NOVA IMS Information Management School

MAA

Mestrado em Métodos Analíticos Avançados
Master Program in Advanced Analytics

METADATA AND TEXTUAL FEATURES TO ADVISE ADMINISTRATIVE COURTS DECISIONS

Predicting judgments of violation processes and identifying similar cases

Hugo Saisse Mentzingen da Silva

Project Work presented as partial requirement for obtaining the Master's degree in Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

METADATA AND TEXTUAL FEATURES TO ADVISE ADMINISTRATIVE COURTS DECISIONS

by

Hugo Saisse Mentzingen da Silva

Project Work presented as a partial requirement for obtaining the Master's degree in Advanced Analytics

Advisor: Nuno Miguel da Conceição António

October 2021

DEDICATION

This project is for Brazilian citizens, hardworking and resilient people who financed this endeavor and deserve all the effort to seek swifter administrative decisions. To the Portuguese people, who kindly made theirs my home for two memorable years.

This project is also for my beloved wife, who put her career on hold and came on board to pursue this master's degree. To my son, source of energy and inspiration. It is also to my parents, who transmitted the restlessness and taste for the challenge necessary to complete this journey. To my godparents, who were always around, providing encouragement, counseling, and care.

ACKNOWLEDGEMENTS

I want to thank the Superintendency of Private Insurance (SUSEP) and ex-Director Paulo dos Santos for allowing me to expand my knowledge through this master's course. I also thank SUSEP for making available the data to this project.

I extend my thanks to Professor Nuno António for his insightful and attentive supervision, also for having challenged me to bring the work to a higher level.

I would also like to acknowledge Igor Oliveira, Livia Bomfim, and Samira Branco, my colleagues from SUSEP, for their outstanding collaboration in analyzing violations similarity and providing the gold standard for one of the studies included in this project. I also acknowledge Cristiano Cesario for diligently curating the data necessary to this project.

To the colleagues and professors of Nova Information Management School, I would like to thank the environment of collaboration and exchange of personal, professional, and academic experiences.

ABSTRACT

Many aspects of citizens' lives are affected by the decision-making process carried out by government organizations, such as administrative justice bodies. However, these bodies have limited resources and struggle to keep up with the increasing cases demand. This work project is composed of two studies to assist administrative courts. In the first one, a two-stage cascade classifier predictive model was proposed for advising administrative decisions. The model employs a first-stage prediction from textual features and a second-stage classifier that includes 'proceedings' metadata. The study was conducted using time-based cross-validation, being entirely built on data available before the predicted judgment. It only considers the first document in each proceeding, along with the metadata recorded when the violation is first registered.

Consequently, the model endows generality and out-of-sample serviceability. Also, by preserving visibility on the textual features and employing the Shapley Additive Explainability (SHAP), the proposed model provides local explainability. Results show that this is an advantageous procedure when both text and metadata are available. With a weighted F1Score of 0.900, the results outperform the text-only baseline by 1.24% and the metadata-only baseline by 5.63%, with better discriminative properties, evaluated by the Receiver Operating Characteristic (ROC) and Precision-Recall curves.

The second study evaluated different combinations of document representations, embeddings, and similarity measures to assess whether these assemblies can identify similar pairs of cases that satisfy the human notion of similarity. A representative dataset of administrative violations was employed, and the models' outputs were compared to an experts' gold standard and a baseline model.

Adopting the mean Average Recall (mAR) as the primary evaluation metric, the Word2Vec models outperformed the baseline, and together with the TF-IDF models, obtained the best Recall scores overall. Stemming showed to be an essential pre-processing technique to influence the results. Findings also suggest that the choice of the model (Word2Vec) prevails over the setup, i.e., the corpus, the number of dimensions, and the implementation algorithm. Additionally, the performance obtained using sentences as text representation, or BM25 and Doc2Vec as models, showed to be noticeably low.

The encouraging results from the studies confirm that both AI-based legal assistance techniques can be instrumental in helping administrative justice bodies improve their decision-making process, both in terms of speed and consistency of decisions.

KEYWORDS

Administrative decision prediction, document similarity, cascade generalization, legal assistance, machine learning, natural language processing

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AI	Artificial Intelligence
AILA	Artificial Intelligence for Legal Assistance
AML	Anti-Money Laundering
AR	Average Recall
AUC	Area under the ROC Curve
BERT	Bidirectional Encoder Representations from Transformers
BRL	Brazilian real
CBOW	Continuous Bag of Words
CNN	Convolutional Neural Network
DFR	divergence from randomness
ECHR	European Court of Human Rights
FIRE	Forum for Information Retrieval Evaluation
FPR	False Positive Rate
HTML	Hypertext Markup Language
IAIS	International Association of Insurance Supervisors
ICP	Insurance Core Principles
IRS	Information Retrieval Systems
KYC	Know Your Customer
LDA	Latent Dirichlet Allocation
LM	language model
mAR	mean Average Recall
ML	Machine Learning
NILC-USP	Núcleo Interisnstitucional para Linguística Computacional – Universidade de São Paulo (Interinstitutional Center for Computational Linguistics – São Paulo University)
NLP	Natural Language Processing
NLTK	Natural Language Toolkit

PDF	Portable Document Format
PL	paragraph-link
POS	part of speech
PPV	Positive Predictive Value
PRF	Probabilistic Relevance Framework
ROC	Receiver Operating Characteristic
RSLP	Removedor de Sufixos da Lingua Portuguesa (Portuguese Language Suffix Remover)
SCOTUS	Supreme Court of the United States
SG	Skip-Gram
SGD	Stochastic Gradient Descent
SMOTE	Synthetic Minority Oversampling Technique
SUSEP	Superintendency of Private Insurance
SVC	Support Vector Classifier
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TF-IDF	Term Frequency – Inverse Document Frequency
TPR	True Positive Rate
USD	United States dollar

1. INTRODUCTION

There is a little-known but hugely important justice system that impacts everyone's life – administrative justice. It deals with more cases in many countries than criminal or private civil justice (Nason, 2018). Many aspects of citizens' lives may be affected by the decision-making process carried out by government bodies. Not only citizen-centered public services, as licensing or social benefits granting, but also the State supervision of economic agents is subject to administrative courts' scrutiny.

Within this scope, the corrective measures taken by national financial system supervisors, namely in the banking and insurance sectors, represent an essential enforcement example of administrative law. The worldwide insurance industry's gross premiums in 2018 reached USD 5.3 trillion, with an average penetration[1] of 7.23% in 2019 (Statista, 2020), reinforcing the need for regulation and supervision. The supervisory authorities' mandates include bankruptcy avoidance, contract fairness evaluation, and consumer protection.

The eleventh from the Insurance Core Principles (ICP) published by the International Association of Insurance Supervisors (IAIS) asserts that the jurisdictions must guarantee that the supervisor "enforces corrective action and, where needed, imposes sanctions based on clear and objective criteria that are publicly disclosed" (IAIS, 2017). This principle directly gives rise to a demand for administrative bodies that timely deliver justified and coherent proceedings when violations are identified. Their actions resulted in the imposition of a total of USD 36 billion of fines to financial institutions for non-compliance with Anti-Money Laundering (AML) or Know Your Customer (KYC) regulations, among other sanctions (Fenergo, 2020). However, producing coherent and timely decisions pose two problems. First, the decision-maker needs to know, in a swift manner, the past propensity for a case decided under the same circumstances. Second, what similar cases were judged in the past.

Artificial Intelligence (AI) may provide valuable tools to help administrative judges and courts to meet this demand. In this work project, the Brazilian insurance supervisor data was used to evaluate solutions for time-efficient decisions while maintaining consistency with the jurisprudence. Two complementary approaches are described in this work. At first, a Machine Learning (ML) supervised model composed of cascading classifiers was used to predict violations proceedings decisions, taking generalization as a premise so that other supervisors and administrative courts may benefit from it. Besides, a second and unsupervised model was applied to identify similar cases.

The Superintendency of Private Insurance (SUSEP) is an independent authority under the Ministry of Finance to license and supervise the Brazilian insurance brokers and companies. Its mandate covers private insurance, open private pension, capitalization, and reinsurance markets. In 2019, SUSEP oversaw 286 insurers and reinsurers, with technical provisions near BRL one trillion (SUSEP, 2020a) and 99.836 insurance brokers (SUSEP, 2020b). The agency's duties include monitoring and prosecuting violations to the rules in force, both for prudential and conduct infractions. As a conduct directive illustration, insurers are required to pay claims within 30 days, a term that only may be suspended if the claim processing requires justified additional information. SUSEP may start a sanctioning proceeding through an inspection or a complaint and enforce penalties if a violation has existed. Similarly, as an example of a prudential rule, companies under its mandate must regularly provide

[1] The ratio of total insurance premiums to gross domestic product.

financial information and maintain investments under specific guidelines, whose violation may result in penalties.

SUSEP has initiated, on average, 822 violation proceedings per year between 2016 and 2019, from which 2471 remain unanalyzed. Also, on average, it took 1113 days to decide about a proceeding initiated in this period. There is an immense opportunity to take advantage of ML on supervision efficiency and insurance customer protection in this context. Decidedly SUSEP and other governmental entities that assume administrative judges' roles can use this work's outcome to speed up the juridical analysis in their jurisdictions while still observing the precedents.

The first part of the dissertation is dedicated to predicting the violations proceedings' decisions, specifically whether a violation has existed or not, a binary classification task. The objective is to provide an advisory tool so that a decision-maker has visibility on the treatment given in the past to similar situations when dealing with a new case. In practice, the model will provide the process stakeholders (legal analysts and judges) a recommended outcome for each case, based on past decisions under similar circumstances. The visibility can be conferred through the decision trend from similar circumstances, i.e., the binary classification result for the new violation. Without neglecting the need for human judgment on each specific case, the proposed model can provide uniformity, legal certainty, and better use of jurisprudence.

The second part of the study pointed to the identification of analogous cases through textual similarity. The similarity scores resulting from different document representations and embedding methods were compared with sample pairs evaluated by legal experts from SUSEP, assumed to be the gold standard. Besides the advantages generated by the first study, identifying which specific past cases are similar to new ones can offer a helping tool for the violation processing itself. Many documents can be reused, cited, or transformed into templates, providing efficiency to decision-making. It also makes it possible to analyze the arguments and evidence considered in similar violations, reinforcing the use of jurisprudence and playing a crucial role in favor of consistency in applying laws and regulations. As an additional advantage, the study's approach, based on the violation's initial document, provides the similarity likelihood in the earliest step of the violation analysis, allowing the grouping of similar cases to be jointly decided or processed by specialized judges.

2. LITERATURE REVIEW

The literature review is thematically divided into two parts. The first part is dedicated to the research made on the administrative decisions classification task. Afterward, a revision was conducted on the studies proposing methods to identify precedents for a legal case, i.e., similar older cases.

2.1. STUDY 1 - PREDICTING ADMINISTRATIVE COURTS DECISIONS

A few recent works established practices for predicting judicial decisions, either strictly using metadata or text-based approaches. Ruger et al. (2004) proposed a statistical method for predicting the Supreme Court of the United States (SCOTUS) decisions, composed of general case characteristics, and compared its results with legal experts' beliefs. The result was a much better performance achieved by the classification trees. The authors explain it was due to the model's ability to predict the more ideologically central votes, indicating that it was more accurate in separating the classes in a binary decision task. Despite being very specific on the choice of the variables and not generalizing for SCOTUS terms different from 2002, this work brought an optimistic scenario in which metadata-based decision prediction could outperform human-based judgment.

Chen & Eagel (2017) applied the Random Forests algorithm (an ensemble of randomized decision trees) to a much larger dataset related to asylum adjudications. In this case, they used internal metadata and extraneous factors such as news and local weather. Again, it was able to show that the outcome was predictable with high accuracy from a set of know variables. On the other hand, the number of variables involved difficulted its interpretation, and incorporating external variables demanded more analysis on whether they represented a causality or mere correlation. From this perspective, the proposal had a complex replication, also because it took into account a large number of variables, both internal and external to the subject.

Random Forests' success for metadata-based prediction tasks was confirmed by Katz et al. (2017) in a paper that described the design of a new model for predicting SCOTUS decisions. The authors established consistency, out-of-sample applicability, and generality as principles to make predictive ML models useful for real-world applications. Consistency may be described, in summary, as the quality of having the lowest possible performance variance across time, case issues, and Justices. Out-of-sample applicability, in turn, is directly related to ensuring that all information required for the model is knowable before the date of the decision. The generality was addressed by comparison with Ruger et al. (2004) as the former work, even though representing an enormous contribution to the legal forecasting subject, was not appliable to other terms or different compositions of the SCOTUS. In the authors' description, a general model is a model that can learn across time with new samples. Finally, to ensure out-of-sample applicability, Katz et al. (2017) divided the dataset into yearly terms and used data before each period for training.

From the author's knowledge, Aletras et al. (2016) produced the first study on predicting the outcome of cases tried by the European Court of Human Rights (ECHR) based solely on textual content. With a bag-of-words model, the authors extracted n-gram features and topics based on n-gram similarity. Their study provided strong evidence that the factual background described in the textual data may serve as a good predictor for its result, i.e., the Natural Language Processing (NLP) approach's usefulness.

Medvedeva et al. (2020), in their turn, extended these findings by testing how well Support Vector Machines (SVM) were able to predict future cases by dividing the samples used for training and testing based on the year of the case. They concluded that forecasting decisions for future lawsuits based on the distant past negatively impacts performance.

Word and paragraph embeddings were studied as an alternative by O'Sullivan & Beel (2019) because of the n-grams approach's limitations in considering words' semantics, order, and context. They also brought considerations on the importance of using test sets for more realistic model validation. Their study also indicated that algorithms different from SVM should be tested as classifiers, especially models that can disclose how the predictions are made, which should be more helpful to Judges.

Following the need for testing different classification assemblies, Pillai & Chandran (2020) applied convolutional neural networks (CNN) to bag-of-words features derived from Indian Courts data, and Sivaranjani et al. (2021) applied hierarchical convolutional neural networks (HCNN) on a feature set extracted from Indian Supreme Court to predict outcomes of appeal cases. Regardless of claims for the feasibility and promising deep learning results for text-based prediction, their study seems to lack generality and out-of-sample applicability for not considering the time dimension when testing the findings. As in some previous works, the mentioned studies also present the results based on the implementations' Accuracy, which can skew the imbalanced datasets' results. Accuracy mixes true positives and true negatives, being prevalence-dependent, which enforces the importance of the choosing and adequate score-evaluation metric.

The increasingly frequent application of deep learning techniques and black-box models in the legal context motivated the study of Bibal et al. (2021). In the authors' words, "despite their high performance, they may not be accepted ethically or legally because of their lack of explainability." Their review paper clarified the concept of explainability in law and how the different levels of legal requirements could be interpreted and translated into ML models explainability. In their study, sets of requirements that can be asked for ML models were derived from the need for decision motivation.

To the author's knowledge, no previous works investigated the use of metadata and textual information to build a cascading classifier model for a legal decisions prediction task.

2.2. STUDY 2 - IDENTIFYING ANALOGOUS CASES

Exploring and providing accessibility to the profusion of information in any knowledge area is an arduous task to which the Information Retrieval Systems (IRS) field of study is dedicated. From the manual classification based on bibliographic schemes, a lot has evolved with the help of artificial intelligence to provide decision support and reduce textual data overload.

The search engines are the most notable use case in which a query is confronted with the IRS internal indexing procedure, matching a set of previously analyzed documents. An extension task of the cited search engines, where the queries are in a range of few words, tries to identify semantically similar documents in corpora from different domains, e.g., scientific research, medical reports, and legal texts.

The legal field, to which this study is related, brings evident particularities to this unsupervised learning problem like the complex semantics, the amount of domain-specific words, and, in most situations, cross-references and previous case citations. Additionally, the case of SUSEP posed a large variety of different violation types as a challenge. From the author's research, computational linguistics to

analyze document similarity for legal texts is about ten years old. The experiments are commonly assembled by varying two main components: document representation and a similarity measurement technique. The latter is usually a combination between a text vectorization model and a vector distance metric.

Kumar et al. (2011) used judgments from the Supreme Court of India to evaluate the success of four different models combining text representations and similarity measures: cosine similarity of all terms, cosine similarity of legal terms identified in the verdicts, a similarity score derived from the number of equal citations in a pair of documents, and a similarity score represented by the number of documents in which the analyzed pair has been cited together. The authors compared the approaches' results with similarity scores given by five experts to 20 pairs of judgments and concluded that the pairs with the highest experts' scores were best matched by the second and the third models. In their words, "since a legal concept can be explained using various combinations of words, text-based similarity methods fail to satisfy the human notion of similarity." Regarding the third model, the authors concluded that "if two judgments cite the same judgments, both agree to the context of the judgment. A typical judgment is popular for certain subtopics. So, if two judgments cite the same judgment, they also agree on the subtopic most of the cases."

Data from the Indian Supreme Court was also the basis for a study performed by Mandal et al. (2017) on measuring similarity between legal documents. The authors' related work review classified the previous proposals into network-based methods (which utilize the citation network among legal documents), text-based methods (which utilize the textual content of legal documents), and hybrid methods (based both on the text and on the citation network). In the latter case, the authors refer to the *paragraph-link* (PL) concept introduced by Kumar et al. (2013), in which every document is first broken into its constituent paragraphs. Then, each paragraph is considered as a separate entity. A similarity score across all pairs of paragraphs is measured. If the similarity between any two paragraphs is higher than a threshold, then a PL is added between the documents containing these two paragraphs.

As a limitation of the network-based approach, Mandal et al. (2017) mention the usual sparsity of the citation networks, making the method's applicability sometimes unviable. Regarding the text-based approaches, the authors sustained that previous works' methods were predominantly primitive, based mainly on term-frequency inverse document frequency (TF-IDF) for text vectorization. The authors explored different document representations (the whole document, set of paragraphs of the document, summaries of the document, and the text surrounding a citation). Likewise, other vectorization techniques such as topic modeling (Latent Dirichlet Allocation - LDA) and neural-network-based approaches (word embeddings – Word2Vec and document embeddings – Doc2Vec) were evaluated, having the similarity score calculated by the cosine between two vectors. The hypothesis was that these methods could capture the semantics of the documents and better formulate the similarity between them, even if different terms having the same semantic context were used in different documents.

