INSTITUTO
UNIVERSITÁRIO
DE LISBOA

Automatic Classification of Complaints from Public Administration

Francisco Miguel Silva Caldeira

Master in Computer Engineering

Supervisor:
Doctor Luís Miguel Martins Nunes, Associate Professor,
Iscte – Instituto Universitário de Lisboa

Co-Supervisor:
Doctor Ricardo Daniel Santos Faro Marques Ribeiro, Associate Professor, Iscte – Instituto Universitário de Lisboa

October, 2022

TECHNOLOGY
AND ARCHITECTURE

Department of Information Science and Technology

Automatic Classification of Complaints from Public Administration

Francisco Miguel Silva Caldeira

Master in Computer Engineering

Supervisor:
Doctor Luís Miguel Martins Nunes, Associate Professor,
Iscte – Instituto Universitário de Lisboa

Co-Supervisor:
Doctor Ricardo Daniel Santos Faro Marques Ribeiro, Associate Professor, Iscte – Instituto Universitário de Lisboa

October, 2022

Acknowledgment

Quero agradecer a todas e todos que me ajudaram e acompanharam durante este percurso. Com um especial nota para os meus pais, Conceição e Paulo, que me suportaram nos momentos altos e baixos desta da vida. Quero agradecer à minha namorada Mariana por toda a motivação e assim como todo o apoio que me deu. Deixo um especial agradecimento à minha gata Zelda pela companhia e por todas as vezes que tentou escrever por mim.

Aos meus orientadores, os Professores Luís Nunes e Ricardo Ribeiro, cujo o auxílio foi indispensável para o progresso da dissertação bem como do artigo dela extraído. Agradeço toda a ajuda do pessoal do IGSJ cuja parceria possibilitou a realização desta dissertação.

Aos meus colegas da Sky, especialmente ao Diogo, por me aturarem bem como todas as discussões que ajudaram progredir e fazer sentido muitos dos tópicos. Aos meus colegas Francisco Barros, Nuno Pires e ao meu colega da Sky Pedro Silva, com quem tive a oportunidade de realizar muitos dos trabalhos para chegar à concretização desta dissertação.

This work was partly funded through national funds by FCT - Fundação para a Ciência e Tecnologia, I.P. under project UIDB/04466/2020 (ISTAR)

Resumo

A classificação de texto é uma área de estudo em aberto, dependendo do problema dos dados disponíveis e estudo em questão, o melhor método nem sempre é mesmo. Dentro da área da inteligência artificial No caso das empresas a classificação de queixas (como neste trabalho) ou mesmo de incidentes é uma tarefa que ainda requer muito trabalho manual. Neste trabalho vai ser abordada a classificação automática de queixas recebidas por uma instituição pública. No processo de tratamento das queixas a classificação é parte do grande panorama e a sua automatização permite acelerar muito os processos manuais que são actualmente usados. Neste contexto, foram trabalhados os sumários das queixas e as técnicas usadas para aplicar modelos de classificação automática. O conjunto de dados é consideravelmente pequeno e apresenta um grande desequilíbrio na distribuição das classes, sendo que as três maiores têm perto de 95% dos dados. Para colmatar este problema foram analisadas duas abordagens: classificação em duas etapas e aumento do conjunto de treino com base em traduções dos sumários. Neste contexto foram usados alguns modelos de classificação como k-NN, SVM, Naïve Bayes, boosting e BERT. Usando modelos treinados com os sumários foi também realizada uma experiência de classificação dos textos completos das queixas. Apesar dos resultados serem piores do que os obtidos usando o dados resumidos, estes apresentam alguma taxa de sucesso, especialmente para classificação da classe mais frequente. Com base neste trabalho foi possível concluir que a classificação das classes com menos representação é um desafio, mas através de técnicas de aumento do conjunto de treino é possível melhorar substancialmente o resultado obtido. Também utilizar uma estratégia de classificação multietapa permite melhorar os resultados obtidos. Os melhores modelos para a classificação foram SVM e BERT.

Palavras-chave: Classificação de Texto, Processamento de Linguagem Natural, Aprendizagem Automática, BERT

Abstract

Complaint management is a problem faced by many organizations that is both vital to customer satisfaction and retention, while being highly dependent on human resources. This work attempts to tackle a part of the problem, by classifying summaries of complaints using machine learning models in order to better redirect these to the appropriate responders. To solve the aforementioned problem text mining, and more specifically natural language processing, were used alongside machine learning algorithms for automatic classification. The main challenge of this task is related with the diverse set of characteristics real world datasets have, in this case being small and highly imbalanced. This can have a big impact on the performance of the classification models. The dataset analyzed in this work suffers from both of these problems, being relatively small and having labels in different proportions the three most common labels account for around 95% the dataset. In this work, two different techniques are analyzed: multistage classification with for classifying the more common labels first and the remaining on a second step; and, generating new artificial examples for some classes via translation into other languages. The classification models explored were the following: k-NN, SVM, Naïve Bayes, boosting, and Deep Learning approaches, including transformers. Although, in general using summaries leads to better results, we also experimented with the full documents. Using the models trained with the summarized documents the classification of the full documents. Even though the results were not on par with the summarized dataset the experimented presented good results for signaling the most common label of the documents. We conclude that although, as expected, the classes with little representation are hard to classify, the techniques explored helped to boost the performance, especially in the classes with a low number of elements. SVM and Transformer-based models outperformed their peers.

