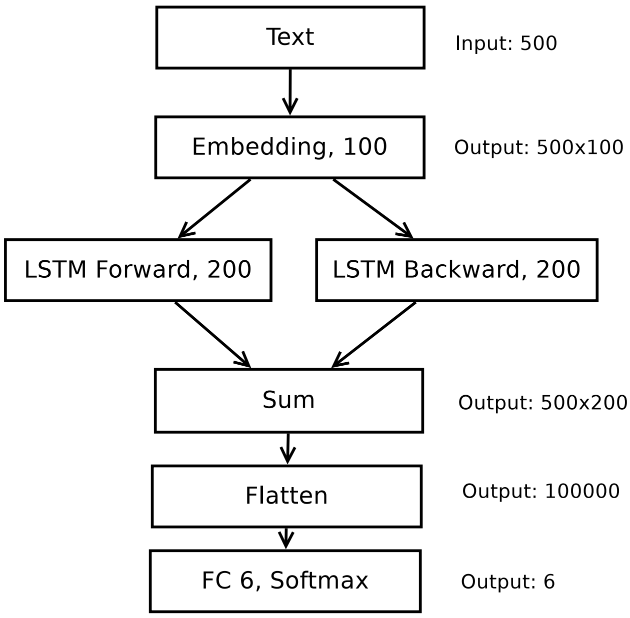


Figure 5: Bi-LSTM architecture for document type classification.

Table 3: Performance before and after CRF processing on the test sets.

	MVic		SVic	
Classes	CNN	CNN-CRF	CNN	CNN-CRF
Acórd.	71.06	75.02 / +5.57%	86.43	90.60 / +4.82%
ARE	58.11	62.89 / +8.23%	55.92	59.54 / +6.47%
Desp.	56.04	62.55 / +11.62%	59.88	56.69 / -5.33%
Others	97.37	97.66 / +0.30%	97.30	97.68 / +0.39%
RE	68.71	74.38 / +8.25%	76.23	78.77 / +3.33%
Sent.	72.35	77.77 / +7.49%	79.29	81.13 / +2.32%
Wtd.	94.64	95.37 / +0.77%	94.72	95.33 / +0.64%
Avg.	70.61	75.05 / +6.29%	75.84	77.40 / +2.06%

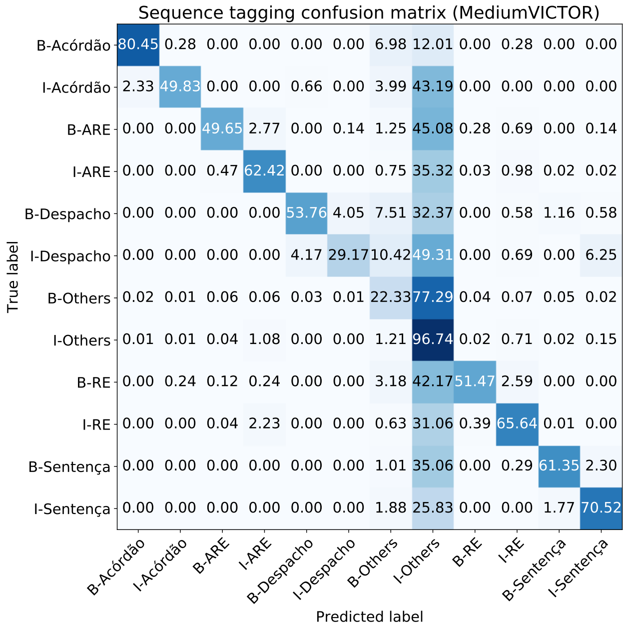
batch normalization and max pooling layer of size 2. Another max pooling operation (of size 50) is applied to the result of the concatenation and the output is flattened. Finally, the flattened tensor is processed by two fully connected layers and a softmax function produces the final output. A dropout mask is applied to the first fully connected layer with 50% dropping probability.

We use Adam (Kingma and Ba, 2015) to optimize the cross-entropy loss function with a learning rate of 0.001 and train the model for 20 epochs with mini-batches of 64 samples.

