UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Natural Language Processing on the legal domain: Judgment Prediction under article 8 of the European Court of Human Rights

Yawri D. Carr Quirós

Monograph - MBA in Artificial Intelligence and Big Data

ICMC USP
SÃO CARLOS

Yawri D. Carr Quirós

Natural Language Processing on the legal domain: Judgment Prediction under article 8 of the European Court of Human Rights

Monograph presented to the Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, as part of the requirements for obtaining the title of Specialist in Artificial Intelligence and Big Data.

Concentration area: Artificial Intelligence and Big Data

Orientador: Prof. Dr. Ricardo Rodrigues Ciferri

Versão original

São Carlos

2023

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi, ICMC/USP, com os dados fornecidos pelo(a) autor(a)

S856m

Carr, Yawri
Natural Language Processing on the legal domain: Judgment Prediction under article 8 of the European Court of Human Rights / Yawri D. Carr Quirós ; orientador Ricardo Rodrigues Ciferri. – São Carlos, 2023.
55 p. : il. (algumas color.) ; 30 cm.

Monograph (MBA in Artificial Intelligence and Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2023.

1. LaTeX. 2. abnTeX. 3. Classe USPSC. 4. Editoração de texto. 5. Normalização da documentação. 6. Tese. 7. Dissertação. 8. Documentos (elaboração). 9. Documentos eletrônicos. I. Ciferri, Ricardo Rodrigues, orient. II. Título.

Yawri D. Carr Quirós

Natural Language Processing on the legal domain: Judgment Prediction under article 8 of the European Court of Human Rights

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial e Big Data

Advisor: Prof. Dr. Ricardo Rodrigues Ciferri

Original version

São Carlos

2023

RESUMO

Carr, Y. D. Q. **Processamento de linguagem natural no domínio jurídico: previsão de julgamentos do artigo 8º do Tribunal Europeu dos Direitos Humanos**. 2023. 55p. Monografia (MBA em Inteligência Artificial e Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

Este estudo visa melhorar o processo de tomada de decisão judicial no Tribunal Europeu dos Direitos Humanos (TEDH), através do desenvolvimento de modelos de aprendizagem automática capazes de prever com precisão os resultados dos casos do Artigo 8.º. O estudo procurou descobrir padrões e características inerentes às violações do Artigo 8, aplicando aprendizagem automática e análise de dados, informando assim estratégias e decisões no âmbito do TEDH. Utilizando um conjunto de dados abrangente e métodos como Support Vector Machine, Logistic Regression, Naive Bayes e K-Nearest Neighbours, a pesquisa demonstrou com sucesso a capacidade dos modelos de prever resultados de casos com alta eficácia. Os modelos exibiram forte precisão preditiva, com o modelo SVM atingindo a maior taxa de precisão. As descobertas contribuem para o campo da análise jurídica, fornecendo uma base para a aplicação da inteligência artificial no direito internacional dos direitos humanos, sugerindo um potencial significativo para estes métodos otimizarem a análise de casos e apoiarem a proteção dos direitos humanos.

Palavras-chave: Processamento de Linguagem Natural, Aprendizado de Máquina, Modelagem Preditiva, Previsão de Julgamento Legal, Tribunal Europeu de Direitos Humanos.

ABSTRACT

Carr, Y. D. Q. **Natural Language Processing on the legal domain: Judgment Prediction under article 8 of the European Court of Human Rights**. 2023. 55p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

This study aims to improve the judicial decision-making process within the European Court of Human Rights (ECtHR) by developing machine learning models capable of accurately predicting the outcomes of Article 8 cases. The study sought to uncover patterns and characteristics inherent to Article 8 violations by applying machine learning and data analytics, informing strategies and decisions within the ECHR framework. Employing a comprehensive dataset and utilizing methods such as Support Vector Machine, Logistic Regression, Naive Bayes, and K-Nearest Neighbors, the research successfully demonstrated the models' ability to forecast case outcomes with high efficacy. The models exhibited predictive solid accuracy, with the SVM model achieving the highest accuracy rate. The findings contribute to legal analytics by providing a foundation for the application of artificial intelligence in international human rights law, suggesting a significant potential for these methods to optimize case analysis and support the protection of human rights.

Keywords: Natural Language Processing, Machine Learning, Predictive Modeling, Legal Judgment Prediction, European Court of Human Rights. .

LISTA DE FIGURAS

Figura 1 – Support Vector Machine decision boundary and margins. Source: Medina (2021) 36
Figura 2 – Linear regression vs. logistic regression. Source: Medina (2021) 37

LISTA DE TABELAS

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
ACtHPR	African Court on Human and Peoples' Rights
AI	Artificial Intelligence
ECHR	European Convention of Human Rights
ECtHR	European Court of Human Rights
IACtHR	Inter-American Court of Human Rights
IHRCs	International Human Rights Courts
kNN	k Nearest Neighbors
LJP	Legal Judgment Prediction
ML	Machine Learning
MLE	Maximum Likelihood Estimation
NB	Naïve Bayes
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
SVM	Support Vector Machines
TF-IDF	Term frequency-inverse document frequency

SUMÁRIO

1 INTRODUCTION 21
1.1 Motivation 21
1.2 Research question and Objectives 22

2 TRABALHOS RELACIONADOS 25

3 THEORETICAL FOUNDATION 29
3.1 Artificial intelligence in social and legal research 29
3.2 Legal Judgment Prediction 29
3.3 European Court of Human Rights (ECtHR) 30
3.3.1 Article 8 31
3.4 The matter of privacy 32
3.5 Preprocessing in text data 34
3.5.1 Tokenization 34
3.5.2 Term frequency-inverse document frequency 34
3.5.3 Machine learning techniques 34
3.5.4 Support Vector Machine 35
3.5.4.1 Logistic regression 36
3.5.5 Naive Bayes 38
3.5.6 k-Nearest Neighbors 39

4 METHODOLOGY 41
4.1 Data collection 41
4.2 Preprocessing 41
4.2.1 Preprocessing of the train set 41
4.2.2 Preprocessing of the test set 42
4.2.3 Domain knowledge to supervise categorization 43
4.3 TF-IDF 44
4.3.1 TF-IDF on the training set 44
4.3.2 TF-IDF on the test set 44
4.4 Model selection and description 45
4.4.1 Support Vector Machine 45
4.4.2 Logistic Regression 45
4.4.3 Naive Bayes 46
4.4.4 k-Nearest Neighbors 46

5 RESULTS 47

5.0.1 **Support Vector Machine** 47
5.0.2 **Logistic Regression** 49
5.0.3 **Naive Bayes** 49
5.0.4 **K-Nearest Neighbors** 50

6 CONCLUSIONS 51

Referências 53

1 INTRODUCTION

The rise of international human rights legislation and practice has significantly changed the old paradigm of international adjudication. This judicial process contributes to delivering a court's resolution and is dominated by states' notions. International human rights courts have become essential in comprehending international adjudication (EBOBRAH, 2014).

The European Court of Human Rights (ECtHR) is a regional tribunal that protects and enforces human rights while moving through complex and generally slow legal procedures. The rights to privacy, home, family life, and correspondence are safeguarded through Article 8 of the European Convention on Human Rights (Council of Europe: European Court of Human Rights, 2020).

This project studies Article 8 violations and attempts to predict whether a case before the ECtHR will be determined as a violation of Article 8 by applying machine learning methods. For this purpose, case law provides historical insights, perspectives, and approaches. The case law texts have been transformed into comprehensible numerical data points by preprocessing, including tokenization and cleaning data (LEE, 2023; UCAK; ASHYRMAMATOV; LEE, 2023; VIJAYARANI; JANANI, 2016).

Afterward, various techniques were built to develop machine learning models for predicting the outcome of cases and to evaluate which performs better. For instance, Support Vector Machines (SVM), Logistic Regression, Naive Bayes, and k-Nearest Neighbors were the applied algorithms for this task (ALETRAS *et al.*, 2016; MEDVEDEVA *et al.*, 2019).

In this work, conceptual foundations, development, the programming process, and evaluation metrics from these methods will be explained throughout the present document (ALETRAS *et al.*, 2016; MEDVEDEVA *et al.*, 2019). While Neural networks, deep learning, and transformers are mentioned techniques in related works (CHALKIDIS *et al.*, 2019), these will not be employed for the present work, as with the more traditional machine learning methods, results were already satisfying for the task. This study expects to add significant knowledge to the continuing conversation about human rights jurisprudence by bridging the gap between legal reasoning and machine learning (CHALKIDIS *et al.*, 2019).

1.1 Motivation

The motivation and relevance of this project lie in its multidisciplinary nature, between the legal domain, technology, and societal impact. The ability to predict outcomes on legal judgments becomes a valuable tool for practitioners, policymakers, and the public.

This project attempts to contribute to the legal sector by leveraging machine learning techniques on a dataset with ECtHR cases.