On Mandal et al. (2017) experiment evaluation, 47 pairs of cases were valued by legal experts according to similarity on a scale of 0 to 10 and considered gold-standard. The Pearson correlation between the similarity measure obtained from the models and the gold standard was then computed. The Doc2vec similarity over the whole document was found to correlate most with the expert

judgment (Mandal et al., 2017). A two-class classification also evaluated the same model, a document pair being considered similar if the expert score was greater than five or judged similar by the Doc2vec model with a score greater than 0.5, resulting in superior Accuracy compared to the baseline.

Noteworthy document representation approaches have been proposed by Thenmozhi et al. (2017) when studying precedence retrieval for legal documents. In their experiments, the authors represented the documents by extracting their concepts or concepts and relations. After employing part-of-speech (POS) tags, the authors considered all the nouns to obtain the concepts of a document and the set of verbs to capture the relations. The different representations were compared among them, showing significant differences in the performances. Despite not comparing the assemblies with a baseline including full documents, their study presented an approach that deserves further exploration.

In 2019, an Artificial Intelligence for Legal Assistance (AILA) track was released by the Forum for Information Retrieval Evaluation (FIRE) to assess different methods for retrieving precedent cases and statutes given a factual scenario (Bhattacharya et al., 2020). With nine teams and 23 submissions, one of the tasks was for 50 different queries to identify the most similar case documents from a 2.914 Indian Supreme Court prior case database consisting, therefore, in a rank task. The assemblies that achieved the best and the third-best results were built by Zhao et al. (2019) using BM25-based ranking functions. The assemblies that achieved the second-best and fourth-best results were built by Gao et al. (2019), applying, respectively, cosine similarity to TF-IDF vectors and BM25 scores to topics extracted with TF-IDF. BM25 is a term-weighting and document-scoring function grounded in the probabilistic information retrieval model, characterized by associating the notion of relevance to a query document pair as an explicit variable. According to Robertson & Zaragoza (2009), the model revolves around the notion of estimating a probability of relevance for each pair and ranking documents with a given query in descending order of probability of relevance. The authors refer to the family of models that led to the development of BM25 as the Probabilistic Relevance Framework (PRF). It is worth mentioning that BM25 provides a score that is a function of each query and a document. The ranking is made by computing the score against each document in the corpus. This ranking is, however, specific to the query and does not have an upper bound, i.e., there is no concept of 'most significant' relevance, meaning the scores should not be compared between queries.

Still, the vector distance and so-called prevalent method for measuring similarity between documents, the cosine similarity, was confronted by Whissell & Clarke (2013) with measures based on three ranking functions, specifically BM25, LM, and DFR, employing one nearest neighbor experiment and one clustering experiment on eight different annotated datasets.

The authors concluded that the proposed measures would be preferable as inter-document similarity measures than TF-IDF with cosine, independent of the application. Also, in the more relevant experiment to this study, the nearest neighbor one, the BM25-based approach revealed the best performance.

Ranera et al. (2019) tested an assembly consisting of a Doc2Vec vectorization and similarity measured by cosine applied on a dataset composed of 59,742 Philippines Supreme Court case decisions written in English. Their related work review included the text-based and network-based approaches introduced by Kumar et al. (2011), the hybrid-based approach presented by Kumar et al. (2013), and the previously cited ensembles described in the study of Mandal et al. (2017). The authors trained a

Doc2Vec model on the dataset and evaluated the experiment through Spearman's correlation coefficient with a set of twenty-five pairs of cases evaluated by legal experts. The model was not compared to a baseline, and the result was a positively strong correlation (0.7691) which led the authors to classify the model as effective.

An extensive investigation of 56 different assemblies, crossing eight document text representation techniques and seven similarity measurement techniques, was performed by Mandal et al. (2021) when computing textual similarity across Indian Supreme Court Cases. The applied methods included authors' designed methodologies, the Bidirectional Encoder Representations from Transformers (BERT) model proposed by Devlin et al. (2019), and legal domain Word2Vec embeddings (Law2Vec) trained and made public by Chalkidis (2018).

As document representations, the authors examined the whole document, the set of sentences of the document, the set of paragraphs in a document, summaries of the document, catchphrases extracted from the document, and the set of RFCs (the text surrounding a citation) from the document.

While comparing among the different methodologies of measuring similarity, it is found that the embeddings learned by the neural network techniques (Word2vec, Doc2vec, Law2Vec) perform comparably with the other techniques for the stated task. In fact, we see that BERT yields substandard results for the similarity estimation task when trained on the dataset of 33,500 Indian Supreme Court case documents (Mandal et al., 2021). The authors also observed that the more traditional vectorization methods (such as the TF-IDF and LDA) that rely on a bag-of-words representation performed better than the more advanced context-aware methods (like BERT and Law2Vec).

3. METHODOLOGY

In the current violation analysis business process represented in Appendix 1, the proceedings are distributed among internal legal analysts who produce an evaluation of the case, subsiding the decision-making. The normative competence for judging each case depends on the infringed rule. It may lie with a single administrative judge or the agency board of directors, composed of five members. The number of participants in this business process makes it even more challenging to consider the existing jurisprudence. Furthermore, each analyst can evaluate similar cases differently over time, mainly depending on the decision-makers to maintain coherence. A tool based on the studies described in this dissertation could help the stakeholders mentioned above obtain visibility on the history of court judgments. In practical terms, by processing the trial's initial document, we intend that, during the analysis and judgment phases, analysts and judges receive advice for the decision outcome (a binary classification from study 1) and recommendations of similar cases according to study 2.

It is common to have different administrative judges and distinct compositions of the directors' board (in the analyzed data, there were four different judges and two different board compositions), which obstacles performance across time and under other circumstances. On the other hand, the differences between judicial and administrative proceedings are advantageous for text feature extraction. Although complaints may initiate the proceedings, most of the potential violations are identified by inspectors during their routine activities, leading to infraction notices of similar content. They commonly refer exclusively to infringed legislation and do not contain a significant number of references to decisions taken in similar cases.

3.1. DATA EXTRACTION

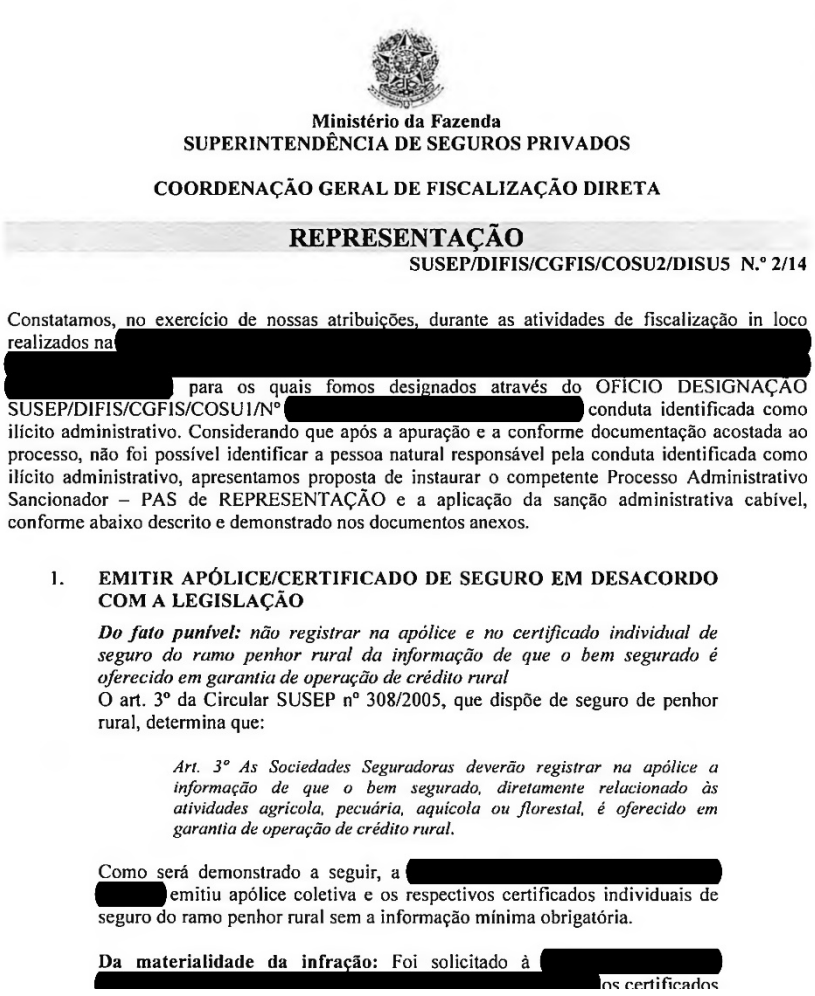
Data from two SUSEP's internal systems were used: the penalties system and the administrative process system. The former keeps track of the potential violation analysis lifecycle, storing metadata as violation date, infringed regulations, whether the violation was committed by a supervised natural person or a legal entity, decision date, and the existence of mitigating or aggravating circumstances. The later system stores the text documents of the proceedings in a timely order.

Three criteria were considered to create the dataset: the potential administrative violation must have had a first instance decision by an administrative law judge, which means having a decision result in the penalties system, the administrative proceeding must have been born-digital so that data can be adequately parsed (documents in HTML or PDF), and the alleged infringement must be still among the rules in force.

Each proceeding may have one or more potential violations that are regularly described in a single complaint or infraction notice, from now on referred to as the primary document. It poses an additional challenge to divide the text related to each infraction properly. The violation was used as the unit of analysis, extracting the primary document for each of them. Some proceedings also contained drafts and rectifications, meaning that there were two or more copies of the same procedural documents in these cases. As a rule, the last versions were believed to be final.

BeautifulSoup (Richardson, 2007) extracted text from the HTML files, and pdfminer.six (Shinyama et al., 2019) served the same purpose for PDF files. Different parsers were built for complaints and

infraction notices, in which sets of regular expressions helped remove prologues and epilogues and extract the core text for each violation. Figure 1 shows an example of an infraction notice stored as a PDF file, and Appendix 2 contains a free English translation of the text.



Ministério da Fazenda
SUPERINTENDÊNCIA DE SEGUROS PRIVADOS

COORDENAÇÃO GERAL DE FISCALIZAÇÃO DIRETA

REPRESENTAÇÃO
SUSEP/DIFIS/CGFIS/COSU2/DISU5 N.º 2/14

Constatamos, no exercício de nossas atribuições, durante as atividades de fiscalização in loco realizados na [redacted] para os quais fomos designados através do OFÍCIO DESIGNAÇÃO SUSEP/DIFIS/CGFIS/COSU1/Nº [redacted] conduta identificada como ilícito administrativo. Considerando que após a apuração e a conforme documentação acostada ao processo, não foi possível identificar a pessoa natural responsável pela conduta identificada como ilícito administrativo, apresentamos proposta de instaurar o competente Processo Administrativo Sancionador – PAS de REPRESENTAÇÃO e a aplicação da sanção administrativa cabível, conforme abaixo descrito e demonstrado nos documentos anexos.

1. **EMITIR APÓLICE/CERTIFICADO DE SEGURO EM DESACORDO COM A LEGISLAÇÃO**

Do fato punível: *não registrar na apólice e no certificado individual de seguro do ramo penhor rural da informação de que o bem segurado é oferecido em garantia de operação de crédito rural*
O art. 3º da Circular SUSEP nº 308/2005, que dispõe de seguro de penhor rural, determina que:

> *Art. 3º As Sociedades Seguradoras deverão registrar na apólice a informação de que o bem segurado, diretamente relacionado às atividades agrícola, pecuária, aquícola ou florestal, é oferecido em garantia de operação de crédito rural.*

Como será demonstrado a seguir, a [redacted] emitiu apólice coletiva e os respectivos certificados individuais de seguro do ramo penhor rural sem a informação mínima obrigatória.

Da materialidade da infração: Foi solicitado à [redacted] os certificados

Figure 1 - PDF Infraction notice sample. This document describes the findings of an inspection team and may contain one or more potential violations.

3.2. DATASET CHARACTERISTICS

The result was a dataset with 1,108 violations, including their primary document text extract. The following metadata was of interest for the first study: infringed regulation identifier (*faltaId*), whether the offender is a natural person or a legal entity (*pessoaFisica*), violation date (*infracaoData*), decision date (*julgamentoData*), and decision type identifier (*julgamentoDecisaoId*). Excepting the features related to the decision, all the others are available at the very beginning of a violation analysis providing an early prediction of its outcome.

In a subset of 109 samples (9.8%), the judgment date was missing. Considering that this is a noticeable percentage of samples, in these cases, the judgment date was set to correspond to the average of all violations previously judged instead of discarding such violations. Decision dates ranged from June 2016 to August 2020. The possible outcomes identified by *julgamentoDecisaoId* were five. One appeared in 830 samples (74.9%) and represented the situation in which the decision-maker

considered a violation to have existed, i.e., the case was subsistent. The remaining four included the following circumstances, jointly representing the negative (not subsistent) case: proceedings filed without judgment on the merits (1.6%), those in which the judge(s) decided that a violation had not occurred (16.6%), proceedings that generated a recommendation for the company and not a penalty (0.5%), or even those extinct by any legal reason (6.3%). Finally, in 14.0% of the samples, the proceeds' origin was a customer complaint, and the remaining originated from infraction notices.

3.3. STUDY 1 METHODOLOGY

3.3.1. Study hypothesis

As illustrated in Figure 2, this study employs a two-layer cascade classification model to predict whether a potential violation will be considered subsistent, i.e., whether an administrative judge will consider a violation to have occurred. The hypothesis under evaluation is that the proposed model reaches a better combination of Precision and Recall than two baselines: a metadata-only classifier and a text-only classifier. The proposed model uses features extracted from the text (n-grams) and features extracted from the metadata. From a business perspective, the objective is to evaluate one outcome against the others, turning the task into a binary classification. We implement an architecture based on the Cascade Generalization (Gama & Brazdil, 2000) to explore such an ensemble model, sequencing the prediction obtained from the textual features and those brought by metadata but not providing the final classifier of the original textual attributes.

Based on Katz et al. (2017) contributions, this study satisfies the concepts of generality, consistency, and out-of-sample applicability, proposing a practical approach for diverse administrations, provided they store decisions text and metadata. The generality property is addressed by evaluating the model on different decision-makers, a single judge, or the board of directors, varying compositions across time. In its turn, the out-of-sample applicability is intrinsic to the validation strategy. All training samples precede the test samples on time, guaranteeing that all information the model needs to produce an estimate is known before the decision date. Furthermore, consistency across time is directly addressed by time-based cross-validation. This validation method also indirectly delivers consistency across case issues and courts.

Text-based features were used to train the first classifier. The decision to use text representations based on n-grams and apply TF-IDF vectors or Latent Dirichlet Allocation (LDA) topics as embeddings relates to preserving visibility over the textual variables that contribute to the classification result in the first stage. Although semantic features may be captured by neural network-based models (e.g., Doc2Vec, Word2Vec, GloVe) and transformer-based models (e.g., Universal Sentence Encoder, BERT), they are not interpretable by their nature. Moreover, the cascading strategy for combining heterogeneous features through two classifiers enables one to evaluate the classification's contribution from the first stage to the second stage.

The second classifier was fed with the predicted class probabilities from the early stage as an additional feature, along with violation metadata. It should be noted that ensemble techniques such as bagging and boosting are not feasible to connect the first and second stages in this assembly because they bootstrap with samples from the same data set. Similarly, the widespread stacking technique could only be employed if the second stage was a meta-classifier.

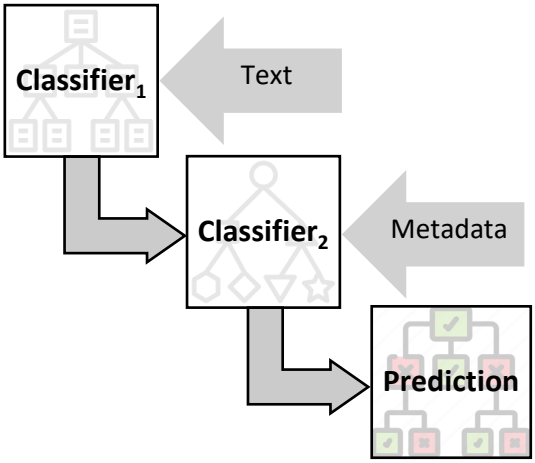


Figure 2 - Cascading classifiers to combine text and metadata

Following the explainability levels described by Bibal et al. (2021), we chose word embeddings and an ensemble technique to provide features meaningful for human interpretation. Also, after predicting each stage for a decision with the cascade model, we use the Shapley Additive Explainability (SHAP) method detailed by Lundberg & Lee (2017) to approximate each feature's contribution and reverse-engineer the output. This model agnostic approach can provide local explainability, i.e., understanding a decision prediction standalone stages, as it preserves human-recognizable features to interpret their contributions. In this sense, not only does the proposed model address the combination of variables in a decision, but it also allows identifying the text extracts that contribute to the decision (first stage) and the infringed legislation relevant to the case (second stage).

3.3.2. Feature engineering

Three features were also extracted from data to address possibly salient aspects of the decisions and feed the second stage classifier. First, the time difference between the decision date and the violation date, i.e., the time the administrative court took to deal with the case and decide on it, was stored in the feature *tempoAteJulgar*. Next, the *textLength* feature was created to store the primary document length for each violation. Finally, the time difference between the violation date and January 1st, 2000 was stored in the *daysFrom2000* feature and substituted *infracaoData*, addressing the case recency and treating the date as an ordinal and continuous variable. In summary, the following features were made available to the second stage classifier:

- *faltaId:* infringed regulation identifier;
- *pessoaFisica*: whether the offender is a natural person or a legal entity;
- *julgamentoData*: decision date;
- *tempoAteJulgar*: time difference between the decision date and the violation date;
- *textLength*: primary document length;
- *daysFrom2000*: time difference between the violation date and January 1st, 2000;
- *subsistente*: the binary target feature, corresponding to the decision result.

The *faltaId* feature had a high cardinality when considered the number of samples, with 97 different categories. As a data preparation measure to mitigate this characteristic, the *faltaId* classes were sorted by the number of instances they accounted for and kept the values representing 90% of the

cases, resulting in 40 categories when considered the entire dataset. The remaining possible *faltaId* were grouped into a default category. Finally, this feature was encoded into dummy binary variables, making it suitable for the classification algorithms. *InfracaoData* was transformed into a continuous variable ranging between the minimum and the maximum date in the dataset.

3.3.3. Baselines

Our study aimed to compare the experimental results with both a classifier based only on metadata and a classifier based on features extracted from the text. For the first baseline, the available metadata was used to grid-search on seven estimators using different learning methods: a lazy (or instance-based) learner represented by k-Nearest Neighbors (kNN), a neural network (or Multilayer Perceptron), a tree-based classifier (Decision Tree), a Bagging Classifier using decision trees as its base, a bagging classifier employing random subsampling of features (Random Forest), a boosting classifier (XGBoost) and an analogy-based learner (Support Vector Classifier). All estimators used versions implemented on Scikit-Learn (Pedregosa et al., 2011), excepting XGBoost implemented by T. Chen & Guestrin (2016).