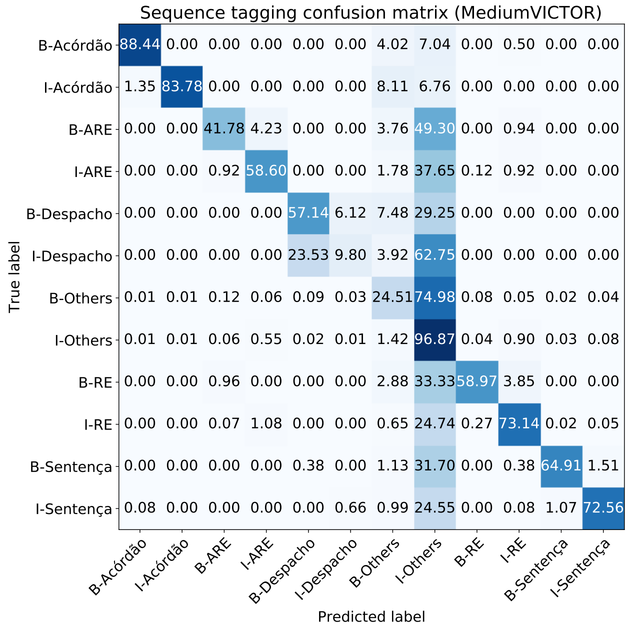
4.3. Bidirectional LSTM Network

For this model, we embed the first 500 tokens from each page into an 100 dimensional space and subsequently feed them into a Bidirectional (Graves and Schmidhuber, 2005) Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) layer with 200 units for each direction. The forward and backward representations of the sequence are summed together and fed to a fully connected layer followed by a softmax activation that calculates the final class probabilities. Figure 5 exhibits the architecture.

We trained the model for 20 epochs with batches of 64 sam-

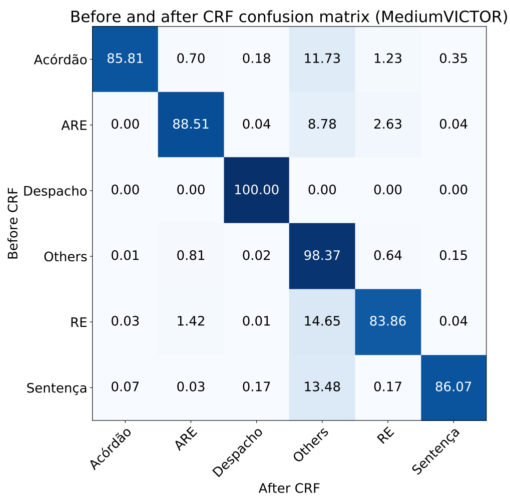


(a) MVic.

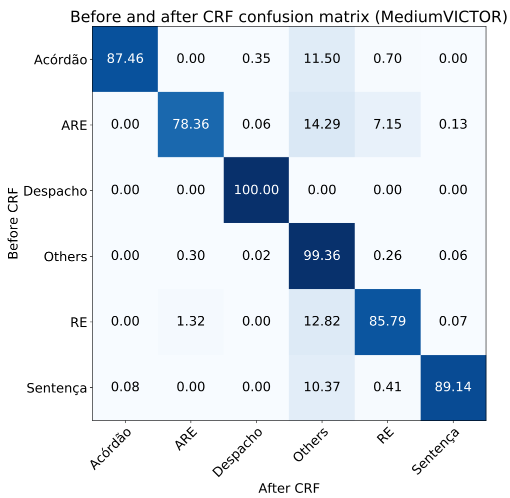


(b) SVic.

Figure 6: Confusion matrix of CRF predictions for the test set and ground truth tags. Each value represents the percentage of samples from the row class that were classified as being from the column class.



(a) MVic.



(b) SVic.

Figure 7: Confusion matrix of test set predictions before and after CRF processing. Each value represents the percentage of samples with the row class prediction before CRF processing that were classified as being from the column class after CRF processing.

ples and learning rate value of 0.001 with Adam optimizer.