First, the project facilitates the development of machine learning tools that can identify relevant legal cases, enhancing the efficiency of courts. These tools can extract patterns and critical factors contributing to specific judicial decisions.

Due to these applications, the case review processes are accelerated, enabling legal professionals to focus on more nuanced aspects of the law. For instance, although Brazil is the country that has the most lawyers per habitant worldwide (Ordem dos Advogados do Brasil, 2022; PETROV, 2022), a massive amount of cases are still waiting to be solved (Conselho Nacional de Justiça, 2023). At the same time, in other countries, cases are not judged on time, are poorly approached, or are not even solved, which affects the most vulnerable sectors of society (CUI *et al.*, 2022).

Second, predictive models can assess the viability of filing a lawsuit and facilitate legal adjudication by identifying cases more likely to be considered violations. These instruments could save significant resources for legal practitioners and citizens by providing an informed analysis of the likelihood of success. This is particularly valuable in filtering out cases with low chances of success and streamlining the legal process.

Third, the project aids in prioritizing the decision-making process. Legal professionals can allocate resources efficiently by quickly identifying cases that are more likely to result in a violation of Article 8. Thus, they can ensure that urgent matters are addressed promptly, preventing potential harm or injustice due to delays.

Lastly, the project contributes to time efficiency, as it addresses the challenge of massive delays in the judicial process. Automating aspects of case evaluation and prediction simplifies the workflow for legal professionals. Thus, reducing waiting periods for cases and enhancing the judicial system's overall efficiency contribute to a timely and fair resolution of legal matters.

1.2 Research question and Objectives

How can the deployment and comparative analysis of various machine learning algorithms identify the most accurate model to enhance the efficiency of judicial decision-making in Article 8 cases at the European Court of Human Rights (ECtHR)?

1. To deploy and evaluate machine learning algorithms that can analyze data and predict outcomes according to Article 8 of the ECtHR.

2. To conduct comparative analysis for accuracy Determination of the implemented machine learning models.

3. To apply machine learning models that could improve the efficiency of judicial

decision-making in Article 8 cases within the ECtHR.

2 TRABALHOS RELACIONADOS

In the context of the present project, some related works have been identified and serve as a guide on how to approach problems. Some of them, such as (ALETRAS *et al.*, 2016; MEDVEDEVA *et al.*, 2019) coincide in implementing supervised machine learning for legal judgment prediction. This way, during the training phase, the computer learns from textual data from the ECtHR, including the decisions taken by judges. Patterns associated with each verdict class, such as previously, are violation vs. no violation.

The machine learning model identifies these labels. Later, the model is tested on a case without judgment during the testing phase, using the identified information to predict the most likely judgment. The third paper goes beyond traditional machine learning methods and introduces innovative approaches to tasks with diverse neural network models. It also uses transformers such as BERT.

An overview of the paper will be done to know more closely about the different methods each paper has developed and deployed. The publication from (ALETRAS *et al.*, 2016), represents one of the most well-known works on how to use machine learning methods to predict whether a specific Article of the European Convention on Human Rights (ECHR) has been violated.

For this task, texts are used as the data for this paper. Sections of the dataset, such as facts, applicable law, and arguments, are used. While creating their dataset, the authors considered the articles with more cases assigned to them, such as the 3rd, 6th, and 8th of the ECHR. They selected equal violation and non-violation classes to maintain a balanced dataset for each article.

After the traditional text standardization by lower-casing and removing stop words, the authors looked for features with two main approaches: N-gram features and topics. N-gram features employed the Bag of Words model, representing text as the frequency of N-grams, allowing them to create a feature matrix for each case section. Topics were created by clustering semantically similar N-grams.

The N-grams and the topics are employed as textual features for training Support Vector Machine (SVM) classifiers with a linear kernel function. They defined their classification as binary, predicting violation or non-violation of a case for a specific article. Positive and negative weights were assigned to each class, with violation and non-violation labels.

For model evaluation, they implemented a 10-fold cross-validation, reserving 10% of the data at each stage to measure how well the performance was. The linear SVM's regularization parameter (C) was fine-tuned using grid-search, implementing data from

other articles for parameter tuning to optimize the model.

The authors argue that their paper differs from previous ones because studies usually use non-textual data. At the same time, they have pioneered the prediction of decision outcomes by using textual data at an international human rights court. The models have achieved an average of 79% of accuracy. The facts section was the most predictive.

After this first approach, another paper applied what the previous one achieved, going beyond. In this new paper, (MEDVEDEVA *et al.*, 2019) have explored a method for automatically categorizing legal texts using natural language processing techniques.

The study uses the data from the ECtHR, and the paper discusses the importance of creating a balanced dataset for training the algorithm. A balanced dataset is created to prevent the algorithm from learning the distribution of violation and non-violation cases rather than specific characteristics. Balancing involves randomly removing violation cases to have equal non-violation cases.

Machine learning was employed here to develop a system for predicting the category, which would be a verdict of violation vs. no violation, associated with a case. The paper introduces an example of training a program with data different from the text to recognize pictures of cats and dogs. The process using text data is similar, as the program learns to recognize patterns and characteristics associated with different classes. The authors developed a machine learning method for this study, analyzing judgments from the ECtHR cases. The objective was to predict whether a particular article of the ECHR was violated.

The choice of machine learning approach ends up with a Support Vector Machine (SVM) Linear Classifier. (MEDVEDEVA *et al.*, 2019) explain that the SVM algorithm selects a hyperplane to separate data points, maximizing the margin for classifying new data correctly. It also separates data based on labels, attempting to get a simple equation that separates different data points with low error.

The authors evaluated the performance of the machine learning approach using a test set that was separated from the training set. The program's decisions are compared to the actual court decisions, measuring the system's accuracy in correctly identifying decisions. Another evaluation method discussed is k-fold cross-validation, where the available data is split into k parts for iterative training and testing to determine optimal parameters. It also assesses performance with different data samples to improve the models' generalization to unseen cases.

(MEDVEDEVA *et al.*, 2019) state that the scope of (ALETRAS *et al.*, 2016) has been extended in this paper by including more articles and increasing the number of cases per article. A pivotal departure was excluding the Law section of cases, reducing model bias by eliminating access to court discussions. Their achieved scores for the three articles

analyzed are slightly lower than (ALETRAS *et al.*, 2016), and their approach is more representative, utilizing a balanced dataset of 1942 cases compared to their 584.

The last paper to be considered in the present section is from (CHALKIDIS *et al.*, 2019), which introduces a neural networks-based approach to address legal judgment prediction. First, it also works on binary violation classification based on the case facts. Second, multi-label violation prediction identifies specific human rights articles violated, considering the total of 66 articles within the European Convention of Human Rights.

Additionally, it predicts the importance of a case on a scale from 1 to 4 so that the scores show the case's contribution to the development of case law. The paper introduces several neural models designed to handle different aspects of the prediction challenges. The BiGRU-Att model utilizes a Bidirectional Gated Recurrent Unit (BIGRU) with self-attention to process case facts and make predictions.

The Hierarchical Attention Network (HAN) employs a two-level BIGRU with self-attention, providing a hierarchical structure for improved text classification. The Label-Wise Attention Network (LWAN) specializes in multi-label classification, employing multiple attention mechanisms for distinct labels. BERT, a language model based on Transformers, and its hierarchical version, HIER-BERT, are also introduced, with the latter addressing BERT's limitation in processing long documents.

The results obtained bring interesting perspectives. HAN outperforms previous methods in binary violation classification, while HIER-BERT demonstrates superior performance, overcoming BERT's truncation limitations. The analysis further explores the models' sensitivity to demographic information, revealing potential biases, especially in the case of HIER-BERT.

In multi-label violation prediction, HIER-BERT does better as it can handle complex tasks. HIER-BERT shows the highest correlation with gold scores for case importance prediction, which tells about its efficacy in capturing case importance. The paper's discussion also touches upon the impact of background knowledge on predicting case importance.

3 THEORETICAL FOUNDATION

3.1 Artificial intelligence in social and legal research

Nowadays, research topics within artificial intelligence are of pressing significance. The enormous number of opportunities and advances in industry and science these technologies bring make them an essential topic of discussion. Nevertheless, many privacy and security issues arise along with its black box.

Artificial intelligence tools are rapidly evolving and valuable in different fields, including natural language processing and recommendations (RUSSELL; NORVIG, 2022). One part of AI is Machine learning (ML) applications, which, more than learning patterns and correlations among data, also analyze and take action based on data, making predictions of outcomes (LEHR; OHM, 2017).

A task is assigned, and a massive amount of data serves as samples of how to carry out the task or to identify patterns in it. The program then learns the most effective way to produce the intended result or output (ALPAYDIN, 2016).

Working with big data and predictions to answer social science questions is a new trend that is important for new social scientists, including legal scholars (KESARI *et al.*, 2022). Data analysis becomes machine learning when automated, and a computer program is trained through data.

Data is evaluated, extracts usable information, anticipates undetermined features, and offers action recommendations (LINDHOLM *et al.*, 2021). For this reason, (LINDHOLM *et al.*, 2021) claim that machine learning is a form of example-based programming.