3.3.4. First stage classifier

Text preprocessing was the first step of the pipeline, removing ubiquitous and non-discriminative words and phrases from the primary documents using regular expressions, e.g., phone numbers and URLs and business unit names. Additionally, POS tagging was used to remove articles, conjunctions, pronouns, and currency symbols. Several POS taggers were tested. The Mac-Morpho Brazilian Portuguese corpus (Fonseca et al., 2015) associated with Brill Tagger (Brill, 1992) from the Natural Language Toolkit – NLTK (Bird et al., 2009) obtained the best performance to classify sentences. NLTK was also used to apply, in sequence, tokenization, stopword removal, and stemming. Tokenization is the text segmentation into basic units (tokens) such as words and punctuation, while Stemming accounts for reducing words to their root form based on pre-established rules. For example, a rule may state that any expression with *-ing* as a suffix will be reduced by removing the suffix (Bird et al., 2009). The last step used the RSLP Stemmer for Portuguese (Orengo & Huyck, 2001).

We also used POS tagging to extract words and summarize the text according to three summarization strategies: the first and most basic returned the full stemmed text. We obtained the second and third summaries by extracting the nouns (concepts) and nouns and verbs (concepts and relations). In both cases, the resulting text was also later stemmed.

For vectorization, the bag-of-words model was used to represent the text corresponding to each violation. The bag-of-words approach assumes that a text is nothing more than a histogram of the words it contains. Thus, words' order or words' context is not considered (Theodoridis, 2020). For vectorial document representation, two approaches were tested. First, words' occurrence frequency was normalized by the number of violations in which each word occurs, i.e., through term frequency-inverse document frequency (TF-IDF). Also, in this case, words appearing in only one document or more than 80% of the corpus were ignored. The second approach involved counting the number of occurrences of each word in the documents and applying the Latent Dirichlet Allocation (LDA) model, representing each document by its composition in LDA topics.

The TF-IDF representation proved effective in different scenarios, generally with up to four words in each n-gram. In our case, n-gram sizes varied from 1 to 4. Each vectorization trial utilized pairs of the following n-gram sizes: (1,2), (2,3), and (3,4). The vocabulary size, i.e., the number of different n-grams obtained when mapping terms to features, reached 185.921 with n-grams of size (1,2) and the full stemmed text. This amount of features required applying a dimensionality reduction technique, both for performance reasons and to avoid the *curse of dimensionality*. This phenomenon is observed when estimating an arbitrary function by a high number of features minimizing the approximation error. According to L. Chen (2009) the number of samples needed for this estimation with a given level of Accuracy grows exponentially with the number of input variables (i.e., the dimensionality) of the function. In terms of computational effort, it grows exponentially with the number of unknown variables. In such cases, dimensionality reduction techniques transform the data from a high-dimensional space into a low-dimensional space, retaining the original representation's properties as much as possible.

Hence, Singular Value Decomposition (SVD) was applied to the vectorized representation for each n-gram set using a components number equal to 1% of the feature space. It means approximating the feature-space matrix A with another lower-rank matrix B, i.e., truncating A to a specific rank *r* corresponding to 1% of this study's dataset features. After obtaining the variance explained by a projection to each principal component, the number of components necessary to explain 95% of the total variance was selected. Thus, it was possible to get a dimensionality that best suited each n-gram set. The correspondence between text summarization strategy, n-grams size, and the number of components varied from 655 components for the full text and n-grams of size (3,4) to 550 components for the concepts and relations text with n-grams of size (1,2). Consequently, it was possible to grid search on various text summaries, n-gram sizes, and their corresponding number of features.

Differently, it was not necessary to apply dimensionality reduction techniques to text representations that used LDA because, in this case, the vector dimension is specified by the number of topics. In this study, we tested configurations with 20, 40, and 80 topics. The last step on the first stage classifier was the classification algorithm, for which the same implementations described in section 3.3.3 were investigated.

Following the literature considerations about out-of-sample prediction and model generality, using timely ordered data is mandatory for cross-validation and real-world applicability. Despite being a challenging premise, it was possible to demonstrate that reasonably high performance is still achievable. The samples for training were limited to only that available before all decisions in the test set. On the other hand, the model used a novel approach to divide the samples between the training and test sets. The previously adopted strategies considering the temporal dimension consisted of ordering the samples and defining cutting points on time for training and testing sets, e. g., setting the training set's final date to be December 31st every year and predicting an arbitrary timeframe within the following year. Our study used time-based cross-validation as proposed by Herman-Saffar (2020). The predictor was trained with samples taken from a specific period (e.g., twelve months) and tested on samples corresponding to an adjacent period (e.g., a month). All test sets were constrained to start after 2020, January 1st. The significant difference in this approach is that the splitting points slide along with the samples, and all of them may be used to apply time-based cross-validation. It is worth mentioning that this method creates sets containing a fixed time frame instead of a fixed number of

records. Tables 1 and 2 present train-test splits for 12 months and 24 months in the training set, respectively, and one month in the test set.

Split	Train period	Test period	Train (1) samples	Train (0) samples	Test (1) samples	Test (0) samples	Test/Train proportion
0	2019-01-01 - 2020-01-01	2020-01-01 - 2020-02-01	175	85	42	18	23%
1	2019-02-01 - 2020-02-01	2020-02-01 - 2020-03-01	216	103	26	0	8%
2	2019-03-01 - 2020-03-01	2020-03-01 - 2020-04-01	240	86	81	13	29%
3	2019-04-01 - 2020-04-01	2020-04-01 - 2020-05-01	320	99	76	15	22%
4	2019-05-01 - 2020-05-01	2020-05-01 - 2020-06-01	393	110	97	36	26%
5	2019-06-01 - 2020-06-01	2020-06-01 - 2020-07-01	486	135	150	8	25%
6	2019-07-01 - 2020-07-01	2020-07-01 - 2020-08-01	635	139	64	78	18%
						Test/Train proportion mean	22%

Table 1 - Train-test split for 12 months x 1 month

Split	Train period	Test period	Train (1) samples	Train (0) samples	Test (1) samples	Test (0) samples	Test/Train proportion
0	2018-01-01 - 2020-01-01	2020-01-01 - 2020-02-01	178	88	42	18	23%
1	2018-02-01 - 2020-02-01	2020-02-01 - 2020-03-01	220	106	26	0	8%
2	2018-03-01 - 2020-03-01	2020-03-01 - 2020-04-01	246	106	81	13	27%
3	2018-04-01 - 2020-04-01	2020-04-01 - 2020-05-01	327	119	76	15	20%
4	2018-05-01 - 2020-05-01	2020-05-01 - 2020-06-01	403	134	97	36	25%
5	2018-06-01 - 2020-06-01	2020-06-01 - 2020-07-01	500	170	150	8	24%
6	2018-07-01 - 2020-07-01	2020-07-01 - 2020-08-01	649	178	64	78	17%
						Test/Train proportion mean	20%

Table 2 - Train-test split for 24 months x 1 month

The next step was to divide the dataset according to four different training/test timeframe combinations (Table 3). The reason for testing different time windows is to assess the best interval for the training sample. Intuitively, it may seem that the model's accuracy will be better the more extensive the time window (and consequently the number of documents used for training). On the other hand, limited windows can better capture law and jurisprudential changes. The results obtained in this study demonstrate the need to evaluate different training windows for each dataset.

Train timeframe (months)	Test timeframe (months)	Number of splits
6	1	7
12	1	7
18	1	7
24	1	7

Table 3 - Training/test timeframes combinations

The purpose of this study is not to fine-tune the classifiers' settings but to evaluate the hypothesis of better performance of a cascading model over metadata-only or text-only models. Consequently, the search space for hyperparameters on classifiers included only the learning rate control and the classifier base architecture (activation functions, number of layers, nodes, neighbors, leaves, and estimators). Table 4 summarizes the parameter space used on grid-search for the first stage.

Pipeline step	Algorithm	Unchanging hyperparameters	Hyperparameters search space		
		Maximum document frequency	Text representation	N-gram range	Components
Vectorization	TF-IDF + SVD	0.8	Full	(1, 2)	605
			Concepts	(1, 2)	550
			Concepts and Relations	(1, 2)	550
			Full	(2, 3)	637
			Concepts	(2, 3)	595
			Concepts and Relations	(2, 3)	593
			Full	(3, 4)	655
			Concepts	(3, 4)	620
			Concepts and Relations	(3, 4)	616
	LDA	-	Number of topics: 20, 40, 80		
Classification	k-Nearest Neighbors	-	Number of neighbors: 3, 5, 7 Weights: uniform, distance inverse		
	Neural Network	-	Hidden layer sizes: 100, 200, (100, 100) Activation function: relu, logistic Solver for weight optimization: adam, lbfgs Alpha: 0.01, 0.03, 0.1, 0.3, 1.0		
	Decision Tree	-	Splitter: best, random Minimum number of samples in a leaf: 1, 6, 11 Maximum number of features: all, log2		
	Bagging Classifier	-	Number of estimators: 10, 25, 50		
	Random Forest	-	Number of estimators: 10, 50, 100 Minimum number of samples in a leaf: 1, 6, 11 Maximum number of features: all, log2		
	XGBoost	-	Number of estimators: 25, 50, 100 Minimum weight needed in a child: 1, 6, 11		
	Support Vector Machine	-	Kernel: radial basis function, polynomial, linear Regularization: 0.001, 0.01, 0.1, 1.0		

Table 4 - First stage parameter space

The first stage grid-search results on classifiers and cross-validation schemes are summarized in Table 5, represented by the F1 weighted scores.

Cross- validation schema X Vectorization X Classifier X Text Representation					
	Classifier	**6 x 1**	**12 x 1**	**18 x 1**	**24 x 1**
TF-IDF	k-Nearest Neighbors	0.876	0.867	0.861	0.856
	Neural Network	0.850	0.852	0.857	0.889
	Decision Tree	0.837	0.863	0.870	0.869
	Bagging Classifier	0.844	0.865	0.847	0.848
	Random Forest	0.843	0.865	0.850	0.850
	XGBoost	0.837	0.874	0.866	0.872
	Support Vector Machine	0.848	0.851	0.851	0.851
LDA	k-Nearest Neighbors	0.837	0.854	0.861	0.854
	Neural Network	0.857	0.864	0.875	0.868
	Decision Tree	0.833	0.830	0.870	0.866
	Bagging Classifier	0.870	0.851	0.870	0.878
	Random Forest	0.823	0.887	0.866	0.850
	XGBoost	0.844	0.844	0.852	0.853
	Support Vector Machine	0.863	0.796	0.837	0.885

Full Concepts Concepts & Relations

Table 5 - First stage F1 weighted scores

F1 Score was chosen as the primary evaluation metric because it best suits datasets with imbalanced classes. It is an evenly-weighted combination of Precision and Recall, defined as the harmonic mean of these two metrics. These and other relevant measures are presented in Appendix 3. The Precision is the ratio between the number of positive samples classified as positive (true positives) and all the samples predicted to be positive. It is also called positive predictive value (PPV). The Recall represents the ratio between the *true positives* and the positive samples (sum of true positives and false negatives).

The weighted average implementation of Scikit-Learn was used to select the best classifier for each stage. This particular case calculates the F1 Score for the positive and negative classes and finds their average weighted by the Support (the number of true instances for each label). Otherwise, because of the higher occurrence of positive samples in the dataset, it would be possible to have good F1 Scores even if the model predicted positives all the time.

Different techniques were used to attenuate the imbalance effects generated by an asymmetrical classes distribution of 3:1 towards the positive outcome. Random oversampling of the minority class and Synthetic Minority Over-sampling Technique (SMOTE) (Chawla et al., 2002) were integrated into the Random Forest classifier assembly and did not improve the model performance.

At the end of the first stage, no classifier showed superior performance in the different cross-validation schemes. The best overall result was obtained with a Neural Network using TF-IDF vectors created from the concepts and relations representation. The second-best overall result was obtained with a

Random Forest Classifier, using LDA vectors created from the full-text representation. None of the cross-validation schemas, vectorizers, or text representations consistently outperformed other configurations. The best scoring setups in each cross-validation schema are highlighted in Table 5.

3.3.5. Second stage classifier

The classifiers of the first stage (Table 5) were later applied to predict outcomes for the entire dataset, and the result was made available to the second stage as a new feature. Thus, the variables were the same as those used for the metadata-only predictors added from the previous step predictions. A grid search with the same classifiers mentioned in Section 3.3.3 was then applied to the new dataset. All cross-validation schemas were tested in conjunction with dimensionality reduction to avoid overfitting.

It is worth mentioning that after selecting the classifier for the first stage, requiring only text-based features, the second classifier must be refitted for each training/test split in the second stage cross-validation. If not, there will be *statistical leakage* between the steps, i.e., a classifier fitted in the first stage to the entire dataset would already incorporate the whole dataset's characteristics. It means including data that would not be available when fitting the second stage in a real-world problem. For the same reason, the transformations implemented in *faltaId* needed to be included in each training/test split.

Complimentary, to compare the results of the proposed model with metadata-only classifiers, we also performed a hyperparameter search on the classifiers cited in this study employing only non-textual features. Unlike the text-based predictions, the results summarized in Table 6 show the outstanding performance of the gradient boosting-based algorithm and a better average result when the training sample set was limited to cases decided less than 12 months ago.

Cross- validation schema X Classifier					
	Classifier	**6 x 1**	**12 x 1**	**18 x 1**	**24 x 1**
Baseline 2 (metadata-based)	k-Nearest Neighbors	0.822	0.818	0.825	0.824
	Neural Network	0.744	0.771	0.773	0.772
	Decision Tree	0.805	0.842	0.816	0.819
	Bagging Classifier	0.830	0.851	0.814	0.822
	Random Forest	0.824	0.848	0.824	0.827
	XGBoost	0.831	0.852	0.834	0.834
	Support Vector Machine	0.754	0.790	0.798	0.846

Table 6 - Metadata-only classifiers F1-weighted scores

3.4. STUDY 2 METHODOLOGY

3.4.1. Study hypothesis

This study assumes that by transforming the violation's primary documents into vectors and applying similarity measures between these text representations, it is possible to identify pairs of cases that

satisfy the human notion of similarity. We do so by comparing the models' outputs with an expert's gold standard and a baseline model. We utilized a representative dataset of administrative violations to investigate combinations resulting from four text representations along with four similarity measurement techniques based on text vectorization and cosine similarity, plus the BM25 ranking function.

3.4.2. Gold-standard and evaluation metrics

Each assembly's results evaluation demanded comparing the retrieved documents (and respective violations) with those considered similar in a real-world scenario. For this reason, a violations sample was randomly selected from the 1,109 dataset records, and three legal experts from SUSEP manually evaluated and classified the pairwise similarity for the extracted sample. To ensure the sample was representative of the dataset, we used Cochran's equation for large populations (Cochran, 1977). In our case, the population is composed of the 614,386 possible combinations of the 1,109 violations. Since we did not know the population proportion, i.e., the percentage of pairs with a level of similarity different from zero, we assumed 50% as a rule of thumb. For a confidence level of 95% and a confidence interval of 3%, the needed sample size was 1065 pairs. By selecting 50 random cases, we formed 1.225 combinations (C_2^{50}). Later, we observed from the experts' evaluations that the population proportion is about 10%, confirming the utility of the results, with a confidence level of 99% and an error margin of 2.21%.

The legal experts scored each violation pair in a range from 0 to 5, where 0 represented no similarity and 5 represented the highest level of similarity. Assuming (A, B) to be a randomly generated violation pair, the concept under evaluation was the utility of violation B as a research result during the analysis of violation A. Moreover, the similarity was understood as reciprocal. Each Expert evaluated all pairs, and the mean of the Expert's scores was considered the gold standard. Table 7 presents the number of pairs comprised in each similarity level, the mean, and the standard deviation of the scores for each Expert, and Table 8 presents the number of pairs comprised in each similarity interval when considered the experts' scores mean.

	Number of samples		
Similarity score	**Expert 1**	**Expert 2**	**Expert 3**
0	1124	1094	1174
1	36	29	8
2	12	21	7
3	12	23	9
4	39	17	12
5	2	41	15
Mean	0.337	0.140	0.214
Standard Deviation	1.099	0.734	0.813

Table 7 - Similarity score distribution according to legal experts

	Number of samples
Similarity score	**Experts mean**
0	1058
0 < score ≤ 1	97
1 < score ≤ 2	21
2 < score ≤ 3	9
3 < score ≤ 4	17
4 < score ≤ 5	23
Mean	0.230
Standard Deviation	0.815

Table 8 - Similarity score distribution when considered the experts' scores mean

It is possible to observe that the experts' scores distribution has very low variability (most results are equal to zero). Considering positive to be the case of a document pair having a level of similarity different from zero, we observe a very low Prevalence of the positive class. As a result, pairs with similarities different from zero can be seen as outliers. This situation is mainly derived from the fact that there are 94 different infringed regulations in a relatively small dataset. Therefore, it is expected that many cases are not similar to each other.

The variation identified among the scores' means and standard deviations denotes the high variability in humans' perception of similarity. This aspect is also recognizable in Figure 3, which represents the score by pair for all experts. Assessing the human cognition of similarity and understanding the parameters that influence this cognition is not among the objectives of this work. Meanwhile, this finding reveals the importance of comparing the various experimental setups' results with the individual experts' assessments, not only with the scores' mean. This approach made it possible to assess the ability to mimic the notion of similarity for each individual, i.e., simulating the case of an individual receiving a suggestion for a document similar to a text under analysis.