4.4. Linear-chain CRF post-processing

Instead of classifying each page by itself, one can use the fact that a suit is composed by a series of document pages and treat the document classification as a sequence labeling problem. Intuitively, a page is more likely to be followed by another of the same type, as documents usually contain more than one page, so taking in consideration the sequential aspect of the data should improve classification metrics. Rather than having a page as input and outputting a document type prediction, the sequence labeling approach outputs a series of type predictions (tags) given a series of input pages. We can consider neighbor tag information by employing linear-chain Conditional Random Fields (CRF), which have been shown to be very effective in sequence tagging problems (Lafferty et al., 2001; Huang et al., 2015; Lample et al., 2016).

To better leverage the sequential information, we adapt the document classes by using the IOB tagging scheme (Ramshaw and Marcus, 1999). We prepend “B-” to the ground truth of first pages of document or “I-” in the

Table 4: F1 score of our methods for theme classification on the test sets. A baseline that always assigns all themes yields a F1 score weighted by class frequencies of 37.47 /37.10/4.31 and an average F1 score of 2.41/2.40/0.94 on BVic, MVic, SVic, respectively.

	BVic			MVic			SVic		
Themes	NB	SVM	XGBoost	NB	SVM	XGBoost	NB	SVM	XGBoost
0	81.63	87.35	**90.70**	79.50	88.85	**92.41**	49.90	**72.29**	69.71
5	17.95	92.47	**94.15**	18.73	79.05	**85.50**	30.22	**84.79**	82.87
6	65.85	61.65	**77.84**	37.45	36.52	**76.81**	21.93	63.11	**77.03**
26	60.38	92.06	**93.33**	14.59	36.48	**94.74**	12.75	**97.44**	94.44
33	30.03	46.32	**77.17**	8.35	14.42	**78.62**	30.71	57.78	**74.65**
139	61.82	81.25	**90.57**	17.54	74.67	**92.59**	14.95	88.89	**94.34**
163	77.38	75.41	**86.09**	25.05	76.19	**88.00**	73.86	86.08	**94.67**
232	40.93	44.64	**69.33**	27.63	13.90	**55.12**	37.32	65.00	**65.08**
313	47.42	58.56	**72.55**	31.11	43.37	**80.77**	60.22	76.12	**82.69**
339	23.17	52.12	**74.47**	20.62	45.84	**77.04**	26.73	74.38	**86.06**
350	73.27	55.26	**86.96**	73.27	12.05	**89.58**	85.06	52.94	**90.11**
406	57.41	44.44	**85.71**	20.27	10.41	**85.71**	55.81	46.15	**84.93**
409	74.42	79.12	**86.25**	29.03	72.64	**90.68**	91.14	90.91	**95.48**
555	39.02	65.06	**83.33**	0.00	17.06	**84.75**	47.06	52.46	**88.89**
589	77.97	82.01	**88.00**	35.02	63.44	**88.71**	82.05	90.16	**90.76**
597	**96.77**	90.91	96.55	53.57	90.91	**96.55**	85.71	88.24	**96.77**
634	89.87	90.91	**95.48**	70.24	89.29	**94.19**	92.81	93.08	**95.42**
660	51.23	74.14	**89.00**	35.30	80.39	**90.07**	36.41	91.10	**93.51**
695	93.27	**97.65**	96.65	95.37	**98.13**	96.68	96.52	**98.49**	96.94
729	**100.00**	**100.00**	97.78	62.07	**95.65**	93.02	63.16	**100.00**	93.33
766	21.88	73.21	**77.65**	21.82	76.64	**82.61**	19.81	81.08	**86.67**
773	68.03	96.40	**97.06**	61.54	95.71	**98.55**	81.30	**94.03**	93.13
793	66.67	84.52	**92.96**	28.26	86.23	**91.43**	26.59	87.80	**90.79**
800	87.70	98.42	**98.73**	87.34	98.41	**98.62**	69.86	**92.71**	91.10
810	62.28	88.72	**95.32**	23.89	92.16	**94.87**	21.06	**95.62**	94.69
852	64.67	82.61	**87.34**	54.40	76.68	**89.74**	49.08	89.41	**92.31**
895	25.10	63.68	**89.66**	14.64	94.08	**98.32**	24.07	92.17	**95.93**
951	94.74	**100.00**	99.54	39.04	98.21	**98.62**	57.36	**99.50**	95.29
975	86.15	91.67	**94.44**	15.62	68.69	**91.43**	41.61	**89.74**	**89.74**
Weighted	69.55	82.35	**89.57**	60.62	81.37	**90.72**	48.75	82.31	**86.34**
Average	63.35	77.61	**88.43**	37.97	66.42	**88.82**	51.21	82.46	**88.87**