3.2 Legal Judgment Prediction

Legal judgment prediction (LJP) holds a protagonist role within artificial intelligence (AI) in the legal domain (ZHANG *et al.*, 2023). Several studies, such as (CUI *et al.*, 2022), (LAGE-FREITAS *et al.*, 2022), and (SANTOSH *et al.*, 2022), have delved into the application of natural language processing (NLP) techniques in LJP.

NLP is an interdisciplinary field combining computer science and linguistics that intends to train computer programs to understand and generate texts or speech (KHURANA *et al.*, 2022). Hence, it allows for extracting valuable information and knowledge from texts. It is essential for understanding the reasoning of legal arguments and developing models that can predict case outcomes. Thus, Legal Judgment Predictions use extensive datasets of legal texts to develop models that can accurately predict the outcomes of legal cases.

LJP has been present for many years until now, but it has gotten many advances. It started to be based on statistical methods, but these were performing poorly in various legal areas due to noise in data and other limitations (KHURANA *et al.*, 2022). Due to this, machine learning became the field that was trying to improve the results of the first ones.

Moreover, as AI advances and more resources are available, new methods exist to tackle the related issues. In this sense, the most recent advancements are led by neural networks, deep learning, and pre-trained transformers (KHURANA *et al.*, 2022).

One of the critical aspects of legal judgment prediction is the utilization of enormous datasets encompassing a wide range of legal cases. These datasets often include case details, legal arguments, historical judgments, and relevant legal precedents. By employing advanced algorithms, machine learning models can analyze these datasets to identify patterns and correlations that may influence judicial decisions.

Machine learning algorithms can notice patterns and correlations within legal datasets (MEDVEDEVA *et al.*, 2019). The data typically includes case facts, relevant arguments, and jurisprudence (CUI *et al.*, 2022). Working in the legal domain is challenging because such documents are characterized as being very extensive, using judicial vocabulary not understandable for people without studies in law, and relying on background knowledge.

More recently, some authors have been using deep learning, such as recurrent neural networks (RNNs) and transformers, to tackle the issue of an artificially intelligent system that can also capture sequences and contexts within legal texts. The intention of improving the accuracy of predictions is the reason for such applications (CHALKIDIS *et al.*, 2019).

While LJP has revolutionized the legal field and has performed well in the mentioned papers, it does not intend to replace human legal scholars, lawyers, or judges. Rather than that, it is supposed to support legal enforcement, the legal work, and the citizens waiting for solutions from the legal system (MEDVEDEVA *et al.*, 2019).

As with any AI system, it has its limitations, which are enhanced by the nature of the legal system, which is inherently complex and constantly faces unforeseen challenges. Legal frameworks can be reformed, and political processes are also elements that play a role in legal decisions and public opinion influences, which can impact the outcomes of cases. Hence, necessary ethical measures have to be taken into consideration by data scientists so that they are aware of the different influences around legal judgment.

3.3 European Court of Human Rights (ECtHR)

International human rights courts (IHRCs) are specialized courts whose practice is tied to a binding human rights document. Usually, their purpose is to monitor the execution of their relevant human rights conventions.

As a result, the objective of those courts is to rule on infringements of human rights based on their human rights conventions. The African Court on Human and Peoples' Rights (ACtHPR), the Inter-American Court of Human Rights (IACtHR), and the European Court of Human Rights (ECtHR) are the three courts of this kind that are in operation nowadays (EBOBRAH, 2014).

The present project is based on data from case law from the ECtHR. It was established in Strasbourg, France, by the European Convention on Human Rights (ECHR) in the period after wars, specifically in 1959 (European Commission, Accessed: 2023). After this year, Europe intended to recover peace and the rule of law, establishing a court that could ensure that human rights were promoted, protected, and safeguarded.

This international human rights tribunal hears allegations of human rights violations that their signatory states could have committed. The court is based on and follows the ECHR, which contains all the protected human rights. The applicants can be one person, a group, and other states (European Commission, Accessed: 2023).

Rulings from the ECtHR set precedents in various highly relevant topics for societies worldwide. Now, the institution states that more than ten thousand cases have been judged by this well-known court (European Commission, Accessed: 2023).

A dataset released by (CHALKIDIS *et al.*, 2021) containing 11 thousand cases from the ECtHR until 2021 has been used for the present project. The structure of the cases are:

- Procedure: Describes the whole legal process that each case has gone through.
- The Facts: It is a very detailed description of the factual foundation of the case, including all relevant events that have provoked the applicant to submit the case to the ECtHR process. This section also introduces laws and principles that are also related to the events in the case.
- The Law: More than the applicable law, the principles and doctrine regarding each case are analyzed. An interpretation of the previously mentioned elements is held so that these can support the facts.
- Operative Provisions: The judgment is presented with legal remedies or penalties that are imposed, taking into consideration the first couple of previous sections. Conclusions of the whole case are established.

3.3.1 **Article 8**

The present research will apply machine learning techniques based on Article 8 of the ECHR. Due to the interest in enhancing the human right to private and family life, this particular interest will be explained in the following sub-section.

Article 8 constitutes a cornerstone for protecting the right to a private sphere. As enshrined in Article 8, "Everyone has the right to respect for private and family life, his home and his correspondence."(Council of Europe: European Court of Human Rights, 2020) This broad but inclusive formulation encapsulates various elements, from interpersonal relationships to personal privacy issues.

A transcendental case illustrating the expansive reach of Article 8 is (NIEMIETZ. . . ,), where the European Court of Human Rights (ECtHR) emphasized the broad scope of the right to private life, encompassing facets such as the right to cultivate relationships with others. Further, Article 8 and the intersection of technology have brought new challenges in the contemporary cyberspace and digital landscape.

Thus, article 8, which protects the right to private life, is under threat in the current context of the instant sharing of large amounts of data, mainly when technology is not managed responsibly. For instance, the case of (ROMANIA, 2017), shows the need to consider the right to privacy on behalf of telecommunications and establishes a precedent regarding the meaning and importance of private life in the digital age. In the case of Case of Mockutė v. Lithuania, the intersection of Article 8 with data protection was faced by the ECtHR, presenting a violation of this article (CASE. . . , ; Council of Europe/European Court of Human Rights, 2022).

Using computational methods, legal scholars can navigate the intricacies of ECtHR decisions, identifying underlying patterns that shape the court's decisions regarding violations or non-violations of Article 8. The predictive potential of machine learning models makes correlations within human rights jurisprudence.

3.4 The matter of privacy

The main interest in focusing this study on Article 8 is because it is one related to private life. The reasons why this right is relevant and how it has become more vulnerable with technological advancements will be explained in the following paragraphs.

Véliz (VéLIZ, 2021b) argues that privacy is power for various reasons. Consequently, power uses knowledge as an instrument to conquer more. Thus, if we do not protect our privacy, we empower someone else, as this other party knows our intimate details. Consequently, when there is an imbalance of knowledge between two parties, this becomes a very destructive power and is dangerous for the party that does not know about the other, which is the relevance of privacy preservation.

Privacy is not just individual but collective (VéLIZ, 2021b). It is also a collective accomplishment. Cambridge Analytica put this on evidence because it affects collectivity when one person puts their privacy at risk. It should matter as individuals and as a community. Due to the nature of its social phenomenon, sociotechnical privacy is the

conceptual approximation we should use (NISSENBAUM, 2010).

Privacy contains social and cultural components, so it appears difficult to detach its linkages to intimacy, sovereignty, and autonomy. Furthermore, depending on the social context, people likely have different conceptions of privacy (VéLIZ, 2021b).

Due to this reason, it is more proper to investigate what privacy means to distinct players across various social worlds and how those stakeholders interact with numerous established value systems, rules, and interests. Rather than presenting privacy as a solely individual issue. As well as how they attempt to materialize them. Practices like these are not just from the users, providers, and regulators. Such constellations may contain opposing ideas influenced by diverse norms and values (VéLIZ, 2021b).

Although the privacy and digital ethics scenario during the COVID-19 pandemic revealed some problematic patterns, there are some changes regarding the perception of privacy. The public worldwide is becoming more skeptical of the technology industry and is following its practices closely. Since then, they have demanded better and more trustful processes. Likewise, there is an increasing consciousness that current software's insufficient privacy represents a security risk on a national level, which could lead governments to take measures (VéLIZ, 2021a).

(JENA *et al.*, 2020) describe some of the current challenges for protecting privacy in the current database scenario. Within this context, there are some aspects to consider: personally identifiable information immediately connected to the data subject, such as a telephone number. Also, quasi-identifiers, with other complement data, would then be linked to the data subject, such as a PIN code or sex; additionally, sensitive columns are those not related to the previous two but still contain relevant information for the individual, like geolocation.