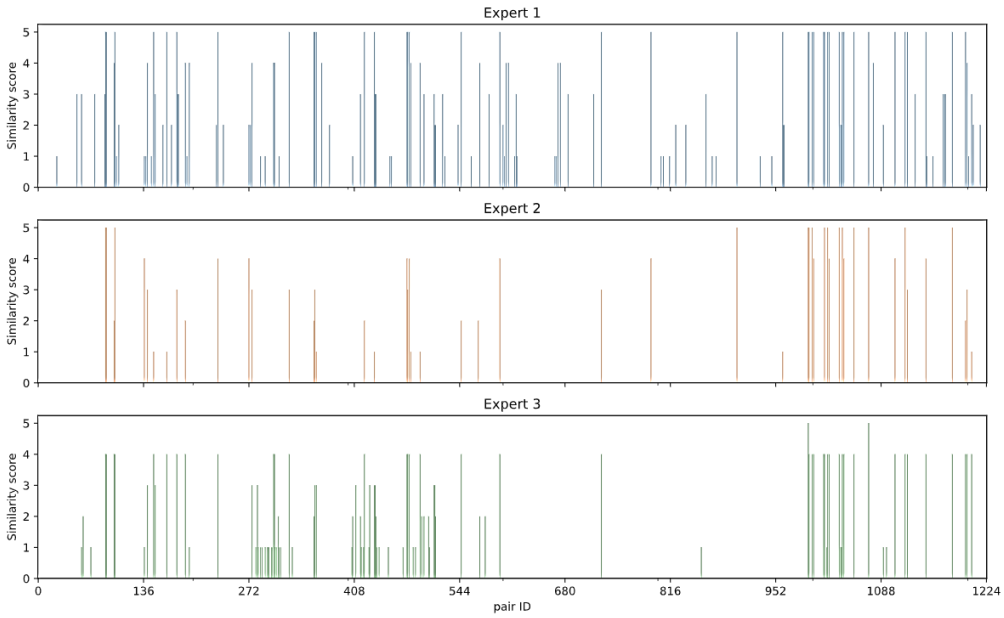


Figure 3 - Expert scores distribution

It is also expectable that the variability of the Expert's scores is much lower than the variability of the similarity scores obtained from the models. Furthermore, if running a linear regression from the Expert's scores versus the models' scores, we may expect error terms varying substantially because most of the Expert's scores are zero. Hence the data suffers from heteroskedasticity. In such situations, using the correlation coefficient to measure the strength of the relationship between the gold standard and the models' outputs can be misleading.

From the business perspective, it is essential to ensure that the highest number of relevant documents is retrieved when a new case is under analysis. To this purpose, first, the experiment was transformed into a two-class classification problem, with the positive class meaning that a document pair is 'similar' and the negative meaning 'not similar'. When a document pair had a non-zero similarity in the gold standard, it was classified as 'similar'. Otherwise, it was labeled as 'not similar'. The experiment objective is to retrieve the highest possible number of 'similar' documents for a legal case document under analysis. Hence, Recall is the most critical metric in this business problem, i.e., the fraction of relevant documents that are successfully retrieved or the number of appropriately retrieved violations divided by the number of results that should have been returned.

Precision or the Positive Predictive Value (PPV), in turn, becomes a secondary metric since it measures the fraction of relevant documents among the retrieved documents. It is also worth considering that Prevalence impacts the PPV of experiments. As the prevalence increases, the PPV also increases. Likewise, as the Prevalence decreases, the PPV decreases. In our dataset, we observe a very low prevalence of the positive class, which can distort the evaluation of an assembly if the PPV is the only metric taken into consideration. For the same reason, Accuracy is also not a good metric to evaluate the models' performance.

Consequently, this study adopts the mean Average Recall (mAR) as the primary evaluation metric, defined as the mean of the Average Recall (AR) across all recommendation thresholds, ranging from 1 to 10 recommended documents. The AR, in turn, is the Recall averaged over all fifty documents in the gold standard. Equations 1 and 2 illustrate these definitions.

For a real-world situation, the recommendation threshold is the number of predicted-similar suggestions presented to a legal analyst when a case is under analysis. Hence we set the upper limit due to the human capacity to analyze presented recommendations. This upper limit is also essential to restrict the effect of the number of recommendations on the Recall value. Otherwise, a Recall equal to 1 could be obtained by suggesting all documents in the dataset.

$$AR = \frac{\sum_{i=1}^{K} Recall_i}{K}, \qquad K = 50$$

Equation 1 - Average Recall

$$mAR = \frac{\sum_{i=1}^{N} AR_i}{N}, \qquad N \in [\![1,10]\!]$$

Equation 2 - mean Average Recall

As a secondary metric, we use Recall@5, defined as the proportion of similar documents found in the top-5 recommendations, averaged over all documents in the gold standard. In some cases, the calculation of Recall can cause a division by zero, happening if there are no relevant documents for a violation in the gold standard sample. For these particular cases, we set Recall to 1 since no relevant item is left unidentified in the top results.

$$Recall@5 = \frac{\sum_{i=1}^{K} \left(\frac{\#\ of\ relevant\ documents\ in\ top5}{\#\ of\ relevant\ documents} \right)_i}{K}, \qquad K = 50$$

Equation 3 – Recall@5

Finally, the mean Average Precision (mAP) and the Precision@5 are also presented. The former is defined similarly to the mAR, and the latter represents the proportion of recommended cases in the top-5 set that are relevant.

$$Precision@5 = \frac{\sum_{i=1}^{K} \left(\frac{\#\ of\ relevant\ documents\ in\ top5}{5} \right)_i}{K}, \qquad K = 50$$

Equation 4 – Precision@5

3.4.3. Experimental setup

The second study (finding analogous cases based on the primary document similarity) included the same textual preprocessing described in Section 3.3.4 was applied. The most promising components (document representations and similarity measurement techniques) were implemented following the related work.

3.4.3.1. Document representation

The first document representation was also the most basic, composed of the full stemmed text. From the study of Thenmozhi et al. (2017), which described encouraging results by extracting concepts and relations from the text, we used POS tagging to generate the second and the third representations by extracting, respectively, the nouns (concepts) and nouns and verbs (concepts and relations) from the whole document. In both cases, the resulting text was later stemmed.

This work also considered the results achieved by Mandal et al. (2021) by seeking legal issues into different levels of granularity, i.e., splitting the documents into paragraphs and sentences, obtaining document summaries, and extracting catchphrases from the text.

Because part of our dataset was extracted from PDF files, we lacked the paragraph markers to partition all documents into their paragraphs. On the other hand, it was possible to break documents into sentences by using the existing dots. To avoid the improper break of sentences when abbreviation dots were found, we ignored dots in a set of standard abbreviations for the Portuguese language (art., arts., s.a., sa., doc., docs., fl., fls., dr., dra., drs., cia., cias.) as well as consecutive dots that could represent suspension points. As in the other representations, the final text was stemmed.

The representations based on document summaries or the extraction of catchphrases, in their turn, showed the best results when associated with the TF-IDF vectorization method but still with a lower performance in comparison with the full-text representation. For this reason, these last specific representations proposed by Mandal et al. (2021) were not implemented in this work.

3.4.3.2. Document similarity measurement

After obtaining one corpus for each document representation, different methods were applied to convert text into vectors. The first one was the TF-IDF, in which hyperparameter configurations with unigrams plus bigrams and bigrams plus trigrams were tested. Like the first study, we ignored terms with a document frequency higher than 80% of the corpus or appearing in only one document. Besides the mentioned n-gram intervals, four TF-IDF vectorizers were trained on the four corpora, then the vectorizer trained on the full documents corpus was applied to all corpora, and the other three vectorizers were used to obtain embeddings for documents in the respective corpus. Thus, the fourteen configurations represented in Figure 4 were experimented with by employing TF-IDF vectorizers. For the sake of comparison to the prevalent method for document similarity retrieval, this study assumes the TF-IDF model created with the full stemmed documents using unigrams and bigrams as the baseline.

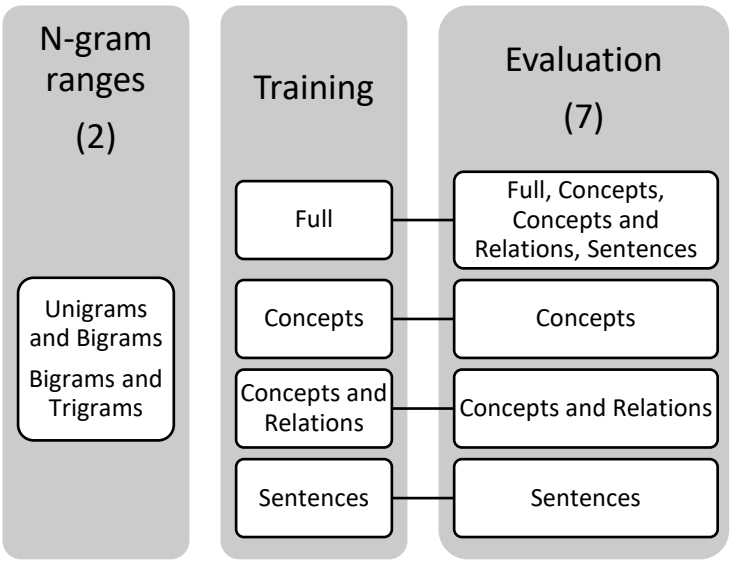


Figure 4 – The fourteen TF-IDF experimental configurations

The Word2Vec technique, based on word embeddings, was the second method used to create document representation vectors. Considering that Word2Vec generates embeddings for each word, the vectors for the text segments were obtained by the mean of the constituent word vectors, weighted by the TF-IDF score of the word. First, pre-trained Word2Vec models for the Portuguese language were obtained from the repository maintained by Hartmann et al. (2017) on the Interinstitutional Nucleus of Computational Linguistics (NILC-USP) of São Paulo University. Both the continuous bag-of-words (CBOW) and Skip-gram methods with the available embedding sizes of 100 and 300 were employed. Then, following the same methods and embedding sizes, other Word2Vec models were trained on each of the four corpora. Again, the models trained on the full corpus were

applied to all corpora. The other models were applied to the respective corpora, resulting in thirty-two assemblies represented in Figure 5.

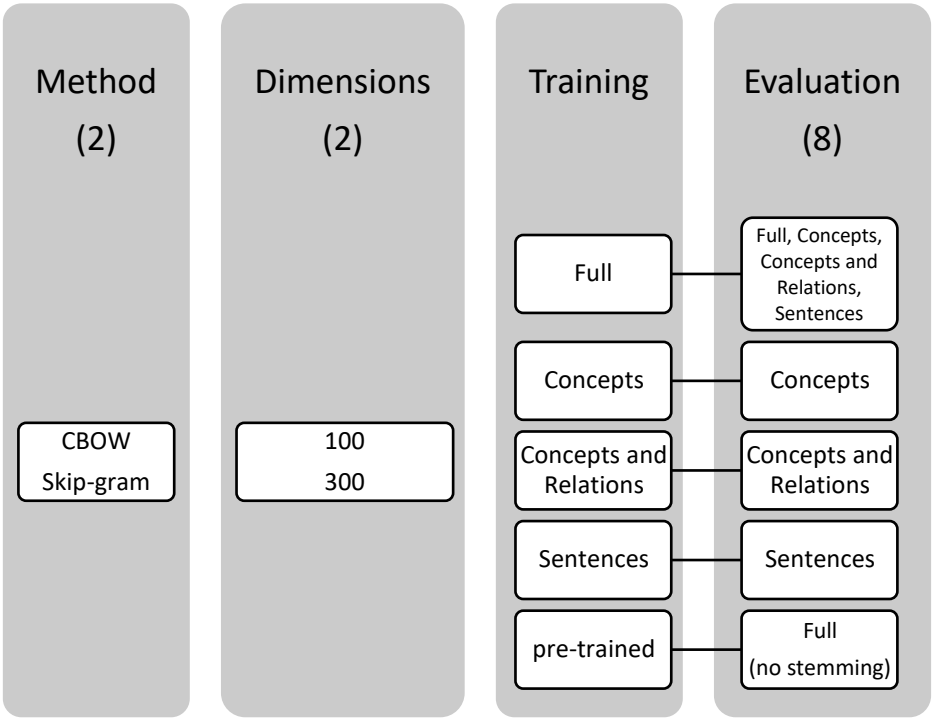


Figure 5 – The thirty-two Word2Vec experimental configurations

Next, the Doc2Vec technique was implemented to transform each document into vectors of 100, 200, and 300 dimensions. By representing the whole text into a vector, unlike the Word2Vec models, Doc2Vec does not require the combination of embeddings into a final vector. Following the previous model families, twenty-one assemblies resulted from training the model on all corpora, applying the full-corpus-trained model to all corpora and the other models to their respective corpora.

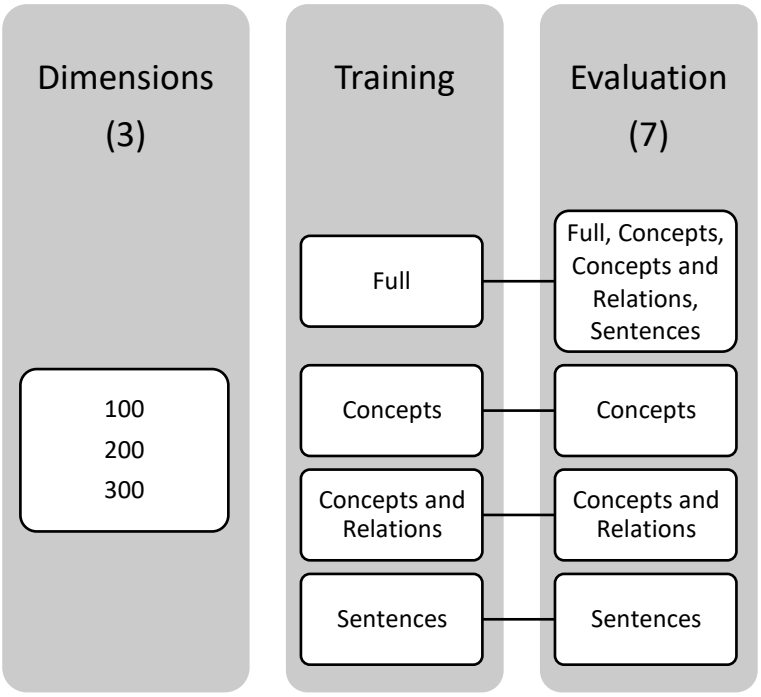


Figure 6 – The twenty-one Doc2Vec experimental configurations

LDA is the fourth model family tested in this study. Each corpus document is modeled as a given finite combination of an underlying set of topics in this model family, where a distribution across words characterizes each topic. In the context of text modeling, the probabilities of belonging to topics provide an explicit representation of a document. Following the previous assemblies, we trained LDA over the full text stemmed corpus with 10, 20, 40, and 80 topics and retrieved vectors with these dimensions for the different document representations. Later, we trained LDA models with the same topic range over the three remaining corpora and retrieved vectors for documents in the same corpora. The twenty-eight LDA assemblies are represented in Figure 7.

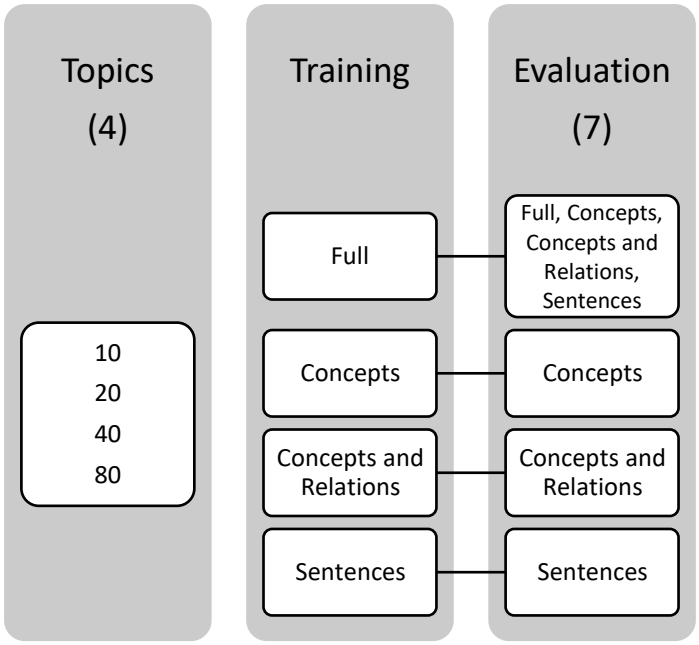


Figure 7 - The twenty-eight LDA experimental configurations

After transforming the document representations into vectors through each previously mentioned method, we applied cosine similarity, the predominant metric to evaluate document similarity. It is formally defined as the dot product between two vectors divided by the product of their magnitudes. It measures the cosine of the angle between these vectors. When applied to document similarity, the cosine similarity is typically preferred over Euclidean distance because even if two similar documents are far apart when compared by the Euclidean distance (due to the difference in the size of the documents), they may still have similar orientations. In this sense, the smaller the angle, the higher the cosine similarity.

Finally, following the promising results made public by Bhattacharya et al. (2020), we experimented with the BM25 model family by employing the BM25+ algorithm. This implementation addresses one deficiency of the standard BM25 in which the component of term frequency normalization by document length is not appropriately lower-bounded. As a result, long documents which do match the query term can often be scored unfairly by BM25 as having similar relevancy to shorter documents that do not contain the query term at all (Lv & Zhai, 2011).

As mentioned in Section 2.2, BM25 directly provides a score that is a function of each query and a document. For ranking documents in terms of similarity to a single document, we used the latter as the query under analysis. Therefore, computing cosine similarity between vectors is not the case, and

the output score can be directly used to retrieve the most similar cases. Figure 8 illustrates the experimental setup for this model's family.

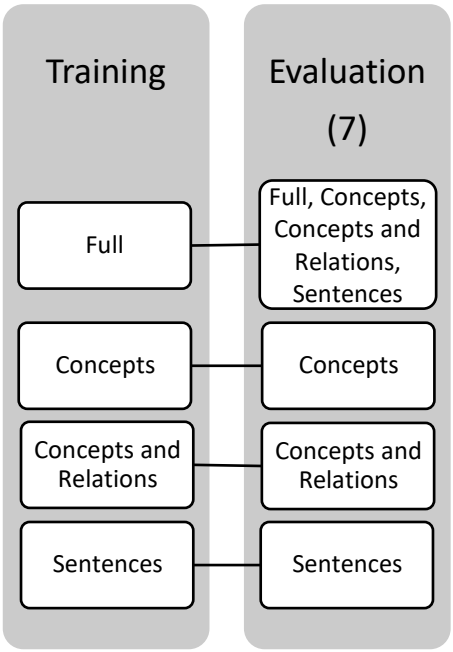


Figure 8 – The seven BM25 experimental configurations.

Among the one hundred and two setups evaluated in this study, we experimented with an approach different from that used by Mandal et al. (2021). In their study, these authors trained the model instances by presenting the full documents corpus only and, in each case, retrieved embeddings for the various document representations. As previously stated, in our study's case, the models were also trained on the document-representation-specific corpora to evaluate their performance compared to the versions trained by the full documents.