other cases (e.g. if a suit begins with a RE of three pages followed by an ARE of equal length, the sequence of labels would start with B-RE, I-RE, I-RE, B-ARE, I-ARE, I-ARE). The training instances are the dataset suits, which are sequences of pages. We pre-calculate a six-dimensional embedding for each page by feeding it to our best performing model, the CNN, and saving the output of the softmax. The sequences of page embeddings are then used to train a CRF model.

We employ said procedure in both MVic and SVic. The following section compares the performance of the CNN model before and after the CRF processing for each test set.

4.5. Results and Discussion

Table 2 compares test performance across the evaluated models.

The CNN and the BiLSTM trained and evaluated on MVic outperform the other models in all categories; the SVM followed close behind, while the Naïve Bayes classifier achieved much lower scores. Furthermore, all models are able to beat the baselines for weighted and average F1 score, with the exception of the Naïve Bayes, whose weighted F1 score is 2.63% lower, though the average F1 score is much higher than the baseline. The CNN result represents a relative increase of 8.71% and 344.00%, respectively, for each metric. We can see that, due to the imbalanced nature of the data, the average F1 is a more informative metric of the performance of the model.

Regarding the SVic dataset, the SVM and the CNN were the best-performing models. Similarly to the MVic scenario, all models beat the baseline, with the CNN representing a relative increase of 12.22% and 381.99% for the weighted and average F1 score, respectively. These results suggest that the SVM is able to better generalize the much smaller dataset.

In both scenarios and across all explored models, the category *Others* has the best F1 score. This is not surprising, since it includes the vast majority of pages in the datasets. That being said, our strategies for dealing with data imbalance where effective—without fitting the class prior (NB) or using class weights (SVM) the classifiers behaved approximately as the baseline, predicting almost every sample as belonging to the *Others* class.

Table 3 shows the impact of CRF modeling. Our sequence modeling approach, albeit simple, results in overall improvements in both versions of dataset. The best increase in performance was regarding *Despacho* classification on

Table 5: F1 score of a XGBoost trained without and with *Others* pages on BVic test set filtered to include only lawsuits with at least one page not classified as *Others*.

Themes	Without	With	Count
0	91.15	**92.55**	832
5	**93.33**	85.71	8
6	70.00	**81.82**	13
33	**0.00**	**0.00**	3
139	**50.00**	0.00	2
163	90.65	**91.43**	67
232	69.77	**80.00**	23
313	**77.78**	70.00	11
339	49.32	**70.89**	48
350	**100.00**	**100.00**	1
406	**0.00**	**0.00**	4
409	87.58	**89.93**	71
555	54.55	**83.33**	7
589	86.96	**92.63**	47
597	90.91	**90.91**	6
634	**95.83**	90.57	25
660	33.80	**86.05**	49
695	89.29	**92.86**	29
729	**100.00**	96.97	17
766	57.14	**66.67**	10
773	**94.55**	**94.55**	29
793	**0.00**	**0.00**	4
800	80.40	**97.78**	115
810	76.19	**87.50**	44
852	82.05	**92.68**	19
895	0.00	**100.00**	2
Weighted	84.55	**90.27**	1,486
Average	66.20	**74.42**	

MVic—a relative improvement of 11.62%. On the other hand, SVic *Despacho* saw a relative decrease of 5.33%. The MVic model had the greatest positive changes, perhaps due to the fact that the MVic CNN model had more room for growth than its small counterpart and more training data.

Figure 6 exhibits the confusion matrices of CRF tag predictions. The greatest source of confusion is the I-Others tag (pages classified as others that are not the first page of a document), which is not surprising due to its overabundance. We have a similar scenario when we analyze the confusion between predictions before and after CRF processing (Figure 7): the CRF is more likely to tag a page as *Others* when compared to the original model.