If privacy is violated in these databases, some of the most common privacy risks that could affect them are surveillance, as many companies are tracking their customers in different situations by taking their opinion data or sentiment analysis; disclosure, when the sensitive data of a person is released due to a bad practice on its treatment. Discrimination, due to the disclosure of opinions or information that should be kept anonymous, and abuse, once a company knows specific personal data, they could take unproportionate advantage of others (RAO; KRISHNA; KUMAR, 2018).

These risks usually take place because, despite the advantages of contemporary AI, it is susceptible to several attacks, wherein threat actors attempt to infringe the confidentiality and integrity of machine learning models by including what is purposefully designed to predict erroneously. Thus, this shows the consequence of AI systems that are frequently developed without considering security (GüRSES; ALAMO, 2016). Data is still too easy to steal (VéLIZ, 2021a).

Therefore, as Article 8 of the ECHR encompasses this human right currently in high vulnerability, the present project is focused on contributing to it. Hence, it answers more efficiently to cases that evaluate whether there was a violation of this article.

3.5 Preprocessing in text data

Preprocessing, which involves tokenization, stemming, sentence boundary identification, and stop-word removal, particularly in natural language, considerably decreases the overall length of the input text files. Tokenization is the most crucial and significant step in preprocessing as it facilitates the division of words from the textual data (VIJAYARANI; JANANI, 2016).

3.5.1 Tokenization

As aforementioned, it is necessary to perform tokenization when working with NLP, which is about separating phrases or utterances from a text or speech into units known as tokens (LEE, 2023). Words, fragments of words, characters, numbers, punctuation, and symbols may all be transformed into tokens. Given that each language has unique structures for producing grammar and syntax, this process is a fundamental component of NLP work. Tokenization can produce vocabulary using a document or corpus (LEE, 2023).

3.5.2 Term frequency-inverse document frequency

After getting the tokens from the abovementioned process, a numerical statistical technique called term frequency-inverse document frequency (TF-IDF) is applied. TF-IDF assigns a weight to each word in each document. This procedure is frequently applied in NLP through the weight technique, a metric that assesses the significance of words in the gathering of written records. The content's significance increases directly to how often it appears in documents (TRSTENJAK *et al.*, 2014).

Mathematically, TF-IDF calculates the relevance of term t in document d with respect to the entire document collection D as follows:

$$tfidf(t, d, D) = tf(t, d) \cdot idf(t, D)$$

3.5.3 Machine learning techniques

This research lies in the intersection of legal scholarship and machine learning methodologies. The initial task is a binary classification of cases, whether they violate Article 8 or not. Leveraging the large amount of historical data from the ECHR, patterns that guide the court's decisions concerning Article 8 violations were found. Machine learning is used for this purpose because it is a technique that can discover relationships that would only be done slowly by traditional legal methods.

To develop a predictive model, a variety of machine learning algorithms shall be the object of the experiment. The one with the best performance should be the one that stays as the selected method. Here is a conceptual introduction to the methods that were implemented:

3.5.4 **Support Vector Machine**

This type of supervised learning was originally designed for binary classification, but it has also been adapted to handle more than two classes (GUENTHER; SCHONLAU, 2016). Support Vector Machine (SVM) is a linear classifier characterized by the inclusion of a hyperplane that serves as the decision boundary. This hyperplane facilitates label prediction based on the input features (GUENTHER; SCHONLAU, 2016).

SVM intends to find an optimal decision boundary between the classes. The decision boundary that achieves the largest margin between classes is the preferred one.

The margin refers to the space that exists between the nearest instances of the two classes with respect to the decision boundary (NOBLE, 2006). The hyperplane's equation is based on the linear regression formula as follows:

$$w^T x - b = 0$$

Support vectors are data points located close to class boundaries, while the hyperplane is positioned further away from them. The optimal hyperplane in SVM is composed of the weight vector (w, the input feature vector (x), and the bias (b).

Training SVM involves finding the values of w and b such that the hyperplane effectively divides the data into two classes while maximizing the margin (GUENTHER; SCHONLAU, 2016).

Despite SVM's original linear nature, it can handle non-linear classification tasks through the use of kernel functions. The choice of kernel can significantly impact the model's performance, and experimenting with different kernels is essential to determine the best fit for the data (GUENTHER; SCHONLAU, 2016).

In classification tasks, SVM demonstrates superior generalization on unseen data compared to similar models. However, in natural language processing (NLP) tasks, feature vectors representing text data are often high-dimensional but sparse. This results in the separation of positive and negative examples into distinct regions of the feature space.

This separation is primarily advantageous for SVM's classification hyperplane search within the feature space and contributes to its excellent performance in various NLP tasks. To create high-dimensional representations of text data, a wide range of linguistic features is used, and kernel functions are frequently employed to transform feature vectors into higher-dimensional spaces (LI *et al.*, 2004).

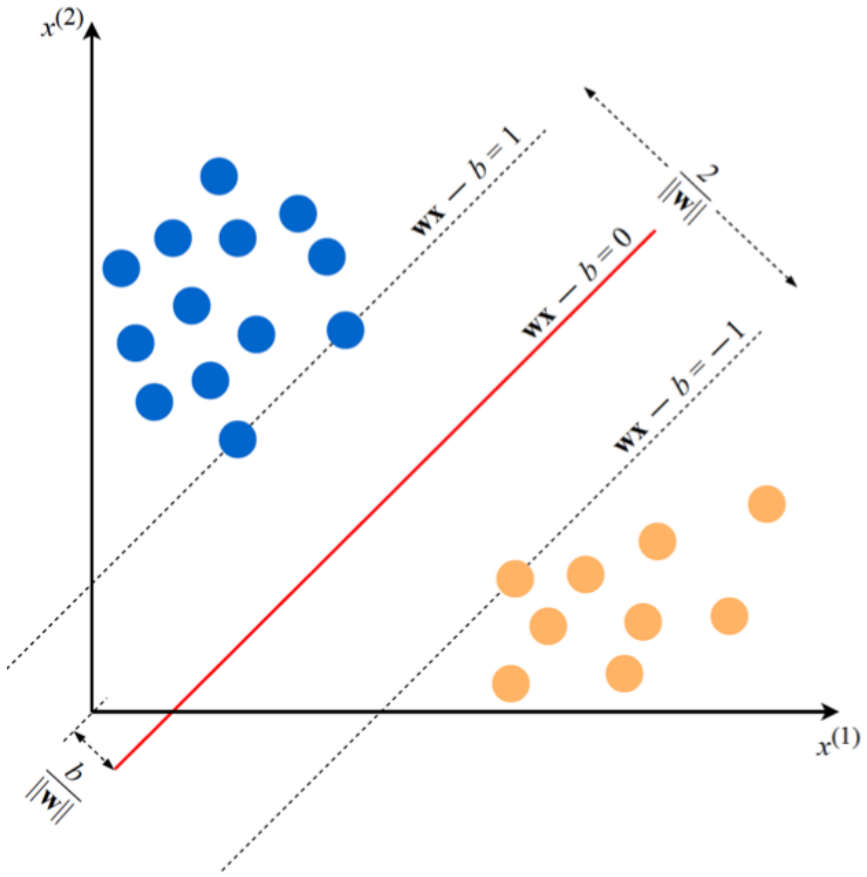

Figura 1 – Support Vector Machine decision boundary and margins. Source: Medina (2021)

3.5.4.1 **Logistic regression**

Logistic regression is one of the probabilistic methods of supervised machine learning. It is a model that classifies a binary outcome and is considered a discriminative classifier. Thus, although Logistic Regression incorporates regression in its name, it is not for that but for classification. Regression is included on its name due to the similarity of its mathematical formula with linear regression (MEDINA, 2021).

A discriminative model focuses on learning how to differentiate classes and does not need many features. One feature that can distinguish between the classes is enough for logistic regression (JURAFSKY; MARTIN, 2008). Its discriminative nature means that by computing direct likelihood, the model will classify. Hence, a higher weight would be given to the most relevant features for distinguishing between two classes. However, logistic regression cannot produce an instance or illustration of any classes (JURAFSKY; MARTIN, 2008).

Training and testing are the two steps a logistic regression must pass through. The first one is oriented to train the weights and the bias through stochastic gradient descent and the loss function of cross-entropy. The second will test through computing a

probability likelihood $P(y|x)$ given a test set that will give back the highest probability label between $y = 1$ if the data point observed is assigned with the class and $y = 0$ if it is not (JURAFSKY; MARTIN, 2008).