4. RESULTS AND DISCUSSION

4.1. STUDY 1

As mentioned in Section 3.3.5, a grid-search among the classifiers from Section 3.3.3 was implemented, with cross-validation according to the technique described in Section 3.3.4. Table 9 presents the results obtained in three assemblies: the first stage and baseline 1, built only on n-grams extracted from violations text, the baseline 2, using only violations metadata, and the proposed cascade classifier (composed of the first and second stages).

F1 Weighted Score					
Classifier		**6 x 1**	**12 x 1**	**18 x 1**	**24 x 1**
Baseline 1 First stage (text-based)		0.876	0.887	0.875	**0.889**
Baseline 2 (metadata-based)		0.831	**0.852**	0.834	0.846
Cascade classifier	**Stage 1 (text)**	k-Nearest Neighbors	Random Forest	Decision Tree	k-Nearest Neighbors
	Stage 2 (metadata)	Neural Network	XGBoost	XGBoost	XGBoost
		0.881	**0.900**	0.893	0.895

Full Concepts Concepts & Relations

Table 9 - F1 weighted score comparison between text-based prediction (first stage and baseline 1), metadata-based prediction (second stage and baseline 2), and the final assembly.

The results show that the two-stage model consistently surpassed both the n-gram and the metadata-based approaches. It obtained the best F1-weighted score for all cross-validation schemas, i.e., whatever the size of the sample used for training, the metadata classifier obtained superior performance, benefiting from the feature extracted from the text. Still, the full-text representation provided the best results for feeding the cascade classifier with the first stage predictions. With a higher number of n-grams, this representation extracts more variance from the data. Despite being computationally costlier, it seems to allow the cascade classifier to benefit from more variance in the features.

A setup obtained the best overall performance with a Random Forest classifier applied on LDA topics in the first stage and an XGBoost classifier in the second stage. Its F1 weighted score was 1.24% higher than the best score achieved with a standalone setup.

The standard deviations obtained by the proposed approach are compared to the standard deviations of the two baselines in Table 10. The cascading classifiers have a higher overall standard deviation (and variance) than the text-based classifiers, meaning that these models will return results varying in a broader range when predicting future data. It can be beneficial to the final predictor due to the bias-variance tradeoff. Bias refers to assumptions in the learning algorithm that narrow the scope of what can be learned. It can accelerate learning and lead to stable results at the cost of the assumption differing from reality (Browlee, 2018). In this sense, models built from the cascading method tend to

learn more from new data while balancing the bias-variance tradeoff compared to metadata-based models. Considering that the small size of the dataset is a cause for high variance, fitting the model on more training data can reduce the overall variance.

Standard Deviation				
Classifier	**6 x 1**	**12 x 1**	**18 x 1**	**24 x 1**
Baseline 1 First stage (text-based)	0.030	0.037	0.039	0.035
Baseline 2 (metadata-based)	0.092	0.078	0.078	0.077
Cascade	0.058	0.050	0.055	0.063

Table 10 - Comparison of standard deviations between the proposed model and the baselines

Table 11 presents a comparison between the baselines and the proposed model according to the Area Under the Receiver Operating Characteristic Curve (ROC AUC) scores. The ROC plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various probability thresholds, i.e., the probability of belonging to the majority class. Also from this perspective, the cascade classifier shows superior performance better differentiating classes in different probability thresholds.

Area Under the Receiver Operating Characteristic Curve (ROC AUC) scores				
Classifier	**6 x 1**	**12 x 1**	**18 x 1**	**24 x 1**
Baseline 1 First stage (text-based)	0.784	0.820	0.770	0.825
Baseline 2 (metadata-based)	0.774	0.823	0.741	0.747
Cascade	0.788	0.861	0.828	0.850

Table 11 - Comparison of ROC AUC scores between the proposed model and the baselines

Taking the training/test split in which the cascade classifier obtains the best results compared to the baselines, Figure 9 presents the Receiver Operating Characteristic (ROC) for a range of probability threshold values on the classification algorithms' predictions. Two probability thresholds are marked as examples. At the discrimination point of 0.691 for the majority class, the cascade classifier has an FPR of 0.462, and the TPR is 0.988. For an FPR of 0.231 and a TPR of 0.802, the baseline classifier threshold is 0.898. The curve is the Recall (or True Positive Rate – TPR) plot versus the False Positive Rate (FPR). It is a standard technique for presenting a classifier performance over a range of tradeoffs between TPR (benefits) and FPR (costs). The optimal performance, or the optimal probability cut-point, combines the highest benefit with an acceptable cost, depending on the problem to which the model is applied. As shown in Table 12, for the sample under analysis, at the probability threshold of 0.691, the cascade classifier has a TPR of 0.988 and an FPR of 0.462. Since there are points where FPR and TPR are the same for both curves, the model of choice depends on the tradeoff that best suits the task under analysis.

TPR	FPR	Probability Threshold
0.111	0.000	0.990
0.309	0.077	0.978
0.593	0.154	0.958
0.802	0.231	0.898
0.864	0.308	0.825
0.988	**0.462**	**0.691**
1.000	0.846	0.167

Table 12 - TPR and FPR for different discrimination thresholds (sample split 6)

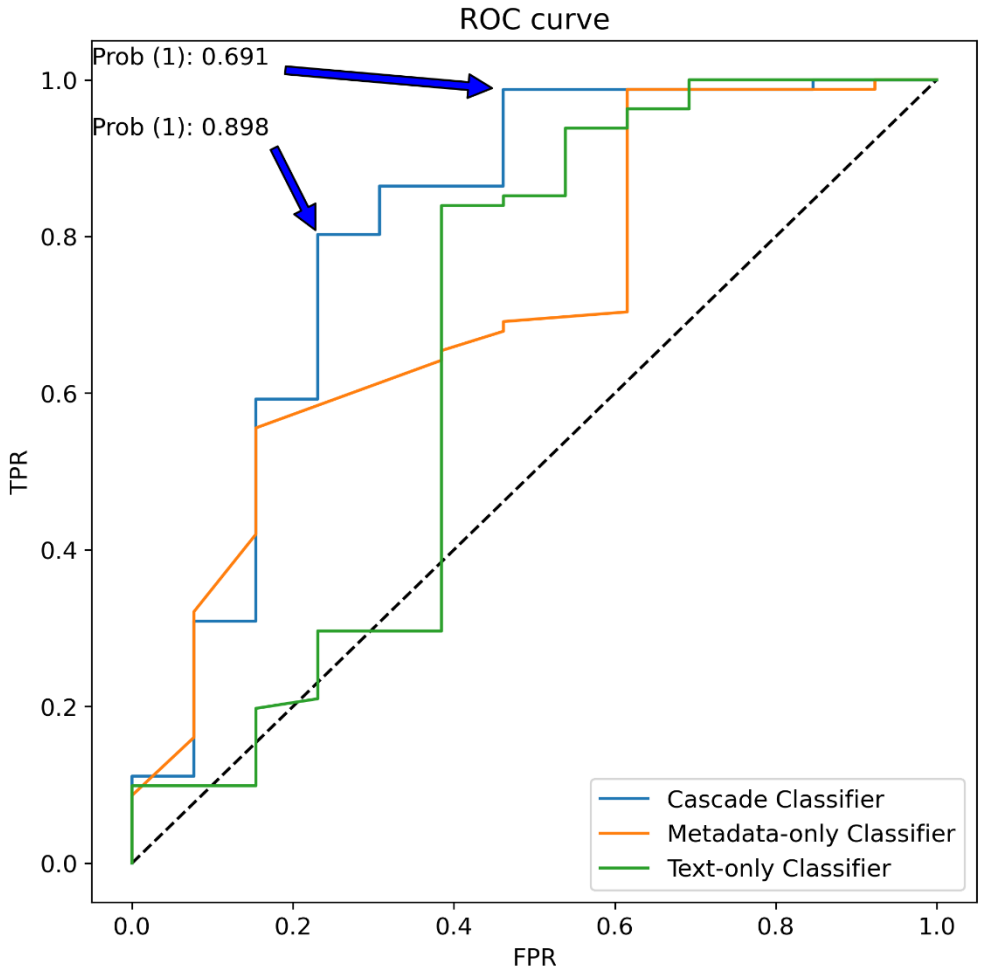


Figure 9 - ROC curves for cascade and baseline classifiers.

More substantial evidence of the improved performance of the cascade classifier comes when the ROC curves are cross-validated using all training/test time splits (Figure 10). The cascade classifier outperforms the baselines in most splits. It shows a higher mean value for the Area Under the ROC curve (AUC), indicating better discrimination power between the two classes.

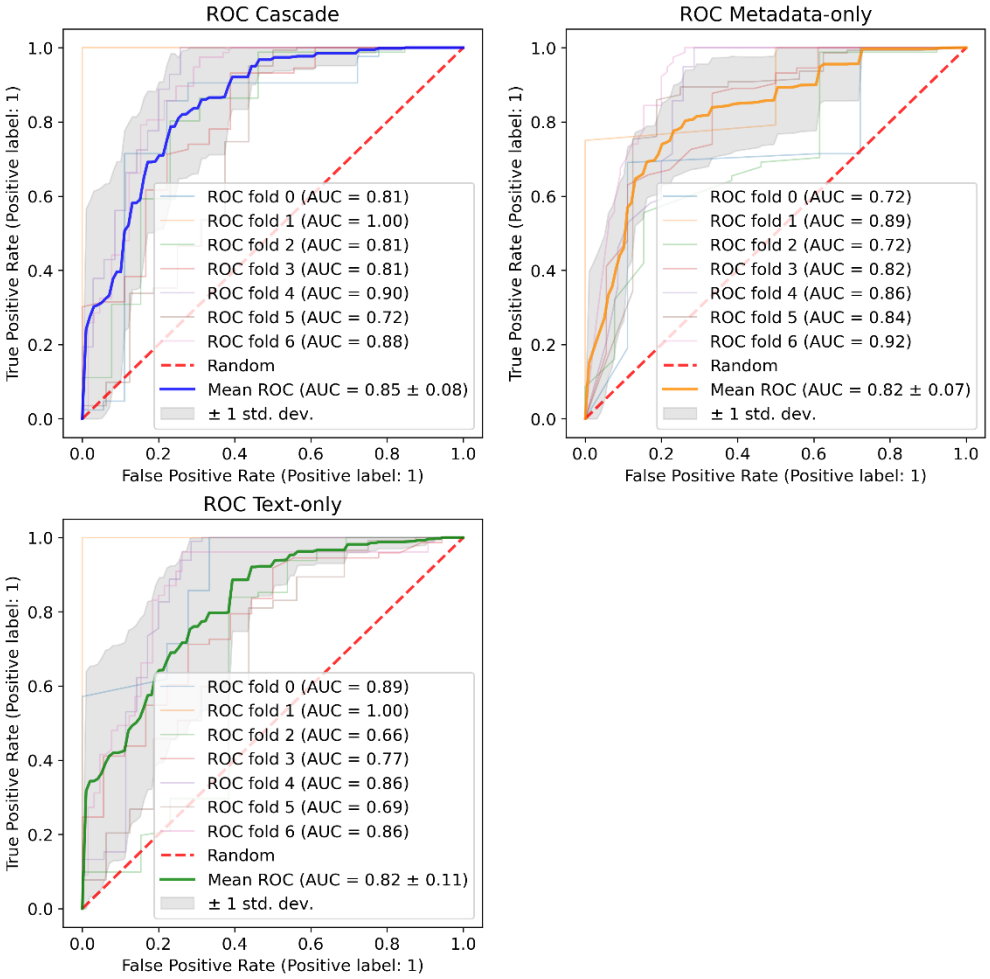


Figure 10 - Cross-validated ROC curves

The Precision-Recall curves of Figure 11, in their turn, show a comparison of the tradeoffs between Precision and Recall for different probability thresholds. Also, under this perspective, the cascade classifier outperforms the baselines with a better tradeoff behavior. We may observe a drop in Precision when the Recall achieves values closer to 0.7 in the metadata-only and text-only baselines. However, the cascade classifier maintains Precision with a smooth decay path as Recall increases.

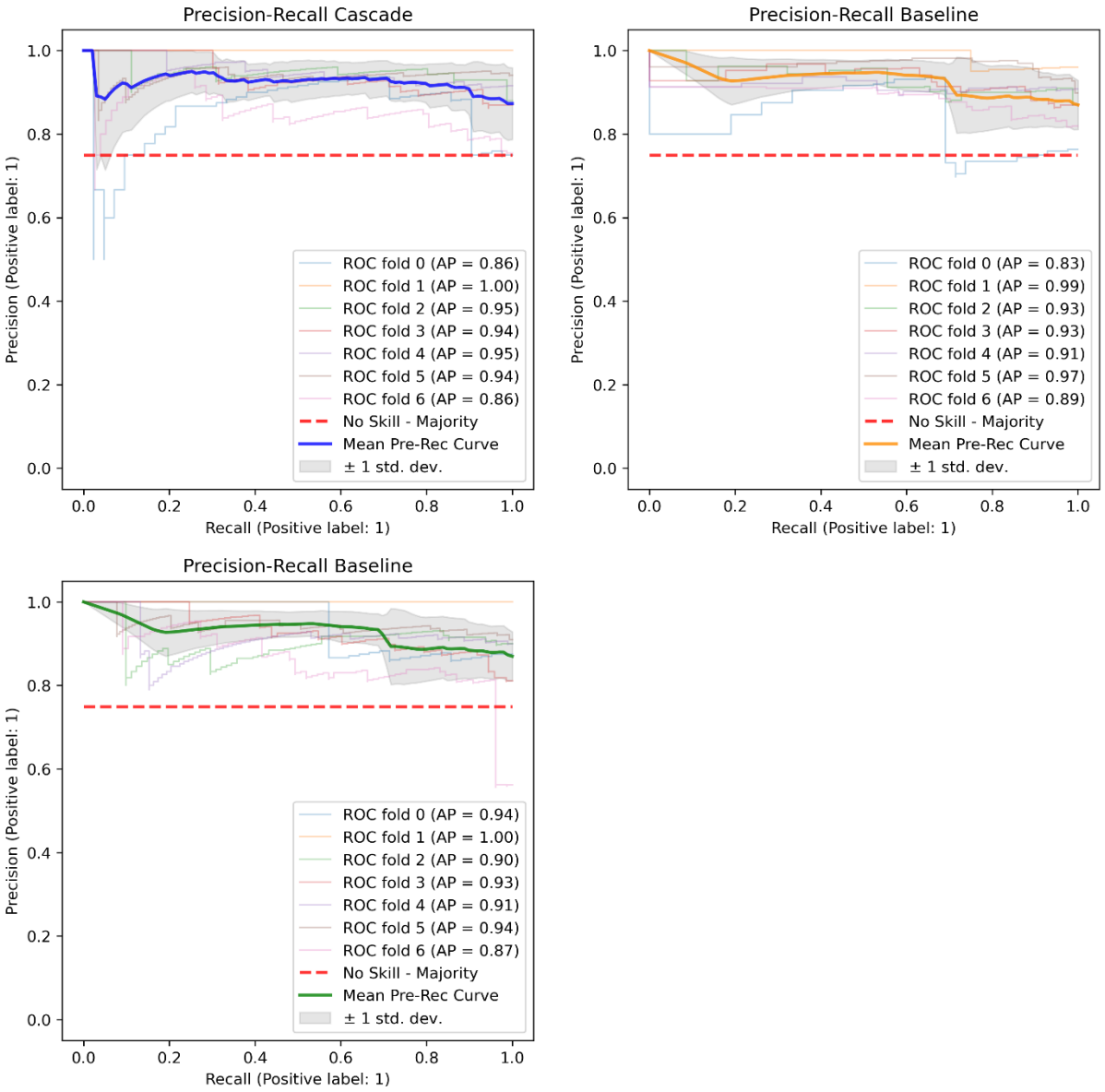


Figure 11 – Cross-validated Precision-Recall curves

These results are consistent with the work premise, i.e., the cascade classifier proposed in this work outperforms both the metadata-only and the text-only assemblies. The results also reveal that despite being a more challenging task to take the time dimension into account while maintaining out-of-sample applicability, it is possible to achieve reasonably high results, as the authors observed when working with this dataset. Additionally, the difference in performance observed in all comparisons using different algorithms shows the significance of evaluating this modeling in prediction tasks when text and metadata are available. For reducing variance, we emphasize the importance of continually incorporating new judgments into the data set.

Finally, we use SHAP values, an additive feature attribution method, to approximate each feature's contribution to the prediction and reverse-engineer the output, providing local explainability. The best explanation of a simple model is the model itself; it perfectly represents itself and is easy to

understand. For complex models, such as ensemble methods or deep networks, we cannot use the original model as its own best explanation because it is not easy to understand. Instead, we must use a simpler explanation model, which we define as any interpretable approximation of the original model (Lundberg & Lee, 2017). SHAP assigns each feature an importance value for a particular prediction. In our study's case, we used complex models as classifiers in the standalone stages. However, we preserved visibility on the features by using TF-IDF vectors or LDA topics in the first stage and metadata in the second stage.

We use the cascade classifier with the best overall performance as an example. The first stage is constituted by an LDA transformer whose output is consumed by a Random Forest classifier. LDA assumes that each document is represented by a distribution of a fixed number of topics, and each topic is a distribution of words. For a random violation whose prediction in both stages was "positive" or one, Figure 12 shows the influence of the most significant topics in the first stage prediction. The base value is the averaged predicted probability across all samples. The topics in red have a positive influence on the prediction (driving it beyond the base value). The blue arrows represent the topics that drive the prediction down. The bold value is the actual prediction for this sample.

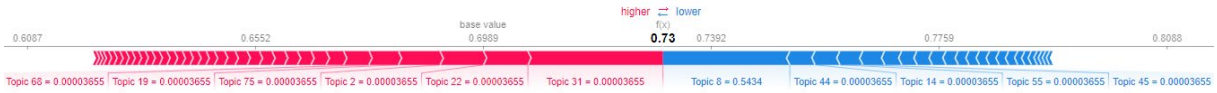


Figure 12 - SHAP values for the first stage prediction

The relevance of an n-gram to a topic, in its turn, is defined as:

$$r(w,k|\lambda) = \lambda \log \phi_{kw} + (1-\lambda) log \frac{\phi_{kw}}{p_{kw}}$$

Equation 5 – Relevance of an n-gram to a topic

In Equation 5, ϕ_{kw} is the probability of an n-gram w occurs in topic k and ϕ_kw/p_kw is the lift calculated by the n-gram's probability within a topic and its marginal probability across the entire corpus. The second term helps to discard globally frequent n-grams. Therefore, a lower λ gives more importance to n-grams' topic exclusivity. Using the pyLDAvis library (Mabey & English, 2015), we visualize the topics' interrelation and n-grams relevance. In Figure 13, by lowering λ to 0.5, we can find that topic 31 top-ranked stemmed n-grams related to ethics in the Portuguese language, e.g., "étic", "comit étic" and "códig étic".