One possible way to improve the sequence tagging approach is leveraging the sequential information in the document embedding level, that is, using an end-to-end approach where we jointly train the CRF layer and the feature extractor. Furthermore, our technique employs a vector of 6 dimensions that, while sufficient for our viability assessment needs, cannot sufficiently encode relevant document attributes. Higher dimensional embeddings should improve the task accuracy.

5. Lawsuit Theme Classification

5.1. Bag-of-words Methods

For the task of lawsuit theme classification we represent each document as a vector of tf-idf features. This approach is better suited than using CNNs or RNNs due to the great size of the samples, where dozens of pages are not uncommon, which leads to vanishing gradients and excessive memory needs. Besides the classifiers we mentioned in the previous section, we also train an Extreme Gradient Boosting (XGBoost) (Chen and Guestrin, 2016) classifier. XGBoost is an optimized tree boosting system that has become very popular amongst Kaggle competitions for various ML tasks.

Since theme classification is a multilabel and multiclass problem we employ an one-vs.-rest approach where we train one classifier for each class and set a threshold value for assigning a theme to a document. That is, given C the set of all possible classes, t the threshold value, $f_c(\cdot)$ the classifier function for class c, and a document d:

$$\forall c \in C,\ \text{we assign } c \text{ to } d \text{ if } f_c(d) \geq t\,. \tag{1}$$

We use 0.5 as the threshold value. All the following reported metrics are on the test set. As a baseline result we choose to assign to each document only the most frequent theme, which gives us a F1 score weighted by class frequencies of 37.47 /37.10/4.31 and an average F1 score of 2.41/2.40/0.94 on B/M/SVic test set.

Feature extraction: The best performing configuration on the validation set uses only unigrams with a minimum document frequency of 10%. We also limit the vocabulary to the 10,000 most frequent words.

Naïve Bayes and SVM: We employ the same hyperparameters discussed in Section 4..

XGBoost: We train 500 trees with a maximum depth of 4 and a shrinkage factor of 0.1.

5.2. Theme Classification with Domain Knowledge

One intuition legal experts have is that the most informative pages about a suit's themes are the ones not classified as *Others*. On that premise, one possible improvement for theme classification models is to take into consideration only the suit's pages that do not have an *Others* label.

On the other hand, at test time we do not have ground truth knowledge about page type classification. Thus, such method can propagate errors from the document type classification model, which may negatively impact accuracy. To test the feasibility of the idea, we train and test a XGBoost model only with the relevant pages of BVic to establish a upper-bound of performance. When we eliminate all pages labeled as *Others* we lose the suits that contain no other kinds of pages. To establish a fair comparison to a method that uses no domain knowledge, we also train a model on the same suits without removing pages labeled as *others*. We show the results in the following section.

5.3. Results and Discussion

Table 4 exhibits the models' performance in each VICTOR version. All models are able to beat the baselines for both weighted and average F1 score. The XGBoost outperforms the other models across all versions of VICTOR, excluding a few themes better assigned by the SVM, and, on two occasions, the Naïve Bayes. Furthermore, the SVM overall results were fairly consistent through the different datasets in comparison with the Naïve Bayes and the XGBoost.

The data imbalance impact of the results here is far less pronounced than in the previous task. XGBoost, the best classifier, has very similar weighted and average F1 scores in all versions of VICTOR, even though the theme distribution is heavily skewed towards class 0. In addition, the model outperforms the B/M/SVic baselines by 139.02%/144.55%/$1,905.49\%$ (F1 score weighted by class frequency) and $3,571.92\%$/$3,602.87\%$/$9,350.90\%$ (Average F1 score). These results show that TFIDF values are good features when classifying huge documents.

Table 5 compares models trained with and without pages labeled as *Others*, thought to be less informative by the Court experts. The classes' F1 scores show great variability, with numbers ranging from 0 to 100 in both cases. That is not surprising, considering the number of examples for the themes with extreme scores, which is between 0 and 4. Due to the small number of samples, such scores are not very reliable.