In a binary classification task, the sigmoid function is employed to estimate the probability (MEDINA, 2021). It involves using a linear combination of input features, expressed as follows:

$$P(y = 1|x) = \sigma_{w,b}(x) = \frac{1}{1 + e^{-(w^T x+b)}}$$

The sigmoid function is what actually differentiates linear regression from logistic regression (MEDINA, 2021). The following figure shows why sigmoid is relevant:

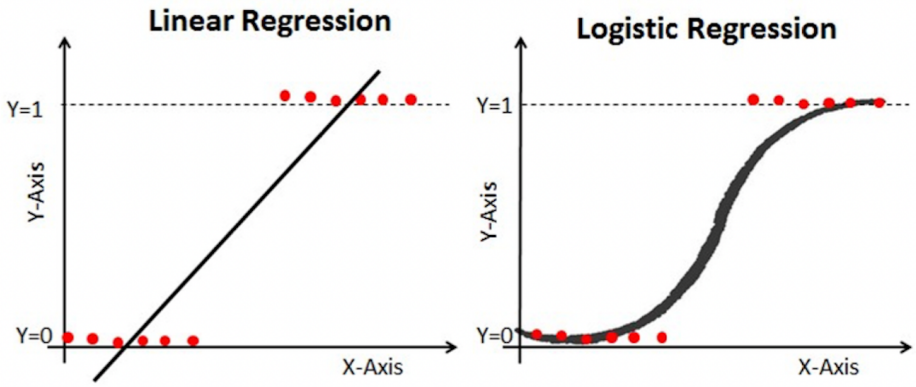

Figura 2 – Linear regression vs. logistic regression. Source: Medina (2021)

The decision boundary in logistic regression guides the decision of the class categorization, which is usually 0.5. If the probability is higher than 0.5, it is assigned to 1 or the positive class, but if the probability cannot exceed this set threshold, the input feature is assigned to the other class, that is usually 0 or a negative class (JURAFSKY; MARTIN, 2008; MEDINA, 2021).

Logistic regression employs the maximum likelihood estimation (MLE) method during training. The objective is to maximize the likelihood function, quantifying the probability of observing the given outcomes under the assumed model (WASSERMAN, 2013). The likelihood function for logistic regression is expressed as:

$$L(\theta|x) = \prod_{i=1}^{n} P(y_i|x_i;\theta)^{y_i}(1 - P(y_i|x_i;\theta))^{1-y_i}$$

Nevertheless, in the majority of machine learning algorithms is employed the log-likelihood function, as it is more convenient (MEDINA, 2021).

$$\log L_{w,b} = \sum_{i=1}^{N} y_i \ln(\sigma_{w,b}(x_i)) + (1 - y_i) \ln(1 - \sigma_{w,b}(x_i))$$

Regularization L1 and L2 are some techniques logistic regression usually implements to reduce overfitting. These kinds of processes are necessary so that the model can also work with unseen data (HASTIE *et al.*, 2021).

On behalf of natural language processing, logistic regression is often applied. (HASTIE *et al.*, 2021) argue that due to the robustness of identifying correlation between features. In the case of working with big data or oversized documents, if many features are considered correlated, logistic regression can perform better than other classifiers (HASTIE *et al.*, 2021).

3.5.5 **Naive Bayes**

Naive Bayes is based on Bayes' rule, and one of the most relevant characteristics of this algorithm is that it considers that features given the class are conditionally independent (WEBB, 2016). Estimations of the posterior probability of each class given an object are conducted. These estimations are what the algorithm would use to classify (WEBB, 2016).

In contrast to logistic regression, Naive Bayes is a generative classifier, which means that implementing naive Bayes involves the model to generate data for each class. This way, it goes beyond simply understanding the decision boundary to separate classes, as logistic regression does (JURAFSKY; MARTIN, 2008).

When testing a naive Bayes, the program would ask which classes better fit each input data point and choose it as a label. Contrary to logistic regression, the likelihood is not the only element that can be considered to assign a class. The likelihood and the prior probability are essential elements of this classifier, as both have to be computed to assign one of the classes to a document (JURAFSKY; MARTIN, 2008).

Naive Bayes has proved more accurate and efficient when trained with short documents and small datasets (HASTIE *et al.*, 2021). One of the general advantages of using Naive Bayes is its insensitivity to noise in both test and train data. In the first one, due to the usage of all predictions to classify, while in the second one, due to the usage of probabilities. Following these characteristics, Naive Bayes also presents robustness for dealing well when values are missing. It takes information from other values if some are missing (WEBB, 2016).

In the context of categorical variables, the frequency of each one will determine the probabilities. When dealing with numerical attributes, probability density estimation is conducted, or the data is discretized (WEBB, 2016).

Naive Bayes can be deployed in two ways when developing text mining projects: the multi-variate Bernoulli and multinomial models. The first one will use vectors of binary variables that could have or not have a word, and this would be to represent a document. The second one works by separating each word independently so that it counts how many

times a word is in a paper (WEBB, 2016).

3.5.6 **k-Nearest Neighbors**

Among the most straightforward machine learning algorithms is k-nearest Neighbor (KNN). Categorizing items into one of the established classes of a sample is the algorithm's goal. Data from training can be utilized during the testing phase of the algorithm, but it is not necessary to use it for classifications. KNN is centered on determining the mutual Euclidean distance between the most alike items or observations from the sample sets (TRSTENJAK *et al.*, 2014).

The k closest neighbors of the data item t are obtained to classify it; this creates the neighborhood of t. The categorization for t is often determined by a majority vote amongst the data records in the neighborhood, regardless of taking distance-based weighting into account. Nevertheless, to use kNN, a suitable value for k must be selected, and the classification's outcome greatly depends on this number (TRSTENJAK *et al.*, 2014).

The kNN approach is somewhat k-biased. While there are other methods for determining the k value, one straightforward method is to repeatedly run the algorithm with various k values and select the one that performs the best (TRSTENJAK *et al.*, 2014).

4 METHODOLOGY

4.1 Data collection

This research relies on a dataset sourced from the work of Chalkidis et al. (CHALKIDIS *et al.*, 2021) concerning the European Court of Human Rights (ECtHR). This dataset, an extension of their 2019 ECtHR dataset, encompasses 11,000 cases adjudicated by the ECtHR. It offers an in-depth exploration of alleged breaches of the European Convention of Human Rights (ECHR) by European states.

The dataset comprises several vital components. First, the 'Facts' section provides a chronological list of paragraphs detailing the main events relevant to each case. Second, 'Allegedly Violated Articles' serve as ground truth, encompassing 40 violable ECHR articles forming the foundation for alleged violations.

It includes Violated Articles, indicating decisions by the court on substantiated violations. Silver Allegation Rationales capture references to case facts and law extracted from ECtHR decisions. Finally, Gold Allegation Rationales represents an annotated subset by legal professionals and experts on the ECtHR, identifying the important paragraphs from the case section supporting alleged violations of the article (CHALKIDIS *et al.*, 2021).

4.2 Preprocessing

4.2.1 Preprocessing of the train set

A list of case facts is extracted from the training data. The 'Facts' column of the Data Frame is then cleaned, with missing values filled as empty strings and the text converted to lowercase. A cleaning function is applied to remove numbers, special characters, and specified punctuation, resulting in a refined in a cleaned version of the facts, containing preprocessed and standardized text data.

The 'Facts' column is then processed to eliminate words commonly found in English stop words. After observing that some irrelevant words frequently appeared in the dataset but were not part of the standard English stop words list, a supplementary list of irrelevant words was created.

The irrelevant words list includes terms like 'application,' 'respectively,' and month names, such as 'January' or 'February'. By incorporating these extra words, the cleaning procedure was adapted to the specific characteristics of the dataset, addressing what the standard English stop words might have missed. The NLTK library is employed for tokenization, and list comprehension filters out words that fall under either category.

Additionally, words with less than four characters are excluded so that short connection words, usually without importance, are removed. The resulting cleaned text from the facts is then joined on a data frame that matches the violated article and the 0 or 1 label.

Lastly, the training dataset was transformed in this data preprocessing step. The primary objective was to convert unstructured train data, into an organized format suitable for subsequent machine learning analysis. Importantly, this process was exclusively for the training data to extract pertinent information.

At its core, a structured dataframe was established using Pandas. This data frame comprised three columns: one containing the facts, another containing 0 or 1, and the third with the violated article label. All of these are important for the classification.

The "Facts"column was designated to store the textual content associated with each case in the training data. Before integrating into the data frame, a preprocessing step was applied to the text. This step involved the removal of common, non-informative words known as 'stopwords', as well as data cleaning. This process ensured the text data was appropriately cleaned and free from irrelevant terms, enhancing the quality of the textual features in subsequent analyses.

The column containing 0 and 1 were critical in identifying whether each case in the training dataset could violate Article 8. It was initialized as a binary indicator variable, where a value of 1 meant a case relevant to article 8 concerns, while a value of 0 designated cases unrelated to article 8.

An inner loop was employed to inspect each case's "violated articles"field systematically. If any articles in this field matched the value '8,' the variable 'privacy' was set to 1, signifying an article 8-related case. This logical assessment ensured that cases aligned with the research's focus on private life violations were classified. Finally, the "VA"column captured the list of violated articles linked to each case.

The training dataset was iterated throughout this data preprocessing phase, with the outlined processing steps applied to each case. Thus, a new data row within the DataFrame for each case was created to append these rows to the existing DataFrame while preserving unique row identifiers.