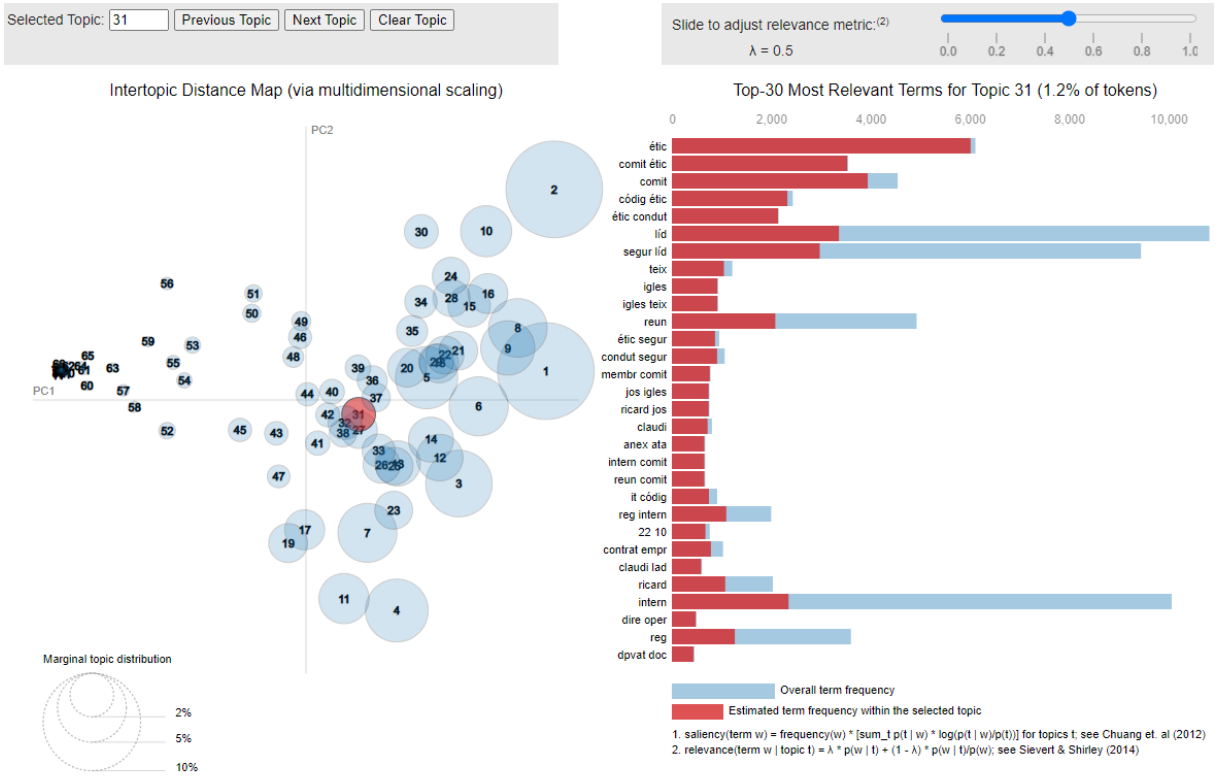


Figure 13 - Topics relatedness and n-grams relevance on topic 31

Next, we again use the SHAP values to estimate the importance of variables in the second stage, which employs an XGBoost Classifier. In this case, the time difference between the violation date and January 1st, 2000 ("daysFrom2000"), the infringed regulation ("5534"), and the previous stage prediction ("pred_subsistente") positively influenced the output. The time difference between the decision date and the violation date ("tempoAteJulgar") drove the prediction towards the negative output.

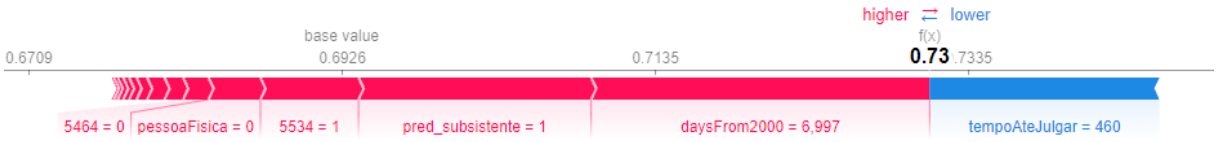


Figure 14 - SHAP values for the second stage prediction

4.2. STUDY 2

As previously stated in Section 3.4.2, the similarity measures obtained for the 1225 pairs formed from a sample of 50 randomly selected cases were compared with a gold standard composed of similarity scores awarded by legal experts. The comparison between models was made by employing the most context-relevant metrics, i.e., the mean Average Recall (mAR) as the primary metric and Recall@5 as a secondary metric. Table 13 presents the models that returned one of the three best results for any

combination between metric (mAR and Recall@5) and assessment set (Experts one, two, and three, and the Experts mean)[2]. The complete results set is presented in Appendix 4.

Documents with similarities greater than zero are more numerous when the average from individual evaluations is considered. The Recall, in turn, represents the fraction of relevant documents that are successfully retrieved, and the documents considered similar are those having a score higher than one. Hence, the mAR and Recall@5 results obtained on the Experts' mean assessment tend to be lower than those observed in the Experts' individual evaluations.

							Experts Mean		**Expert 1**		**Expert 2**		**Expert 3**	
Model	**Training Corpus**	**Evaluation Corpus**	**Topics**	**Dimensions**	**Algorithm**	**N-Grams**	**mAR**	**Recall@5**	**mAR**	**Recall@5**	**mAR**	**Recall@5**	**mAR**	**Recall@5**
W2V	F	C & R	-	100	CBOW	-	**0.705**	0.389	0.746	0.456	0.867	0.764	0.750	0.523
W2V	C	C	-	100	CBOW	-	**0.705**	0.357	0.743	0.431	0.856	0.710	**0.772**	0.526
W2V	C	C	-	100	SG	-	**0.701**	**0.393**	0.747	**0.486**	0.862	0.721	0.753	0.511
W2V	C & R	C & R	-	300	CBOW	-	0.698	0.358	0.733	0.436	0.863	0.725	0.747	**0.536**
W2V	F	F	-	100	CBOW	-	0.697	0.330	**0.761**	0.420	0.876	0.684	0.725	0.481
W2V	C	C	-	300	SG	-	0.696	0.391	0.741	**0.480**	0.858	**0.770**	**0.761**	**0.533**
W2V	F	F	-	300	CBOW	-	0.696	0.357	**0.755**	0.438	**0.878**	0.721	0.724	0.509
TF-IDF	F	F	-	-	-	Uni & Bi	0.695	0.385	**0.759**	0.457	**0.884**	0.752	0.733	0.493
W2V	F	C & R	-	100	SG	-	0.681	0.378	0.713	0.414	0.836	0.741	0.749	**0.530**
W2V	C & R	C & R	-	300	SG	-	0.681	0.381	0.730	**0.468**	0.873	**0.768**	0.741	0.528
LDA	S	S	40	-	-	-	0.678	0.330	0.686	0.356	0.799	0.537	**0.762**	0.443
TF-IDF	F	F	-	-	-	Bi & Tri	0.655	**0.395**	0.710	0.462	0.874	**0.769**	0.706	0.498
TF-IDF	F	S	-	-	-	Uni & Bi	0.650	0.334	0.712	0.411	**0.880**	0.683	0.702	0.452
TF-IDF	C	C	-	-	-	Bi & Tri	0.638	**0.395**	0.698	0.462	0.856	0.748	0.691	0.507

Table 13 - Experimental setups with top three scores by metric and expert scores set. The baseline setup is highlighted.

Considered the mAR from the Experts' mean scores, the Word2Vec models outperformed the baseline, and together with the TF-IDF models, obtained the best Recall scores overall. Among the tested setups, these are also the models that use the most granular form of text representation (words). It suggests that the combination of relatively small corpora, the presence of business-specific words, and the use of common words with particular meanings for the context can favor models that rely on more granular representations. Satisfactory results seem achievable using bag-of-words as in TF-IDF or adopting Word2Vec's Distributional Hypothesis, which suggests that words occurring in the same context tend to have similar meanings.

On this path, it is also significant that the pre-trained Word2Vec models, evaluated using the complete corpus without stemming, did not obtain good results and are not present in Table 13. Two factors can explain this phenomenon: the first is that stemming considerably reduces the word vector space,

[2] The three best results for each combination are marked in bold. The Full, Concepts, Concepts & Relations and Sentences corpora are represented by the F, C, C&R, and S labels. For continuous bag-of-words and skip-gram methods, we used the CBOW and SG tags.

making it easier to identify similar word stems. The second factor is that as the words in the Word2Vec templates are analyzed in context, pre-training a model without financial-sector-related texts may not capture the use of certain words with context-specific meanings. Also, suppose a general-purpose corpus does not match the domain's vocabulary (the exact words and words in the same senses). In that case, this cannot be overcome by presenting more data, which would move the word vectors towards standard rather than domain-specific meanings.

When considering the similarity average among experts, the Word2Vec models trained with 100 dimensions stood out and, when considered the individual evaluations, various configurations obtained good results, suggesting that the choice of the model (Word2Vec) prevails over the configuration concerning the corpus, the number of dimensions and the algorithm. An exception is made for using the sentence-based corpus, which did not yield the best Recall scores. This fact represents good news for applying these document similarity models in real scenarios since sentence-level training and evaluation embody a severe increase in computational cost compared to document-level training (about 100 times higher, in our dataset's case).

If we were interested in obtaining the best Precision, i.e., the fraction of relevant documents among the retrieved set, an LDA model trained with 80 topics showed the best results on all evaluation sets. The best scores were primarily obtained using Concepts as the textual representation, and the second-best results were achieved with Concepts and Relations.

These results show that LDA has a better performance in avoiding false positives, which can be explained by how documents are represented in this model family: a vector whose dimensions discriminate which topics occur in that text. For instance, a document can consist of 50% 'topic 1', 25% 'topic 2' and 25% 'topic 3'. LDA often results in document vectors with many zeros since documents are typically related to a limited number of topics. When two documents have no common topics, their cosine similarity equals zero. This pair will not produce a recommendation, differing from other models that rarely result in cosine similarities equal to zero due to the vector construction method. Ultimately, even though in other models the recommendations may include pairs with residual similarities, in LDA these same pairs will have similarities equal to zero and will not be presented. This aspect can be equalized by adopting a minimum similarity (threshold) to recommend a pair as similar. However, the value for this threshold is not known a priori and will largely depend on the model and corpus.

It is also worth noting that the Skip-Gram model with 300 dimensions, trained and evaluated with concepts (nouns), obtained very satisfactory results for Recall@5 concerning the individual Experts' evaluations. Therefore, for a business application in which each document under analysis produced five similar cases, this should be the configuration of choice. The findings, especially when measured the Recall@5, justify using Concepts extracted from the document as textual representation instead of the full text. This approach significantly increased the performance of Word2Vec assemblies, and the same effect was observed when measuring the Precision with LDA models. Compared to the baseline by Recall@5, this Doc2Vec setup (Skip-Gram, 300 dimensions, using Concepts) obtained gains of 5.0%, 2.4%, and 8.1% over the baseline model for Experts one, two, and three respectively (5.2% on average).

Finally, the performance of Doc2Vec and BM25 models was noticeably low in our experiments. Although, by construction, models based on neural networks as Doc2Vec and Word2Vec require large volumes of training data, the dataset used in this study seems to have been sufficient to obtain good

results using Word2Vec, but not with Doc2Vec models, reinforcing the granularity hypothesis mentioned before. Considering that it was impossible to extract paragraphs or sections from the documents, which could contribute with document-level contextual information for the Doc2Vec models, the Word2Vec assemblies benefited from using local contextual information based on the words neighborhood.

Concerning models based on the BM25 ranking function, we emphasize that such models have characteristics that differ and seek to improve the usual TF-IDF measures. One example is the term frequency saturation parameter, decreasing the gain when the same term occurs more times in the same document. These models also consider the ratio between the document length and the average document length, implying that long documents with occurrences of the query will have a lower score than shorter texts with the same number of occurrences. However, in our dataset's case, such characteristics seem to have penalized the performance of BM25-based assemblies.

5. CONCLUSIONS

Our first study stressed NLP and ML techniques through a highly unmapped methodology (cascade generalization) for predicting binary administrative decisions, which is also appliable to judicial lawsuits. The possible barrier for using this kind of predictor in a real-world task, represented by the drop in performance brought by the need for considering the time dimension, was overcome by applying an assembly that can provide more information to the algorithms. Still, using only one document from the proceeding, neither external variables nor any particular assumption about the data, the model might function as a reproducible baseline for researchers and organizations.

No less critical for the model's applicability is the possibility of identifying variables that influence the classification of violations. In this sense, using the state-of-the-art method to identify the contributions of the variables in both stages and preserving the visibility of the variables extracted from the text, it was possible to provide the model with local explainability.

Despite people not expecting a future in which trained machines can substitute decision-makers, this study demonstrates how an ensemble model can improve existing decision-support models. As previously stated, the more accurate the prediction model, the better the advice it provides to administrative courts, offering agility and consistency in decisions by facilitating jurisprudence usage.

The second study, intended to facilitate the violations' human analysis by identifying and suggesting similar cases, showed that it was possible to identify similar cases with the combination of techniques. Word2Vec models worked best in this relatively small dataset with context-specific word usage. However, other techniques improved the results, such as extracting concepts (nouns) and relationships (verbs) from the text. The vectorization of concepts was the best option for an application that suggested five similar cases. Ultimately, the Word2Vec models consistently outperformed the TF-IDF baseline model both on mAR and Recall@5 measures. These results represent robust evidence that different textual variable extraction techniques deserve to be tested for performance improvement and corroborating the effectiveness of vectorization employing neural networks. From a law court applicability perspective, these results demonstrate that using representative parts of the text to implement a document similarity model using Word2Vec can effectively identify precedents and contribute to the agility of analysis and decision-making.

Both studies' results confirm that it is possible to improve administrative court decisions' efficiency and consistency. The decisions predicting study showed objective and robust evidence of reliably outperforming the most used models in important metrics by a gain rate that makes its implementation attractive. The predictive performance was also probed under generalization and out-of-sample applicability conditions. The case similarity study, in turn, revealed that the widely adopted Word2Vec and TF-IDF embedding models are, respectively, the best and the second-best options identifying similar documents in the studied context. Furthermore, a straightforward text representation technique (concepts extraction) soundly improved the results, obtaining the top Recall@5 scores in all scenarios. Word2Vec is also preferable over TF-IDF models because they can be updated with new vocabulary and more documents, scaling for increasing datasets. The latter, in their turn, need to be re-fitted every time new documents are to be added. Finally, good results were obtained in confrontation with a large sample of 1225 decision pairs, independently evaluated by three legal experts. To the author's knowledge, this is the most significant sample used in studies of this

nature. Among the previous works in the literature review, the most significant sample had 47 pairs of texts (Mandal et al., 2017).

6. LIMITATIONS AND RECOMMENDATIONS FOR FUTURE WORKS

The decision outcome prediction study paves the way for future works to investigate whether the proposed approach is the right choice for multiclass problems. Furthermore, larger datasets and ensembles of predictions from two-stage models may be explored to introduce bias and decrease the variance. The possible contribution of using semantic analysis for extracting forecasts in the first stage is another topic, in the author's point of view, that deserves further investigation. There is still a vast field to investigate models that offer global explainability so that humans can easily interpret how classifications are carried out. Like others, this study is not without limitations. Due to restricted data available, further tests should be conducted to validate the model's stability. Reproducing these results may depend on the organization and the type of problems addressed. Furthermore, employing the stemming technique to reduce the feature space in the first stage brought adverse consequences to local explainability from topics and n-grams.

For computing document similarities, the dataset lacked the necessary standardization. Hence, we could not experiment with more document representation techniques, e.g., paragraphs or sections according to a known structure. Additionally, despite not improving the results in the study conducted by Mandal, the BERT models and the document summarization methods proposed by Galgani (MyScore) and Mandal (PScore) may be further investigated. Another possible direction is to examine the applicability of the semantic similarity computing model proposed and tested with sentences by Zang. Law2Vec, in its turn, another Word2Vec model pre-trained on a large legal corpus and cited by Mandal et al. (2021), could not be employed in this study because it has only been trained on English-written texts.

Both studies are meant to be applied as real-time query systems, recommending decision outcomes and similar cases when a new violation case and the respective initial document are analyzed. In this scenario, predicting decisions only one week in advance, for instance, may require a well-suited pipeline for data integration and training. In both cases, embedding models need to be pre-trained and updated promptly. Therefore, future studies should also focus on understanding how these models could be deployed in production environments.

7. BIBLIOGRAPHY

Aletras, N., Tsarapatsanis, D., Preoţiuc-Pietro, D., & Lampos, V. (2016). Predicting judicial decisions of the European court of human rights: A natural language processing perspective. *PeerJ Computer Science*, *2016*(10), 1–19. https://doi.org/10.7717/peerj-cs.93

Bhattacharya, P., Mehta, P., Ghosh, K., Ghosh, S., Pal, A., Bhattacharya, A., & Majumder, P. (2020). Overview of the FIRE 2020 AILA track: Artificial intelligence for legal assistance. *CEUR Workshop Proceedings*, *2826*, 1–11.

Bibal, A., Lognoul, M., De Streel, A., & Frénay, B. (2021). Legal requirements on explainability in machine learning. *Artificial Intelligence and Law*, *29*(2), 149–169. https://doi.org/10.1007/s10506-020-09270-4

Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media. https://doi.org/https://dl.acm.org/doi/10.5555/1717171

Brill, E. (1992). A simple rule-based part of speech tagger. Em *Proceedings of the third conference on Applied natural language processing* -. Association for Computational Linguistics. https://doi.org/10.3115/974499.974526

Browlee, J. (2018). *How to Reduce Variance in a Final Machine Learning Model*. Machine Learning Mastery. https://machinelearningmastery.com/how-to-reduce-model-variance/

Chalkidis, I. (2018). *Law2Vec: Legal Word Embeddings*. https://archive.org/details/Law2Vec

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357.