That being said, the overall results oppose the domain expert intuition, since the weighted and average F1 scores for the model trained with *Others* pages were 6.77% and 12.42% higher, respectively, than the model trained without such pages. That is, contrary to domain knowledge expectations, the data are useful for the task and should not be disregarded.

6. Conclusion

We introduce the VICTOR Dataset, a corpus of legal documents from Brazil's Supreme Court. VICTOR features two types of tasks: document type classification, with six disjoint document categories; and theme assignment, a multilabel problem with 29 different tags. The dataset is made available in three versions: BVic, containing data for the theme assignment task; MVic, containing only type-labeled documents, for both tasks; and SVic, a subsample of MVic. We also establish benchmarks for the presented tasks, comparing textual and sequential data representations. Our experiments with CRF post-processing show that the sequential nature of the suits may be leveraged to improve document type classification. Furthermore,we find that tf-idf features are good descriptors of long texts, where common deep learning approaches are not easily applicable. Finally, we hope our data and benchmarks encourage further exploration of better-performing models and techniques.

7. Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. TdC received support from Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), grant PQ 314154/2018-3. We acknowledge the support of "Projeto de Pesquisa & Desenvolvimento de aprendizado de máquina (machine learning) sobre dados judiciais das repercussões gerais do Supremo Tribunal Federal - STF". We are also grateful for the support from Fundação de Apoio À Pesquisa do Distrito Federal (FAPDF) and Fundação de Empreendimentos Científicos e Tecnológicos (Finatec).

8. Bibliographical References

Aletras, N., Tsarapatsanis, D., Preotiuc-Pietro, D., and Lampos, V. (2016). Predicting judicial decisions of the european court of human rights: A natural language processing perspective. *PeerJ in Computer Science*, 10.

Braz, F. A., da Silva, N. C., de Campos, T. E., Chaves, F. B. S., Ferreira, M. H. S., Inazawa, P. H., Coelho, V. H. D., Sukiennik, B. P., de Almeida, A. P. G. S., Vidal, F. B., Bezerra, D. A., Gusmao, D. B., Ziegler, G. G., Fernandes, R. V. C., Zumblick, R., and Peixoto, F. H. (2018). Document classification using a bi-lstm to unclog brazil's supreme court. In *NeurIPS workshop on Machine Learning for the Developing World (ML4D)*, December 8. Event webpage: `https://sites.google.com/view/ml4d-nips-2018/`. Published at arXiv:1811.11569.

Cardellino, C., Teruel, M., Alonso Alemany, L., and Villata, S. (2017). A low-cost, high-coverage legal named entity recognizer, classifier and linker. In *Proceedints of the 16th International Conference on Artificial Intelligence and Law (ICAIL)*, London, United Kingdom, June. Preprint available from `https://hal.archives-ouvertes.fr/hal-01541446`.

Carter, D. J., Brown, J., and Rahmani, A. (2016). Reading the high court at a distance: Topic modelling the legal subject matter and judicial activity of the high court of australia, 1903-2015. *UNSWLJ*, 39:1300.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *CoRR*, abs/1603.02754.

Conneau, A., Schwenk, H., Barrault, L., and Lecun, Y. (2017). Very deep convolutional networks for text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1107–1116, Valencia, Spain, April. Association for Computational Linguistics.

da Silva, N. C., Braz, F. A., de Campos, T. E., Gusmao, D., Chaves, F., Mendes, D., Bezerra, D., Ziegler, G., Horinouchi, L., Ferreira, M., Carvalho, G., Fernandes, R. V. C., Peixoto, F. H., Filho, M. S. M., Sukiennik, B. P., Rosa, L. S., Silva, R. Z. M., and Junquilho, T. A. (2018). Document type classification for brazil's supreme court using a convolutional neural network. In *10th International Conference on Forensic Computer Science and Cyber Law (ICoFCS)*, Sao Paulo, Brazil, October 29-30. Winner of the best paper award.

de Cássia Carvalho Lopes, R. (2017). Eventual influences of common law on the Brazilian legal sysbrazilian legal system, Mars. [Online; posted 15-Mars-2017. `https://www.migalhas.com/HotTopics/63,MI255372,51045-Eventual+Influences+of+Common+Law+on+the+Brazilian+Legal+System`].