4.2.2 **Preprocessing of the test set**

In the context of the research study, the preprocessing of the test dataset is separated from the preprocessing applied to the training data. This division of preprocessing tasks for the training and test datasets holds significance for several reasons.

Firstly, the test dataset represents unseen data that serves as an evaluation for assessing the performance of machine learning models. Maintaining a clear demarcation

between the training and test data is essential to simulate real-world scenarios accurately. The independent preprocessing of the test dataset ensures that any insights or predictions derived from the model are grounded in its ability to generalize to new, unseen data, thus applicable to other cases in the future.

The preprocessing pipeline for the test dataset commences with retrieving data, a JSON Lines file located at a specific path. Upon loading the test data, the "Facts"component, which constitutes the textual information associated with each test case, is isolated from the rest. This information is then organized into a structured format by creating a Pandas DataFrame containing the test data. It has a column, 'Facts,' which captures the preprocessed textual content.

The preprocessing of the 'Facts' column in the test data is similar to the procedures previously applied to the training data. It encompasses transforming all text to lowercase, replacing missing values with empty strings, and applying a predefined text-cleaning function. This function removes unnecessary elements, such as particular characters or punctuation, from the text, further enhancing the uniformity and cleanliness of the textual data.

Subsequently, tokenization of the 'Facts' column is carried out, breaking down the text into individual words or tokens. Removing stopwords and irrelevant words is intended to eliminate joint and non-informative words that might obscure meaningful patterns during analysis. Additionally, words with fewer than three characters are excluded from the text, as these are typically considered too brief to convey substantial information.

The resultant cleaned and preprocessed text articles from the test set are assembled into a list. The cleaned test data is structured into a data frame in the final step of test set preparation. Subsequently, the feature set X from the test set is derived from the 'Facts' column.

For the target variable, y values are assigned based on whether the original test cases contain '8' among their violated articles. If '8' is present, y is set to 1, indicating a privacy-related case; otherwise, it is set to 0 for non-article 8-related cases.

These processed X and y from the test set are now ready for use in evaluating machine learning models, enabling rigorous assessment of model performance against the target variable.

4.2.3 **Domain knowledge to supervise categorization**

After the classification done in the previous steps, there was the need to verify if it has been done correctly. Hence, the author of this paper has used her domain knowledge of human rights and legal studies to analyze and confirm if the classification was done correctly.

The binary classification was reviewed by examining five cases violating Article 8. After looking at the case law, it was confirmed that the classification was done correctly, which coincides with the most common words. Domain knowledge can provide more reliability in machine learning tasks.

4.3 TF-IDF

4.3.1 TF-IDF on the training set

Subsequently, the text data transforms into a numerical format suitable for machine learning using the TF-IDF vectorization technique. This method quantifies the significance of words or phrases within the textual data. Thus, the data becomes suitable for analysis by machine learning algorithms.

TF-IDF is a numerical statistic that reflects the importance of a word in a document relative to its frequency across multiple documents. It helps capture the significance of words in the text and is widely used in natural language processing tasks (PEDREGOSA *et al.*, 2011).

There is a critical distinction in applying TF-IDF vectorization to the test set. The `fit_transform` method is employed for the training data, enabling the vectorizer to establish the vocabulary and calculate the Inverse Document Frequency (IDF) values solely based on the training data. Thus, this ensures that the model learns exclusively from the training dataset without any influence from the test data.

4.3.2 TF-IDF on the test set

In contrast, the test set requires a different treatment. The transform method is utilized to avoid any potential data leakage and maintain an adequate evaluation process. This method applies the same vocabulary and IDF values learned from the training data to the test data, ensuring that the test data is transformed consistently without introducing any new vocabulary or IDF calculations.

This approach preserves the independence of the test set and ensures that the model evaluation is performed on data that has not influenced the training process. It adheres to the fundamental principles of machine learning model evaluation, wherein the model's ability to generalize to unseen data is rigorously assessed. Following the TF-IDF vectorization step, the code can progress to subsequent stages of model training and evaluation with the assurance of unbiased and reliable results.

Finally, the inclusion of n-grams in the TF-IDF vectorization process is, especially using an n-gram range of (1, 2), a good approach because it allows the model to capture not only individual words (unigrams) but also sequences of two adjacent words (bigrams). Hence, this helps the model consider the context and relationships between words in the

text data, enhancing its ability to understand the meaning and semantics of the text. In text classification tasks, n-grams can provide valuable information about language patterns and improve the model's performance in recognizing relevant features for classification.

4.4 Model selection and description

4.4.1 Support Vector Machine

The Support Vector Machine (SVM) model is implemented and trained for the task using the Python programming language. The steps include converting the target labels to NumPy arrays with specified data types. Subsequently, a linear kernel SVM model is constructed and trained using the TF-IDF vectorized text data. The linear kernel is chosen based on the nature of the data.

In the initial step, the labels or target variables for the training and test sets are transformed into NumPy arrays. This conversion process facilitates compatibility with the SVM model and associated evaluation metrics. It is also specified that the labels should be stored as integers in the NumPy arrays.

Following label conversion, an SVM model is established using the Support Vector Classification (SVC) algorithm. The kernel function employed here is the linear kernel, which is suited for text classification tasks.

The class weights parameter is introduced here and set as balanced. This mechanism automatically adjusts the class weights based on the distribution of target classes. Hence, this contributes when dealing with imbalanced datasets, as it mitigates potential biases towards majority classes during model training.

Using the 'fit' method, the SVM model is then trained on the transformed TF-IDF representations of the training data and the corresponding NumPy array of training labels. After model training, the trained SVM model is deployed to generate predictions for the test data. This predictive process is executed using the 'predict' method, which is applied to the TF-IDF transformed test data.

4.4.2 Logistic Regression

In this segment of the code, a Logistic Regression model is implemented. The code begins by importing the LogisticRegression class from the scikit-learn library.

The class weight parameter is set to 'balanced,' a technique to address the class imbalance in the dataset. This way, the model will assign different weights to the classes to ensure that it does not excessively favor the majority class during training.

The model is then trained using the 'fit' method. It takes two main inputs: First, the representation of the training data is transformed into a numerical format using the

TF-IDF vectorization technique. It contains the features (textual data) that the model will learn. Second, the corresponding target labels for the training data. It indicates the correct class labels for the training examples.

Once the model is trained, it is ready to make predictions on new, unseen data. In this case, predictions are made on the test data using the 'predict' method. The TF-IDF vectorized representation of the test data is input to the model.

Following the predictions, the code proceeds to evaluate the performance of the Logistic Regression model, by calculating the accuracy and the classification report for the Logistic Regression model.

4.4.3 **Naive Bayes**

An instance of the MultinomialNB class from scikit-learn is created and initialized through the Multinomial Naive Bayes classifier. This classifier is chosen for text classification tasks and is suitable for dealing with features representing word frequencies, so it is usually used for natural language processing applications.

The model is then trained using the 'fit' method. The two inputs are, first, the representation of the training data, which has been transformed into a numerical format using the TF-IDF vectorization technique. It contains the features (textual data) the model will learn. Second, the representation of the corresponding target labels for the training data indicates the correct class labels for the training examples.

After training, the Multinomial Naive Bayes model is prepared to make predictions on new, unseen data. The code employs the 'predict' method to generate predictions on the test data.

4.4.4 **k-Nearest Neighbors**

The target labels are converted to NumPy arrays for the K-Nearest Neighbors (KNN) model. The model is built and trained with the TF-IDF vectorized text data from the training set. The number of neighbors, a hyperparameter for the KNN algorithm, is set to 5. After training, the model makes predictions on the test set.

In this algorithm, the classification of data points is based on their proximity to neighboring data points in the training set, so it is set to 5. Hence, the model considers the five nearest neighbors when making predictions. The model is then trained using the 'fit' method. It takes two inputs in the same way as the previous algorithms.

After training, the KNN model is prepared to make predictions on new, unseen data. The code employs the 'predict' method to generate predictions on the test data. Finally, the code proceeds to evaluate the performance of the KNN model.

5 RESULTS

After training, the model is employed to make predictions on the test set, and the accuracy of the model is calculated. The accuracy score indicates how well the model performs on the unseen data. A classification report is also displayed, offering a detailed breakdown of metrics such as precision, recall, and F1-score for each class.

5.0.1 Support Vector Machine

The Support Vector Machine (SVM) model results are analyzed to evaluate its performance in classifying text data into two distinct categories. The overall accuracy achieved is 0.903, indicating that the model correctly classified approximately 90.3% of the instances in the test dataset.

The model demonstrates robust performance for the majority class (Class 0), achieving high precision (0.93) and recall (0.96). This means that the model accurately identifies instances belonging to the majority class while maintaining a low rate of false positives.

However, for the minority class (Class 1), the model's performance is comparatively lower. It presents a lower precision (0.63) and recall (0.52), resulting in an F1-score of 0.57. In imbalanced datasets like this, achieving a balance between precision and recall can be challenging. Nevertheless, the model still demonstrates a reasonable ability to classify instances of the minority class.