Chen, D. L., & Eagel, J. (2017). Can Machine Learning Help Predict the Outcome of Asylum Adjudications? *Proceedings of the International Conference on Artificial Intelligence and Law*, 237–240. https://doi.org/10.1145/3086512.3086538

Chen, L. (2009). Curse of Dimensionality. Em *Encyclopedia of Database Systems* (pp. 545–546). Springer US. https://doi.org/10.1007/978-0-387-39940-9_133

Chen, T., & Guestrin, C. (2016). XGBoost. Em *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. https://doi.org/10.1145/2939672.2939785

Cochran, W. G. (1977). *Sampling Techniques, 3rd Edition*. John Wiley & Sons, Ltd.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Computing Research Repository (CoRR)*. https://arxiv.org/abs/1810.04805v2

Fenergo. (2020). *AML, KYC & Sanctions Fines for Global Financial Institutions Top $36 Billion Since Financial Crisis*. Fenergo. https://www.fenergo.com/press-releases/aml-kyc-sanctions-fines-for-global-financial-institutions-top-36-billion-since-financial-crisis/

Fonseca, E. R., G Rosa, J. L., & Aluísio, S. M. (2015). Evaluating word embeddings and a revised corpus for part-of-speech tagging in Portuguese. *Journal of the Brazilian Computer Society*, *21*(1). https://doi.org/10.1186/s13173-014-0020-x

Gama, J., & Brazdil, P. (2000). Cascade Generalization. *Machine Learning*, *41*(3), 315–343.

https://doi.org/10.1023/A:1007652114878

Gao, J., Ning, H., Sun, H., Liu, R., Han, Z., Kong, L., & Qi, H. (2019). FIRE2019@AILA: Legal retrieval based on information retrieval model. *CEUR Workshop Proceedings*, *2517*(December 2019), 64–69.

Hartmann, N., Fonseca, E., Shulby, C., Treviso, M., Rodrigues, J., & Aluisio, S. (2017). *Portuguese Word Embeddings: Evaluating on Word Analogies and Natural Language Tasks*. http://arxiv.org/abs/1708.06025

Herman-Saffar, O. (2020). *Time Based Cross Validation*. Towards Data Science. https://towardsdatascience.com/time-based-cross-validation-d259b13d42b8

IAIS. (2017). *Insurance Core Principles*. https://www.iaisweb.org/file/69922/insurance-core-principles-updated-november-2017

Katz, D. M., Bommarito, M. J., & Blackman, J. (2017). A general approach for predicting the behavior of the Supreme Court of the United States. *PLOS ONE*, *12*(4), e0174698. https://doi.org/10.1371/journal.pone.0174698

Kumar, S., Reddy, P. K., Reddy, V. B., & Singh, A. (2011). Similarity analysis of legal judgments. *Compute 2011 - 4th Annual ACM Bangalore Conference*, 3–6. https://doi.org/10.1145/1980422.1980439

Kumar, S., Reddy, P. K., Reddy, V. B., & Suri, M. (2013). Finding similar legal judgements under common law system. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *7813 LNCS*(July 2016), 103–116. https://doi.org/10.1007/978-3-642-37134-9_9

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Proceedings of the 31st international conference on neural information processing systems*, *December*, 4768–4777.

Lv, Y., & Zhai, C. (2011). Lower-Bounding Term Frequency Normalization. *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, 7–16. https://doi.org/10.1145/2063576.2063584

Mabey, B., & English, P. (2015). *pyLDAvis* (2.1.2). https://pyldavis.readthedocs.io/en/latest/

Mandal, A., Chaki, R., Saha, S., Ghosh, K., Pal, A., & Ghosh, S. (2017). Measuring similarity among legal court case documents. *ACM International Conference Proceeding Series*, 1–9. https://doi.org/10.1145/3140107.3140119

Mandal, A., Ghosh, K., Ghosh, S., & Mandal, S. (2021). Unsupervised approaches for measuring textual similarity between legal court case reports. *Artificial Intelligence and Law*, *29*(3), 417–451. https://doi.org/10.1007/s10506-020-09280-2

Medvedeva, M., Vols, M., & Wieling, M. (2020). Using machine learning to predict decisions of the European Court of Human Rights. *Artificial Intelligence and Law*, *28*(2), 237–266. https://doi.org/10.1007/s10506-019-09255-y

Nason, S. (2018). *Administrative justice can make countries fairer and more equal – if it is implemented properly*. The Conversation. https://theconversation.com/administrative-justice-can-make-countries-fairer-and-more-equal-if-it-is-implemented-properly-108238

O'Sullivan, C., & Beel, J. (2019). Predicting the outcome of judicial decisions made by the European court of human rights. *CEUR Workshop Proceedings*, *2563*, 272–283.

Orengo, V. M., & Huyck, C. (2001). A stemming algorithm for the portuguese language. *Proceedings Eighth Symposium on String Processing and Information Retrieval*, 186–193. https://doi.org/10.1109/spire.2001.989755

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., VanderPlas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). *Scikit-learn: Machine Learning in Python* (pp. 2825–2830). Journal of Machine Learning Research. https://scikit-learn.org/stable/

Pillai, V. G., & Chandran, L. R. (2020). Verdict prediction for indian courts using bag of words and convolutional neural network. *Proceedings of the 3rd International Conference on Smart Systems and Inventive Technology, ICSSIT 2020*, 676–683. https://doi.org/10.1109/ICSSIT48917.2020.9214278

Ranera, L. T. B., Solano, G. A., & Oco, N. (2019). Retrieval of Semantically Similar Philippine Supreme Court Case Decisions using Doc2Vec. *2019 International Symposium on Multimedia and Communication Technology (ISMAC)*, 1–6. https://doi.org/10.1109/ISMAC.2019.8836165

Richardson, L. (2007). *BeautifulSoup*. https://www.crummy.com/software/BeautifulSoup/

Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, *3*(4), 333–389. https://doi.org/10.1561/1500000019

Ruger, T. W., Kim, P. T., Martin, A. D., & Quinn, K. M. (2004). The Supreme Court Forecasting Project: Legal and Political Science Approaches to Predicting Supreme Court Decisionmaking. *Columbia Law Review*, *104*(4), 1150–1210. https://doi.org/10.2307/4099370

Shinyama, Y., Guglielmetti, P., & Marsman, P. (2019). *pdfminer.six*. https://github.com/pdfminer/pdfminer.six

Sivaranjani, N., Jayabharathy, J., & Teja, P. C. (2021). Predicting the supreme court decision on appeal cases using hierarchical convolutional neural network. *International Journal of Speech Technology*, *24*(3), 643–650. https://doi.org/10.1007/s10772-021-09820-4

Statista. (2020). *Global insurance industry - statistics & facts*. https://www.statista.com/topics/6529/global-insurance-industry/

SUSEP. (2020a). *8º Relatório de Análise e Acompanhamento dos Mercados Supervisionados*. 1–24. http://www.susep.gov.br/menuestatistica/SES/relat-acomp-mercado-2020.pdf

SUSEP. (2020b). *Brokers Statistics*. https://www2.susep.gov.br/safe/Corretores/estatisticas

Thenmozhi, D., Kannan, K., & Aravindan, C. (2017). A Text Similarity Approach for Precedence Retrieval from Legal Documents. *FIRE (working notes)*, 90–91. http://ceur-ws.org/Vol-2036/T3-9.pdf

Theodoridis, S. (2020). *Machine Learning: A Bayesian and Optimization Perspective, 2nd Edition*. Elsevier.

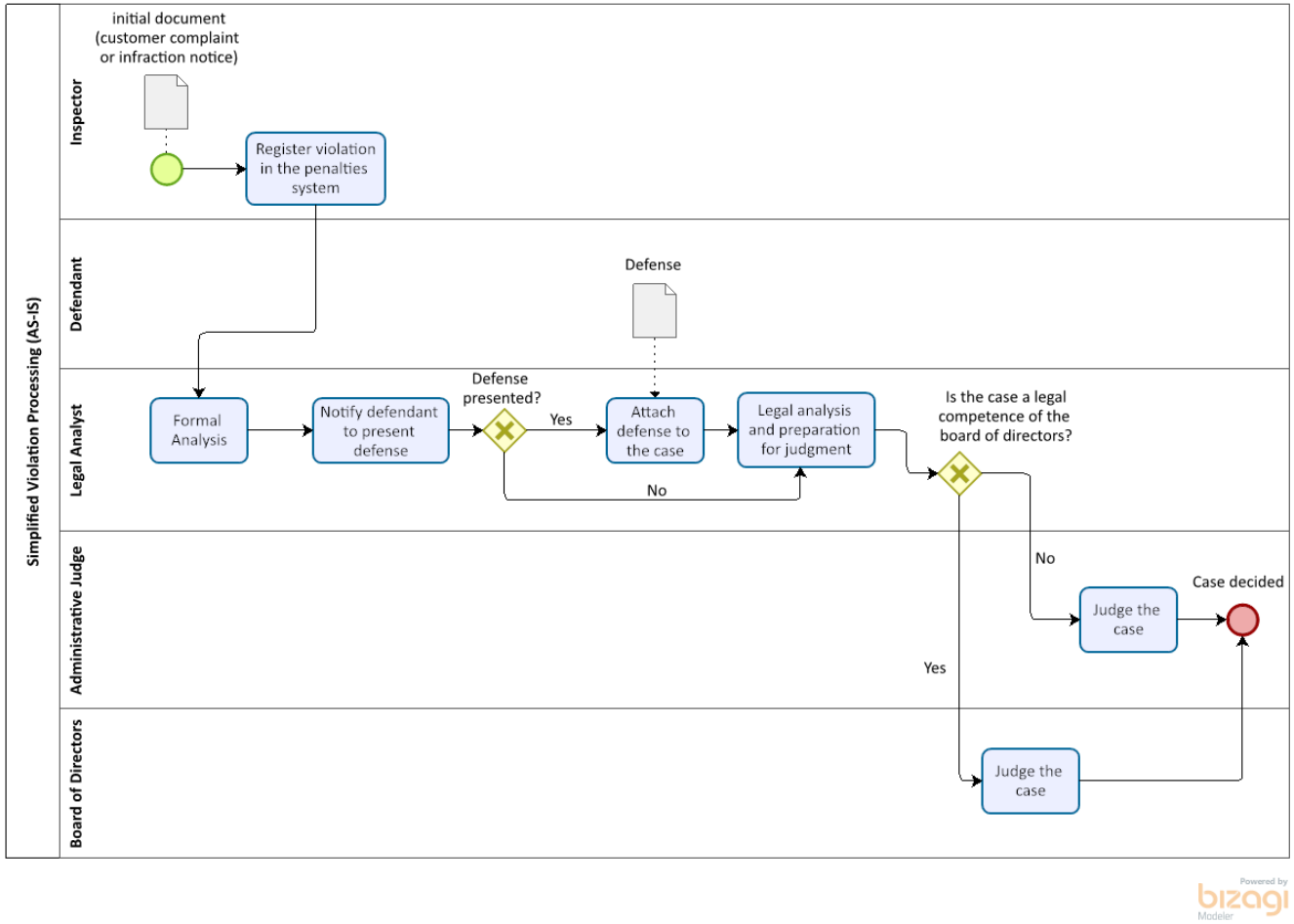
Whissell, J. S., & Clarke, C. L. A. (2013). Effective Measures for Inter-Document Similarity Categories and Subject Descriptors. *22nd ACM international conference on Information & Knowledge*

Management, 1361–1370.

Zhao, Z., Ning, H., Liu, L., Huang, C., Kong, L., & Han, Y. (2019). FIRE2019 @ AILA : Legal Information Retrieval Using Improved BM25. *FIRE (working notes)*, *December 2019*, 12–15.

8. APPENDICES

8.1. APPENDIX 1 - VIOLATION ANALYSIS BUSINESS PROCESS



8.2. APPENDIX 2 – INFRACTION NOTICE ENGLISH TRANSLATION

Ministry of Finance
SUPERINTENDENCY OF PRIVATE INSURANCE

GENERAL COORDINATION OF DIRECT OVERSIGHT

INFRACTION NOTICE
SUSEP/DIFIS/CGFIS/COSU2/DISU5 NO. 2/14

In the exercise of our attributions, we verified during the on-site inspection activities carried out at (...) for which we were assigned through the DESIGNATION TERM SUSEP/DIFIS/CGFIS/COSU1/ NO. (…) conduct identified as an administrative offense. Considering that after verification and according to the documentation attached to the process, it was not possible to identify the natural person responsible for the conduct identified as an administrative offense, we present a proposal to file the competent Sanctioning Administrative Proceeding - REPRESENTATION PAS - and the application of the appropriate administrative sanction, as described below and shown in the attached documents.

1. **ISSUING A POLICY/INSURANCE CERTIFICATE IN DISAGREEMENT WITH THE LEGISLATION**

The punishable fact: *not registering in the policy and the individual insurance certificate of rural pledge type, the information that the insured property is offered as a guarantee of a rural credit operation.*

The article no. 3 of Circular SUSEP No. 308/2005, which addresses the rural pledge insurance, determines that:

> *Art. 3 The Insurance Companies must register in the policy that the insured property, directly related to agricultural, livestock, aquaculture, or forestry activities, is offered in a guarantee of rural credit operation.*

As will be shown below, the (…) issued a collective policy and the respective individual certificates of insurance of the rural pledge type without the mandatory minimum information. On the materiality of the infraction: It was requested to (…) the certificates (…).

8.3. APPENDIX 3 – CLASSIFICATION TASKS TERMINOLOGY AND METRICS USED

Table 14 presents the usual terminology for classification tasks, and Table 15 contains the classification model performance metrics mentioned in this work.

		ACTUAL	
		TRUE **+, 1**	**FALSE** **-, 0**
PREDICTED	**TRUE** **+, 1**	True Positive (TP)	False Positive (FP)
	FALSE **-, 0**	False Negative (FN)	True Negative (TN)
	CLASS COUNT	P = TP + FN	N = FP + TN

Table 14 - Terminology for classification tasks.

Precision **Positive predictive value (PPV)**	$\frac{TP}{TP + FP}$
Recall **True Positive Rate (TPR)**	$\frac{TP}{TP + FN}$
False Positive Rate (FPR)	$\frac{FP}{FP + TN}$
F1 Score	$2 \times \frac{precision \times recall}{precision + recall}$
Weighted F1 Score	$\frac{\sum_{i=1}^{n} w_i F_{1_i}}{\sum_{i=1}^{n} w_i}$ $\{F_{1_1}, F_{1_2}, \dots, F_{1_n}\}$ is the set of F1 scores for each class, and $\{w_1, w_2, \dots, w_n\}$ is the set of class weights.

Table 15 - Statistical measures of performance for binary classification.

8.4. APPENDIX 4 – DETAILED RESULTS OBTAINED IN THE SECOND STUDY

The following table presents the detailed results[3] for the second study, according to the experimental setup described in Section 3.4.3. The three best results for each metric combination (mAR, Recall@5, mAP, and Precision@5) and assessment set (Experts one, two, and three, and the Experts mean) are highlighted in bold.

							Experts Mean				Expert 1				Expert 2				Expert 3			
Model	**Training Corpus**	**Evaluation Corpus**	**Topics**	**Dimensions**	**Algorithm**	**N-Grams**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**
W2V	F	C & R	-	100	CBOW	-	**0.705**	0.389	0.228	**0.436**	0.746	0.456	0.200	0.420	0.867	0.764	0.102	0.252	0.750	0.523	0.151	0.300
W2V	C	C	-	100	CBOW	-	**0.705**	0.357	0.225	0.400	0.743	0.431	0.191	0.372	0.856	0.710	0.099	0.236	**0.772**	0.526	0.154	0.308
W2V	C	C	-	100	SG	-	**0.701**	**0.393**	0.225	0.432	0.747	**0.486**	0.199	0.420	0.862	0.721	0.106	0.260	0.753	0.511	0.150	0.312
W2V	F	C & R	-	300	CBOW	-	0.701	0.370	0.222	0.408	0.740	0.449	0.193	0.392	0.859	0.674	0.100	0.232	0.753	0.509	0.148	0.288
W2V	F	F	-	300	SG	-	0.699	0.359	0.224	0.404	0.751	0.435	0.198	0.384	0.871	0.731	0.108	0.264	0.735	0.501	0.150	0.308
W2V	F	C	-	100	CBOW	-	0.699	0.371	0.223	0.412	0.734	0.430	0.192	0.396	0.869	0.730	0.101	0.240	0.756	0.498	0.147	0.276
W2V	F	C	-	300	CBOW	-	0.699	0.378	0.221	0.416	0.729	0.452	0.191	0.400	0.865	0.716	0.101	0.244	0.760	0.509	0.148	0.288
W2V	C & R	C & R	-	300	CBOW	-	0.698	0.358	0.220	0.408	0.733	0.436	0.190	0.388	0.863	0.725	0.102	0.244	0.747	**0.536**	0.150	0.312
W2V	F	F	-	100	CBOW	-	0.697	0.330	0.218	0.364	**0.761**	0.420	0.196	0.356	0.876	0.684	0.101	0.220	0.725	0.481	0.141	0.264
W2V	C	C	-	300	SG	-	0.696	0.391	0.223	0.432	0.741	**0.480**	0.197	0.412	0.858	**0.770**	0.106	**0.272**	**0.761**	**0.533**	0.151	**0.320**
W2V	F	F	-	300	CBOW	-	0.696	0.357	0.220	0.400	**0.755**	0.438	0.195	0.388	**0.878**	0.721	0.102	0.240	0.724	0.509	0.143	0.292
W2V	C	C	-	300	CBOW	-	0.695	0.355	0.220	0.404	0.737	0.422	0.189	0.372	0.847	0.684	0.098	0.228	0.745	0.509	0.147	0.296
TF-IDF	F	F	-	-	-	Uni & Bi	0.695	0.385	0.227	0.424	**0.759**	0.457	0.203	0.404	**0.884**	0.752	0.105	0.248	0.733	0.493	0.146	0.288
W2V	C & R	C & R	-	100	CBOW	-	0.694	0.366	0.221	0.420	0.734	0.450	0.193	0.408	0.860	0.726	0.102	0.248	0.750	0.527	0.150	0.308
W2V	F	C & R	-	300	SG	-	0.691	0.373	0.223	0.420	0.728	0.437	0.194	0.400	0.866	0.765	0.107	**0.272**	0.748	0.516	0.153	0.312

[3] The Full, Full without stemming, Concepts, Concepts & Relations and Sentences corpora are represented by the F, F (NS), C, C&R, and S labels. For continuous bag-of-words and skip-gram methods, we used the CBOW and SG tags. Unigrams, Bigrams and Trigrams are represented respectively by Uni, Bi and Tri.