Dozier, C., Kondadadi, R., Light, M., Vachher, A., Veeramachaneni, S., and Wudali, R. (2010). Named entity recognition and resolution in legal text. In *Semantic Processing of Legal Texts*, pages 27–43. Springer.

Fariello, L. (2018). CNJ apresenta justiça em números 2018, com dados dos 90 tribunais, August. [Online; posted 27-August-2018. `http://www.cnj.jus.br/noticias/cnj/87512-cnj-apresenta-justica-em-numeros-2018-com-dados-dos-90-tribunais`].

Galgani, F., Compton, P., and Hoffmann, A. (2012). Combining different summarization techniques for legal text. In *Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data*, HYBRID, pages 115–123, Stroudsburg, PA, USA. Association for Computational Linguistics.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia, July. Association for Computational Linguistics.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In Claire Nédellec et al., editors, *Machine Learning: ECML*, pages 137–142, Berlin, Heidelberg. Springer Berlin Heidelberg.

Johnson, R. and Zhang, T. (2016). Supervised and semi-supervised text categorization using lstm for region embeddings. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML, pages 526–534. JMLR.org.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2017). Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431. Association for Computational Linguistics.

Kanapala, A., Pal, S., and Pamula, R. (2017). Text summarization from legal documents: a survey. *Artificial Intelligence Review*, Jun.

Katz, D. M., Bommarito, Michael J, I., and Blackman, J. (2014). Predicting the Behavior of the Supreme Court of the United States: A General Approach. *arXiv e-prints*, page arXiv:1407.6333, Jul.

Kim, M.-Y., Xu, Y., and Goebel, R. (2013). Summarization of legal texts with high cohesion and automatic compression rate. In *New frontiers in artificial intelligence*. Springer.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*. Preprint available at `https://arxiv.org/abs/1412.6980`.

Kumar, R. and Raghuveer, K. (2012). Legal document summarization using latent dirichlet allocation. *International Journal of Computer Science and Telecommunications*, 3:114–117.

Lafferty, J. D., andrew, M., 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*, ICML, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California, June. Association for Computational Linguistics.

Liu, C., Hsaio, W., Lee, C., Chang, T., and Kuo, T. (2016). Semi-supervised text classification with universum learning. *IEEE Transactions on Cybernetics*, 46(2):462–473, Feb.

Luz de Araujo, P. H., de Campos, T. E., de Oliveira, R. R. R., Stauffer, M., Couto, S., and Bermejo, P. (2018). LeNER-Br: a dataset for named entity recognition in Brazilian legal text. In *International Conference on the Computational Processing of Portuguese (PROPOR)*, Canela, RS, Brazil, September 24-26.

O'Neill, J., Robin, C., O'Brien, L., and Buitelaar, P. (2016). An analysis of topic modelling for legislative texts. In *Proceedings of the Second Workshop on Automated Semantic Analysis of Information in Legal Texts*, June.

Ramshaw, L. A. and Marcus, M. P. (1999). Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer. Preprint available at `http://arxiv.org/abs/cmp-lg/9505040`.

Remmits, Y. (2017). Finding the Topics of Case Law: Latent Dirichlet Allocation on Supreme Court Decisions. Bachelor's thesis, Radboud University, July 2017.

Secretaria de Comunicação Social do Conselho Nacional de Justiça. (2018). Sumário executivo do relatório justiça em números 2018. `http://www.cnj.jus.br/files/conteudo/arquivo/2018/09/da64a36ddee693ddf735b9ec03319e84.pdf`.

Smith, R. (2007). An overview of the Tesseract OCR engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR)*, volume 2, pages 629–633. IEEE.

Şulea, O.-M., Zampieri, M., Vela, M., and van Genabith, J. (2017). Predicting the law area and decisions of French Supreme Court cases. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP*, pages 716–722. INCOMA Ltd.

Undavia, S., Meyers, A., and Ortega, J. E. (2018). A comparative study of classifying legal documents with neural networks. In *Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 515–522, Sep.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS, pages 649–657, Cambridge, MA, USA. MIT Press.