The weighted average metrics consider the overall model performance, giving more weight to the majority class. The weighted average precision, recall, and F1-score are all approximately 0.90, indicating the model's overall solid effectiveness in handling imbalanced data.

Two key adjustments significantly contributed to the model's improved performance, particularly for the minority class. First, including n-grams in the TF-IDF vectorization process enriched the representation of the text data, capturing meaningful word combinations alongside individual words. This enhancement allowed the model to discern more complex patterns in the text.

Second, using the class weight parameter with a 'balanced' setting played a crucial role. It adjusted the class weights during training, giving higher importance to the minority class. This mitigation strategy effectively reduced the impact of class imbalance and improved recall for the minority class.

The confusion matrix reveals the performance of a machine learning model designed to predict the outcomes of cases related to Article 8 of the European Convention on Human

Rights. In this matrix, the vertical axis represents the true labels, and the horizontal axis represents the labels predicted by the model. A label of '1' corresponds to cases where a violation of Article 8 is present, while a label of '0' denotes cases without such a violation.

The model demonstrated a propensity to identify violations of Article 8 with many true positives, amounting to 839 instances where the model's predictions aligned with actual violations. This indicates a strong ability of the model to detect violations where they exist. Conversely, the model identified fewer true negatives, with 64 instances where both the model's predictions and the actual cases concurred on the absence of a violation.

However, the model presented errors. It produced 38 false positives, where the model incorrectly predicted a violation of Article 8 when there was none. This represents a relatively small proportion of the predictions, suggesting a cautious approach by the model in predicting violations. The 59 false negatives are more critical, where the model failed to identify actual violations. This type of error is of particular concern as it represents instances where the model overlooked actual infringements on the rights protected under Article 8.

Overall, the model's performance indicates a higher confidence in predicting violations over non-violations, as reflected by the higher number of true positives compared to true negatives. The prediction errors, especially the false negatives, suggest areas where the model could be improved by refining its sensitivity to the nuanced features that characterize Article 8 violations.

The balance between sensitivity (true positives) and specificity (true negatives) and considering the costs of false positives and false negatives is crucial in legal judgment prediction, where the stakes of misclassification are high.

The accuracy, precision, recall, and F1 score derived from the confusion matrix explain the model's performance, providing a quantitative basis for evaluating its effectiveness in legal judgment prediction.

The ROC curve of the SVM classifier on the test data illustrates the trade-off between the true positive rate (TPR) and false positive rate (FPR) at various thresholds. A steep initial ascent indicates the model achieves high sensitivity without incurring many false positives, underscoring its effective class differentiation.

With an AUC of 0.89, the model's discriminative ability is strong, nearing the ideal AUC of 1 and far surpassing the no-skill line at 0.5. Although the curve's proximity to the top left corner reflects high accuracy, it falls short of the perfect classification mark, suggesting potential for further refinement.

Overall, the SVM classifier's ROC curve indicates robust performance, balancing sensitivity and specificity effectively. However, the true value of the model's performance would be context-dependent, depending on the relative costs of false positives and false

negatives in the application domain.

5.0.2 **Logistic Regression**

The performance of the Logistic Regression model on the test set was evaluated, and an accuracy of 88.8% was achieved. The precision, recall, and F1-score metrics were calculated for two classes, as '0' and '1'.

For class '0', which comprised the majority of the test cases (877 out of 1000), high precision (95%) and recall (92%) were observed, resulting in a robust F1-score of 94%. Class '1', with fewer instances (123 out of 1000), exhibited a precision of 54% and a recall of 67%, culminating in a lower F1-score of 59%. The macro-averaged F1-score across both classes was 76%, indicating a reasonably strong performance of the model across classes of varying sizes.

The weighted average F1-score, which accounts for the imbalance in class distribution, was recorded at 89%, closely aligning with the model's overall accuracy. These metrics suggest that while the model is highly adept at predicting the majority class '0', its performance on the minority class '1' is less accurate, although still significant given the context of the class imbalance.

5.0.3 **Naive Bayes**

The Naive Bayes model, when applied to the test dataset, achieved an overall accuracy of 87.7%. This metric suggests that the model correctly predicted the outcome in approximately 88 out of every 100 cases. However, a detailed examination of the classification report reveals significant disparities in model performance across the two classes.

For class '0', which represents most of the dataset with 877 instances, the Naive Bayes model demonstrated high precision (88%) and an even higher recall (100%). The recall of 100% indicates that the model successfully identified all instances of class '0' in the test set. Consequently, the F1-score for this class was a robust 93%, reflecting the model's strong performance in predicting class '0'.

In strong contrast, class '1' performance, which had 123 instances, was markedly different. The model presented a precision and recall of 0%, indicating a complete inability to correctly identify any instances of this class. This resulted in an F1-score of 0% for class '1', suggesting that the model's predictive capacity was effectively non-existent for this particular class.

The macro average F1-score, which treats both classes equally regardless of size, was 47%. This score is significantly lower than the weighted average F1-score of 82%, which accounts for the class imbalance by giving more weight to the majority class '0'.

The weighted average thus appears more favorable due to the model's high accuracy in predicting the more prevalent class.

These results highlight a critical limitation of the Naive Bayes model in this specific application. While it is highly effective in predicting the majority class, it fails to recognize data points of the minority class. This imbalance in predictive performance raises concerns about the model's applicability in scenarios where accurate detection of both classes is crucial.

5.0.4 **K-Nearest Neighbors**

The K-Nearest Neighbors (KNN) model achieved an accuracy of 88.4% on the test dataset, indicating that it correctly predicted the class labels for 884 out of 1000 data points. This level of accuracy shows a reasonably high overall performance of the model in classifying the test data.

In the detailed analysis of the classification report, two classes were evaluated: class '0' and class '1'. Class '0', with 877 instances, exhibited a high precision of 91% and a higher recall of 97%. These values led to an F1-score of 94%, suggesting the model was highly effective in correctly identifying and classifying class '0' data points. The high recall indicates that the model captured the majority of actual class '0' data points, while the high precision reflects the accuracy of these classifications.

For class '1', which had a smaller representation in the dataset with 123 data points, the model demonstrated a lower precision of 55% and a notably lower recall of 29%. The resulting F1-score for this class was 38%, substantially lower than that of class '0'. This disparity indicates that while the model was relatively accurate in its positive predictions for class '1', it failed to identify many actual class '1' data points, as evidenced by the low recall.

The macro average F1-score, which considers both classes equally, was 66%, reflecting the model's combined performance across both classes. The weighted average F1-score, at 87%, considers the uneven class distribution and skews higher due to the model's more robust performance on the more prevalent class '0'.

Overall, the KNN model showed a solid ability to classify the majority class but was less effective in identifying the minority class. This pattern suggests a potential area for improvement, particularly in enhancing the model's sensitivity to the less represented class in the dataset.

6 CONCLUSIONS

The present work has empirically studied the efficacy of machine learning models in the predictive analysis of Article 8 violations within the European Court of Human Rights. The Support Vector Machine (SVM) was the most effective model, as well as the one with the highest accuracy, at 90.3%. Additionally, it has an AUC of 0.89, indicating its discriminative solid ability between violations and non-violations. The model performed excellently for the majority class, but there is room for improvement regarding the minority class.

The Logistic Regression model also performed well, as it correctly predicted the outcome of Article 8 violation cases 88.8% of the time when tested on unseen data. While it has robust precision and recall in the majority class, it presented less precise results in the minority class. Despite this, the average performance across both cases has an excellent result, as the macro-averaged F1-score reflects a reasonably strong performance across both classes.

In contrast, while achieving a high overall accuracy of 87.7%, the Naive Bayes model faced limitations in identifying the minority class, as evidenced by an F1-score of 0% for this group. This highlights the model's struggle in handling class imbalance, an issue that is critical in the context of legal judgment prediction. The K-Nearest Neighbors (kNN) model displayed an exemplary overall accuracy of 88.4% but, like the others, showed a disparity in its ability to identify the minority class, with a notably lower recall and F1-score.

The author of this work considers that SVM, as they have the highest accuracy and are also characterized for having a very straightforward classification technique, should be the preferred model for the task. SVM is also the most transparent because of this straightforward way of working. Thus, for regulation and ethics purposes, they can provide an easier, more justified, and more comprehensible process.

Considering all findings, the study indicates that although machine learning can significantly support predicting court outcomes, achieving a balance between precisely identifying the majority and minority classes is an ongoing challenge.

Thus, the limitations of this study are primarily associated with class imbalance in the dataset regarding article 8. While the SVM model showed a strong ability to predict violations, the lower precision and recall for the minority class means that it could be improved. Similarly, the Naive Bayes model's inability to identify the minority class highlights the need for better data sampling and model training approaches. This is also a limitation given the serious implications of incorrect predictions, such as wrongly

identifying or missing a legal violation.