							Experts Mean				Expert 1				Expert 2				Expert 3			
Model	**Training Corpus**	**Evaluation Corpus**	**Topics**	**Dimensions**	**Algorithm**	**N-Grams**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**
W2V	F	F	-	100	SG	-	0.691	0.355	0.217	0.400	0.737	0.435	0.193	0.388	0.856	0.742	0.103	0.260	0.733	0.505	0.145	0.300
W2V	F	C	-	300	SG	-	0.685	0.355	0.217	0.392	0.718	0.423	0.187	0.364	0.859	0.748	0.105	0.256	0.752	0.520	0.151	0.304
TF-IDF	C & R	C & R	-	-	-	Uni & Bi	0.684	0.376	0.225	0.408	0.744	0.443	0.199	0.388	0.867	0.710	0.104	0.236	0.724	0.492	0.145	0.280
TF-IDF	C	C	-	-	-	Uni & Bi	0.681	0.391	0.226	0.432	0.741	0.467	0.201	0.416	0.875	0.758	0.105	0.256	0.731	0.512	0.147	0.300
W2V	F	C & R	-	100	SG	-	0.681	0.378	0.215	0.404	0.713	0.414	0.186	0.376	0.836	0.741	0.103	0.264	0.749	**0.530**	0.149	0.308
W2V	F	C	-	100	SG	-	0.681	0.355	0.212	0.368	0.710	0.399	0.183	0.340	0.838	0.714	0.101	0.244	0.750	0.519	0.147	0.292
W2V	C & R	C & R	-	300	SG	-	0.681	0.381	0.218	0.424	0.730	**0.468**	0.195	0.416	0.873	**0.768**	0.109	**0.272**	0.741	0.528	0.151	0.316
LDA	S	S	20	-	-	-	0.680	0.245	0.182	0.228	0.683	0.242	0.144	0.184	0.761	0.433	0.063	0.104	0.725	0.353	0.115	0.156
LDA	S	S	80	-	-	-	0.680	0.310	0.197	0.316	0.703	0.366	0.164	0.288	0.819	0.621	0.083	0.180	0.742	0.422	0.130	0.224
D2V	F	F	-	200	-	-	0.680	0.326	0.221	0.356	0.705	0.379	0.192	0.320	0.855	0.676	0.105	0.224	0.742	0.491	0.149	0.276
W2V	C & R	C & R	-	100	SG	-	0.679	0.370	0.221	0.420	0.726	0.437	0.196	0.408	0.858	0.762	0.107	0.268	0.742	0.525	0.152	0.312
LDA	S	S	40	-	-	-	0.678	0.330	0.193	0.292	0.686	0.356	0.154	0.240	0.799	0.537	0.078	0.140	**0.762**	0.443	0.134	0.212
TF-IDF	F	C & R	-	-	-	Uni & Bi	0.676	0.375	0.222	0.404	0.735	0.444	0.198	0.388	0.869	0.727	0.103	0.240	0.718	0.487	0.142	0.276
D2V	C & R	C & R	-	200	-	-	0.675	0.344	0.222	0.372	0.703	0.395	0.194	0.352	0.852	0.667	0.104	0.232	0.750	0.461	0.149	0.272
D2V	C & R	C & R	-	100	-	-	0.674	0.323	0.222	0.360	0.703	0.369	0.193	0.332	0.854	0.667	0.104	0.228	0.744	0.457	0.148	0.268
TF-IDF	F	C	-	-	-	Uni & Bi	0.673	0.360	0.221	0.384	0.731	0.436	0.197	0.376	0.877	0.728	0.103	0.228	0.725	0.473	0.143	0.260
D2V	F	F	-	100	-	-	0.671	0.356	0.220	0.388	0.698	0.424	0.194	0.364	0.855	0.666	0.106	0.228	0.732	0.491	0.147	0.284
D2V	C & R	C & R	-	300	-	-	0.670	0.347	0.223	0.388	0.700	0.390	0.194	0.356	0.852	0.681	0.105	0.244	0.742	0.479	0.152	0.292
D2V	F	F	-	300	-	-	0.669	0.340	0.218	0.364	0.693	0.393	0.190	0.332	0.861	0.690	0.106	0.224	0.733	0.485	0.146	0.268
W2V	F	S	-	300	CBOW	-	0.667	0.320	0.206	0.340	0.695	0.355	0.180	0.320	0.856	0.629	0.097	0.204	0.725	0.465	0.136	0.256
W2V	S	S	-	100	CBOW	-	0.667	0.323	0.208	0.332	0.695	0.364	0.183	0.320	0.851	0.650	0.096	0.208	0.729	0.457	0.138	0.244
W2V	F	S	-	100	CBOW	-	0.664	0.320	0.207	0.348	0.694	0.379	0.181	0.336	0.849	0.649	0.095	0.208	0.721	0.453	0.136	0.256
W2V	S	S	-	300	CBOW	-	0.662	0.306	0.207	0.336	0.694	0.366	0.183	0.328	0.853	0.632	0.097	0.204	0.726	0.424	0.138	0.236

							Experts Mean				Expert 1				Expert 2				Expert 3			
Model	**Training Corpus**	**Evaluation Corpus**	**Topics**	**Dimensions**	**Algorithm**	**N-Grams**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**
W2V	S	S	-	300	SG	-	0.662	0.334	0.209	0.368	0.701	0.408	0.187	0.356	0.846	0.683	0.100	0.228	0.720	0.475	0.139	0.268
D2V	F	C & R	-	200	-	-	0.661	0.387	0.226	0.428	0.688	0.446	0.193	0.392	0.840	0.711	0.105	0.256	0.752	0.516	0.154	0.308
D2V	C	C	-	300	-	-	0.660	0.348	0.216	0.364	0.694	0.400	0.189	0.340	0.857	0.700	0.102	0.236	0.745	0.477	0.146	0.272
D2V	F	C & R	-	300	-	-	0.660	0.378	0.224	0.420	0.679	0.420	0.188	0.380	0.851	0.706	0.104	0.252	0.757	0.507	0.156	0.304
W2V	F	S	-	100	SG	-	0.656	0.320	0.206	0.356	0.697	0.382	0.184	0.348	0.840	0.665	0.099	0.228	0.714	0.443	0.135	0.252
W2V	S	S	-	100	SG	-	0.656	0.337	0.209	0.376	0.697	0.413	0.186	0.364	0.848	0.672	0.100	0.220	0.717	0.468	0.138	0.264
TF-IDF	F	F	-	-	-	Bi & Tri	0.655	**0.395**	0.229	**0.436**	0.710	0.462	0.206	**0.432**	0.874	**0.769**	0.107	0.264	0.706	0.498	0.144	0.284
D2V	C	C	-	200	-	-	0.653	0.334	0.214	0.360	0.685	0.407	0.189	0.348	0.846	0.672	0.102	0.224	0.736	0.470	0.146	0.272
W2V	F	S	-	300	SG	-	0.653	0.334	0.207	0.376	0.697	0.398	0.186	0.364	0.851	0.695	0.102	0.240	0.710	0.477	0.137	0.272
TF-IDF	F	S	-	-	-	Uni & Bi	0.650	0.334	0.208	0.352	0.712	0.411	0.191	0.352	**0.880**	0.683	0.106	0.228	0.702	0.452	0.139	0.256
D2V	F	C	-	200	-	-	0.645	0.367	0.221	0.400	0.686	0.434	0.190	0.372	0.850	0.691	0.103	0.240	0.738	0.489	0.152	0.292
D2V	F	C & R	-	100	-	-	0.644	0.381	0.221	0.420	0.668	0.439	0.188	0.392	0.856	0.732	0.103	0.260	0.739	0.508	0.151	0.292
D2V	C	C	-	100	-	-	0.644	0.324	0.208	0.344	0.675	0.373	0.184	0.328	0.839	0.636	0.100	0.220	0.728	0.460	0.143	0.264
TF-IDF	F	C & R	-	-	-	Bi & Tri	0.644	0.334	0.206	0.344	0.689	0.385	0.181	0.328	0.844	0.675	0.097	0.212	0.711	0.445	0.135	0.240
TF-IDF	F	C	-	-	-	Bi & Tri	0.643	0.333	0.207	0.348	0.687	0.379	0.180	0.328	0.850	0.653	0.096	0.208	0.713	0.458	0.138	0.252
LDA	S	S	10	-	-	-	0.642	0.267	0.169	0.224	0.653	0.284	0.143	0.216	0.762	0.450	0.063	0.108	0.690	0.353	0.100	0.128
W2V	F (NS)	F (NS)	-	300	SG	-	0.641	0.314	0.198	0.324	0.684	0.381	0.170	0.304	0.837	0.678	0.095	0.224	0.721	0.473	0.144	0.276
TF-IDF	C	C	-	-	-	Bi & Tri	0.638	**0.395**	0.227	0.420	0.698	0.462	0.201	0.400	0.856	0.748	0.105	0.260	0.691	0.507	0.142	0.292
TF-IDF	C & R	C & R	-	-	-	Bi & Tri	0.637	0.376	0.225	0.408	0.698	0.430	0.202	0.392	0.850	0.742	0.106	0.264	0.683	0.498	0.141	0.292
D2V	F	C	-	300	-	-	0.635	0.368	0.215	0.392	0.669	0.439	0.186	0.372	0.838	0.686	0.102	0.236	0.731	0.478	0.148	0.276
D2V	F	C	-	100	-	-	0.635	0.377	0.216	0.412	0.670	0.444	0.188	0.384	0.837	0.697	0.103	0.244	0.721	0.483	0.146	0.284
TF-IDF	F	S	-	-	-	Bi & Tri	0.632	0.349	0.208	0.356	0.679	0.422	0.186	0.352	0.865	0.715	0.106	0.244	0.705	0.473	0.139	0.256
TF-IDF	S	S	-	-	-	Uni & Bi	0.631	0.316	0.195	0.328	0.684	0.367	0.177	0.324	0.856	0.670	0.099	0.224	0.689	0.423	0.127	0.220

							Experts Mean				Expert 1				Expert 2				Expert 3			
Model	**Training Corpus**	**Evaluation Corpus**	**Topics**	**Dimensions**	**Algorithm**	**N-Grams**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**
W2V	F (NS)	F (NS)	-	300	CBOW	-	0.629	0.325	0.201	0.348	0.674	0.399	0.174	0.332	0.843	0.694	0.096	0.224	0.720	0.489	0.146	0.288
W2V	F (NS)	F (NS)	-	100	SG	-	0.629	0.316	0.197	0.328	0.673	0.380	0.168	0.304	0.821	0.668	0.092	0.216	0.708	0.464	0.142	0.268
W2V	F (NS)	F (NS)	-	100	CBOW	-	0.622	0.307	0.196	0.312	0.668	0.371	0.168	0.296	0.806	0.630	0.092	0.208	0.710	0.467	0.141	0.260
TF-IDF	S	S	-	-	-	Bi & Tri	0.622	0.316	0.201	0.324	0.670	0.386	0.180	0.320	0.845	0.667	0.100	0.220	0.692	0.433	0.132	0.232
D2V	S	S	-	300	-	-	0.614	0.194	0.170	0.180	0.638	0.223	0.145	0.172	0.754	0.431	0.073	0.116	0.695	0.310	0.110	0.116
D2V	S	S	-	100	-	-	0.610	0.198	0.168	0.180	0.637	0.224	0.144	0.172	0.758	0.445	0.075	0.128	0.691	0.325	0.110	0.128
D2V	S	S	-	200	-	-	0.610	0.196	0.168	0.192	0.637	0.224	0.143	0.180	0.760	0.423	0.074	0.128	0.696	0.327	0.110	0.136
BM25	F	C	-	-	-	-	0.601	0.263	0.165	0.240	0.622	0.277	0.136	0.204	0.704	0.340	0.047	0.052	0.642	0.304	0.094	0.116
BM25	F	F	-	-	-	-	0.599	0.262	0.165	0.236	0.617	0.273	0.133	0.196	0.701	0.340	0.047	0.052	0.640	0.304	0.095	0.116
BM25	C	C	-	-	-	-	0.598	0.266	0.165	0.244	0.619	0.281	0.135	0.208	0.705	0.340	0.047	0.052	0.640	0.303	0.094	0.112
BM25	F	C & R	-	-	-	-	0.596	0.263	0.164	0.240	0.618	0.277	0.134	0.204	0.697	0.340	0.046	0.052	0.636	0.304	0.094	0.116
BM25	C & R	C & R	-	-	-	-	0.595	0.266	0.165	0.244	0.617	0.281	0.134	0.208	0.693	0.340	0.046	0.052	0.637	0.304	0.094	0.116
D2V	F	S	-	200	-	-	0.574	0.164	0.156	0.164	0.582	0.188	0.128	0.148	0.705	0.353	0.060	0.084	0.683	0.296	0.104	0.108
D2V	F	S	-	100	-	-	0.572	0.166	0.155	0.168	0.582	0.195	0.128	0.156	0.706	0.357	0.060	0.088	0.674	0.294	0.102	0.108
D2V	F	S	-	300	-	-	0.572	0.168	0.156	0.172	0.581	0.193	0.128	0.156	0.710	0.357	0.060	0.088	0.678	0.299	0.103	0.112
BM25	S	S	-	-	-	-	0.566	0.185	0.150	0.172	0.602	0.209	0.128	0.152	0.696	0.391	0.056	0.072	0.595	0.276	0.086	0.092
LDA	C	C	10	-	-	-	0.512	0.287	0.239	0.324	0.557	0.332	0.202	0.288	0.673	0.496	0.089	0.148	0.556	0.352	0.145	0.200
LDA	C & R	C & R	10	-	-	-	0.511	0.293	0.185	0.264	0.587	0.362	0.175	0.264	0.686	0.474	0.089	0.152	0.520	0.320	0.112	0.168
LDA	F	C	10	-	-	-	0.506	0.292	0.234	0.317	0.566	0.334	0.200	0.285	0.701	0.540	0.091	0.148	0.531	0.359	0.141	0.188
LDA	F	C & R	10	-	-	-	0.501	0.273	0.191	0.288	0.574	0.331	0.175	0.280	0.797	0.614	0.094	0.180	0.591	0.393	0.121	0.208
LDA	F	C	20	-	-	-	0.435	0.249	0.210	0.271	0.489	0.287	0.182	0.239	0.589	0.437	0.075	0.126	0.508	0.370	0.135	0.190
LDA	F	F	10	-	-	-	0.427	0.221	0.180	0.224	0.488	0.277	0.160	0.208	0.647	0.463	0.080	0.108	0.517	0.330	0.110	0.140
LDA	F	F	20	-	-	-	0.418	0.253	0.220	0.250	0.482	0.307	0.204	0.234	0.706	0.585	0.113	0.147	0.551	0.411	0.165	0.194

							Experts Mean				Expert 1				Expert 2				Expert 3			
Model	**Training Corpus**	**Evaluation Corpus**	**Topics**	**Dimensions**	**Algorithm**	**N-Grams**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**	**mAR**	**Recall@5**	**mAP**	**Precision@5**
LDA	C	C	20	-	-	-	0.409	0.303	0.305	0.333	0.472	0.353	0.281	0.303	0.684	0.625	0.172	0.203	0.493	0.439	0.202	0.235
LDA	C & R	C & R	20	-	-	-	0.407	0.209	0.187	0.211	0.421	0.239	0.165	0.199	0.678	0.487	0.098	0.133	0.525	0.331	0.114	0.144
LDA	C & R	C & R	40	-	-	-	0.376	0.238	0.277	0.289	0.420	0.275	0.249	0.271	0.589	0.465	0.140	0.172	0.441	0.342	0.163	0.198
LDA	F	C & R	40	-	-	-	0.374	0.273	0.262	0.302	0.428	0.316	0.248	0.287	0.622	0.509	0.149	0.184	0.478	0.382	0.171	0.215
LDA	C	C	40	-	-	-	0.361	0.308	0.326	0.356	0.422	0.375	0.312	0.351	0.681	0.629	0.201	0.230	0.478	0.418	0.216	0.235
LDA	F	C	40	-	-	-	0.359	0.223	0.244	0.277	0.400	0.258	0.223	0.261	0.626	0.536	0.142	0.182	0.509	0.368	0.165	0.206
LDA	C & R	C & R	80	-	-	-	0.356	0.302	**0.390**	**0.444**	0.413	0.367	**0.365**	**0.424**	0.681	0.640	**0.254**	**0.294**	0.477	0.429	**0.302**	**0.342**
LDA	F	F	80	-	-	-	0.355	0.294	**0.381**	0.378	0.418	0.359	**0.372**	0.372	0.589	0.564	**0.252**	0.254	0.418	0.392	**0.277**	0.278
LDA	F	C & R	20	-	-	-	0.352	0.237	0.199	0.234	0.414	0.298	0.178	0.210	0.611	0.562	0.117	0.152	0.435	0.368	0.147	0.182
LDA	F	F	40	-	-	-	0.339	0.265	0.368	0.394	0.388	0.313	0.347	0.370	0.586	0.506	0.222	0.241	0.422	0.338	0.245	0.269
LDA	F	C & R	80	-	-	-	0.311	0.221	0.297	0.333	0.369	0.273	0.276	0.312	0.595	0.511	0.222	0.256	0.419	0.355	0.221	0.253
LDA	F	C	80	-	-	-	0.308	0.240	0.272	0.291	0.366	0.280	0.258	0.275	0.513	0.443	0.156	0.165	0.389	0.325	0.199	0.213
LDA	C	C	80	-	-	-	0.280	0.255	**0.437**	**0.465**	0.336	0.308	**0.437**	**0.465**	0.654	0.643	**0.340**	**0.362**	0.415	0.409	**0.325**	**0.345**
LDA	F	S	80	-	-	-	0.076	0.063	0.060	0.060	0.098	0.084	0.060	0.060	0.200	0.200	0.040	0.040	0.160	0.160	0.040	0.040
LDA	F	S	20	-	-	-	0.075	0.063	0.020	0.020	0.096	0.084	0.020	0.020	0.200	0.200	0.000	0.000	0.160	0.160	0.000	0.000
LDA	F	S	10	-	-	-	0.075	0.063	0.020	0.020	0.096	0.084	0.020	0.020	0.200	0.200	0.000	0.000	0.160	0.160	0.000	0.000
LDA	F	S	40	-	-	-	0.074	0.063	0.040	0.040	0.096	0.084	0.040	0.040	0.200	0.200	0.020	0.020	0.160	0.160	0.020	0.020
BM25	F	S	-	-	-	-	0.070	0.063	0.020	0.020	0.091	0.084	0.020	0.020	0.200	0.200	0.000	0.000	0.160	0.160	0.000	0.000

Table 16 - Complete results set for the second study