Future research could explore several avenues to overcome these limitations. One potential improvement involves using advanced techniques, such as deep learning to enhance model robustness and accuracy. Moreover, expanding the dataset and applying these models to more diverse legal articles could enhance the generalization of the findings. It could evaluate whether the patterns observed for Article 8 are consistent across other articles and whether machine learning models could offer similar predictive capabilities in these new contexts.

Additionally, researching this work's social, legal and ethical implications would constitute a very important aspect to find out. Legal scholars, judges and applicants can be supported through this work to get a more rapid and efficient overview of a case that might violate Article 8. However, this study does not pretend to substitute human professionals but facilitates processing the amount of work involved in each case.

Finally, this study contributes to the intersection of law and artificial intelligence, showing a transformative step towards a more informed and efficient judicial process. By integrating these predictive models, the legal field can benefit from a computational and quantitative approach to case analysis, supporting law enforcement.

REFERÊNCIAS

ALETRAS, N. *et al.* Predicting judicial decisions of the european court of human rights: A natural language processing perspective. **PeerJ Computer Science**, v. 2, 2016.

ALPAYDIN, E. **Machine Learning: The New AI**. [*S.l.: s.n.*]: MIT Press, 2016.

CASE of Mockutė v. Lithuania. This judgment has become final under Article 44 § 2 of the Convention. It may be subject to editorial revision.

CHALKIDIS, I. *et al.* Neural legal judgment prediction in english. *In*: **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**. [*S.l.: s.n.*], 2019. p. 4317–4323. Available at: <https://doi.org/10.18653/v1/P19-1424>.

CHALKIDIS, I. *et al.* Paragraph-level rationale extraction through regularization: A case study on european court of human rights cases. *In*: **Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**. [*S.l.: s.n.*], 2021. Preprint.

Conselho Nacional de Justiça. **Ombudsman 10 Years: Slow Justice Still the Main Cause of Complaint**. 2023. Last accessed: [September 5, 2023]. Available at: <https://www.cnj.jus.br/ouvidoria-10-anos-lentidao-da-justica-ainda-e-o-motivo-de-maior-reclamacao/>.

Council of Europe: European Court of Human Rights. **Guide on Article 8 of the European Convention on Human Rights - Right to respect for private and family life**. 2020. Accessed: December 17, 2023. Available at: <https://www.echr.coe.int/documents/d/echr/guide_art_8_eng>.

Council of Europe/European Court of Human Rights. **Guide to the Case-Law of the European Court of Human Rights - Data Protection**. [*S.l.: s.n.*], 2022. Updated on 31 August 2022.

CUI, J. *et al.* **A Survey on Legal Judgment Prediction: Datasets, Metrics, Models and Challenges**. 2022.

EBOBRAH, S. T. International human rights courts. *In*: ROMANO, C. P. R.; ALTER, K. J.; SHANY, Y. (ed.). **The Oxford Handbook of International Adjudication**. Oxford Academic, 2014. Accessed June 3, 2023. Available at: <https://doi.org/10.1093/law/9780199660681.003.0011>.

European Commission. **European Court of Human Rights (ECtHR)**. Accessed: 2023. Accessed on June 5, 2023. Available at: <https://home-affairs.ec.europa.eu/networks/european-migration-network-emn/emn-asylum-and-migration-glossary/glossary/european-court-human-rights-ecthr_en>.

GUENTHER, N.; SCHONLAU, M. Support vector machines. **The Stata Journal**, v. 16, n. 4, p. 917–937, 2016.

GüRSES, S.; ALAMO, J. M. del. Privacy engineering: Shaping an emerging field of research and practice. **IEEE Security Privacy**, v. 14, n. 2, p. 40–46, 2016.

HASTIE, T. *et al.* **The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.)**. [*S.l.: s.n.*]: Springer, 2021.

JENA, M. *et al.* Ensuring data privacy using machine learning for responsible data science. **Advances in Intelligent Systems and Computing**, p. 507–514, 2020.

JURAFSKY, D.; MARTIN, J. H. **Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition**. Second. [*S.l.: s.n.*]: Prentice Hall, 2008. University of Colorado Boulder.

KESARI, A. *et al.* **A three-step guide to Training Computational Social Science Ph.D. students for academic and non-academic careers**. 2022. Available at: <https://doi.org/10.31235/osf.io/kgjn2>.

KHURANA, D. *et al.* **Natural Language Processing: State of the Art, Current Trends, and Challenges**. 2022. 3713–3744 p.

LAGE-FREITAS, A. *et al.* Predicting brazilian court decisions. **PeerJ Computer Science**, v. 8, 2022.

LEE, R. **Natural Language Processing: A Textbook with Python Implementation**. [eBook], 2023. Accessed: December 17, 2023. Available at: <https://doi.org/10.1007/978-981-99-1999-4>.

LEHR, D.; OHM, P. Playing with the data: What legal scholars should learn about machine learning. **U.C. Davis Law Review**, v. 51, p. 653–655, 2017.

LI, Y. *et al.* Adapting svm for natural language learning: A case study involving information extraction. *In*: **Deterministic and Statistical Methods in Machine Learning**. [*S.l.: s.n.*], 2004. p. 319–339.

LINDHOLM, A. *et al.* **Machine Learning: The First Course for Engineers and Scientists**. [*S.l.: s.n.*]: Cambridge University Press, 2021.

MEDINA, J. C. **Multiplatform Analysis of Political Communication on Social Media**. 2021. Tese (Doutorado) — Technische Universität München, 2021. Dissertation submitted on 05.07.2021 and accepted by the Faculty of Informatics on 12.11.2021.

MEDVEDEVA, M. *et al.* Using machine learning to predict decisions of the european court of human rights. **Artificial Intelligence and Law**, v. 28, n. 2, p. 237–266, 2019.

NIEMIETZ v. Germany (1992).

NISSENBAUM, H. **Privacy in Context: Technology, Policy, and the Integrity of Social Life**. [*S.l.: s.n.*]: Stanford University Press, 2010.

NOBLE, W. S. What is a support vector machine? **Nature Biotechnology**, v. 24, n. 12, p. 1565–1567, 2006.

Ordem dos Advogados do Brasil. **Brasil tem 1 advogado a cada 164 habitantes; CFOAB se preocupa com qualidade dos cursos jurídicos**. 2022. Last accessed: September 5, 2023. Available at: <https://www.oab.org.br/noticia/59992/brasil-tem-1-advogado-a-cada-164-habitantes-cfoab-se-preocupa-com-qualidade-dos-cursos-juridicos>.

PEDREGOSA, F. *et al.* Scikit-learn: Machine learning in python. **Journal of Machine Learning Research**, v. 12, n. Oct, p. 2825–2830, 2011.

PETROV, A. Brazil has the highest ratio of lawyers per inhabitant in the world. **The Rio Times**, 8 2022. Last accessed: September 5, 2023. Available at: <https://www.riotimesonline.com/brazil-news/brazil/brazil-has-the-highest-ratio-of-lawyers-per-inhabitant-in-the-world/>.

RAO, P. R. M.; KRISHNA, S. M.; KUMAR, A. S. Privacy preservation techniques in big data analytics: A survey. **Journal of Big Data**, v. 5, p. 33, 2018.

ROMANIA, B. v. 2017.

RUSSELL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. Fourth, global edition. [*S.l.: s.n.*]: Pearson, 2022. (Series in Artificial Intelligence).

SANTOSH, T. *et al.* Deconfounding legal judgment prediction for european court of human rights cases towards better alignment with experts. *In*: **Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing**. [*S.l.: s.n.*], 2022. Preprint.

TRSTENJAK, B. *et al.* Knn with tf-idf based framework for text categorization. **Procedia Engineering**, v. 69, p. 1356–1364, 2014.

UCAK, U. V.; ASHYRMAMATOV, I.; LEE, J. Improving the quality of chemical language model outcomes with atom-in-smiles tokenization. **Journal of Cheminformatics**, v. 15, n. 1, 5 2023. Available at: <https://doi.org/10.1186/s13321-023-00725-9>.

VIJAYARANI, S.; JANANI, R. Text mining: Open source tokenization tools – an analysis. **Advanced Computational Intelligence: An International Journal (ACII)**, v. 3, n. 1, 1 2016.

VéLIZ, C. Privacy and digital ethics after the pandemic. **Nature News**, Nature Publishing Group, 2021. Accessed: February 12, 2023. Available at: <https://www.nature.com/articles/s41928-020-00536-y>.

VéLIZ, C. **Privacy is Power: Why and How You Should Take Back Control of Your Data**. Brooklyn: Melville House, 2021.

WASSERMAN, L. **All of Statistics: A Concise Course in Statistical Inference**. [*S.l.: s.n.*]: Springer, 2013.

WEBB, G. Naïve bayes. *In*: **Springer eBooks**. [*S.l.: s.n.*], 2016. p. 1–2.

ZHANG, H. *et al.* Contrastive learning for legal judgment prediction. **ACM Transactions on Information Systems**, 2023.