Keywords: Text Classification, Natural Language Processing, Machine Learning, BERT

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Artificial intelligence and machine learning have become deeply engraved in everyone's daily lives in one way or another. Having processes that can interpret and transform data like a human could provide some automation of a few text analysis tasks. One such task is the classification of user generated textual data, for example data from tweets or comments and even emails. The classification tasks are generally performed using Machine Learning, Data Mining and Natural Language Processing methods. Resorting to models of automatic classification can easily automate a long and hard task providing an improvement of the existing process, a better use of the human resources while even decreasing the human induced error rate. This is the case for Portuguese public services.

Specifically referring to the Portuguese Ministry of Justice a separate entity that oversees the work of the ministry, known as *Inspecção Geral dos Serviços da Justiça*, General Inspection of Justice Services. All citizens are able to raise questions, make complaints and inquire every action made by entities tutored by the ministry. In the recent years with, use of the recent developments in technology allowed governments to expedite and ease the access of the citizens to some of the services provided by the government. This resulted in a increase of requests made to the government entities. For example, during Covid-19 pandemic most rules were temporarily changed to address the epidemic and enabling a more digital approach. Due to this situation some public entities received a large amount of alerts and complaints. Mostly referring the fact that the mandates for Covid-19 protection were not being thoroughly fulfilled. As an example ASAE(Autoridade de Segurança Alimentar e Económica)[1] received a huge amount of complaints related to the pandemic. This led to an increase in demand such that the workers were unable to keep up the pace. They were required to read and sort the requests that arrived daily and dispatch them to the correct entity while even receiving wrong complaints. The whole process of the manual sorting and processing is slow and tedious, requiring a lot of human resources, being prone to human error.

This work will focus on the field of Natural Language Processing (NLP), an area of Artificial Intelligence that studies the different approaches for computers to handle and process human made languages. Using NLP, the contents of the data are transformed into a format that is understandable by computers and will enable the data to be classified.

[1]https://www.jn.pt/justica/asae-recebeu-quase-duas-mil-queixas-em-14-dias-relacionadas-com-covid-19-11995914.html

NLP is, in itself, a challenge since most languages were developed for human-human interaction and for computers to understand and handle human language some transformations are necessary.

1.1. Motivation

Justice public services have a dedicated entity to manage the complaints the users of the Portuguese justice services have, the entity is *Inspecção Geral dos Serviços da Justiça* (General Inspection of Justice Services), IGSJ[2]. Citizens can submit complaints respecting to the services provided by fourteen entities that are tutored by the Ministry of Justice, Table 1.1.

Table 1.1. IGSJ tutored entities

Entity	Acronym
Centro de Estudos Judiciários (Center for Judicial Studies)	CEJ
Comissão de Programas Especiais de Segurança (Special Security Programs Commission)	CPES
Comissão de Proteção às Vítimas de Crimes (Commission for the Protection of Victims of Crime)	CPVC
Comissão para o Acompanhamento dos Auxiliares da Justiça (Commission for the Monitoring of Justice Assistants)	CAAJ
Direção-Geral da Administração da Justiça (Directorate-General for the Administration of Justice)	DGAJ
Direção-Geral da Política de Justiça (Directorate-General for Justice Policy)	DGPJ
Direção-Geral de Reinserção e Serviços Prisionais (Directorate-General for Reinsertion and Prison Services)	DGRSP
Inspeção-Geral dos Serviços de Justiça General Inspection of Justice Services	IGSJ
Instituto de Gestão Financeira e Equipamentos da Justiça, I. P. (Institute of Financial Management and Justice Equipment)	IGFEJ
Instituto dos Registos e do Notariado, I. P. (Institute of Registries and Notaries)	IRN
Instituto Nacional da Propriedade Industrial, I. P. (National Institute of Industrial Property)	INPI
Instituto Nacional de Medicina Legal e Ciências Forenses, I. P. (National Institute of Legal Medicine and Forensic Sciences)	INMLCF
Polícia Judiciária (Judiciary Police)	PJ
Secretaria-Geral Ministério da Justiça (General Secretariat Ministry of Justice)	SGMJ

However, manual complaint analysis is a time-consuming and costly task, hence the desire to achieve better processing and less repeated work.

[2]https://igsj.justica.gov.pt/

The complaints can be submitted in a several different ways(online form, email, post office or in person), and in 2020, 1853 of the 2223 complaints received were submitted using the form, 253 via email, 114 post office and 3 were filled personally, refer to Figure 1.1. Out of the 2223 complaints received in 2020 around 10% were erroneously submitted for IGSJ. Processing all the complaints poses a great challenge, each complaint must be handled with the appropriate diligence, reviewed, and redirected to the corresponding entity.

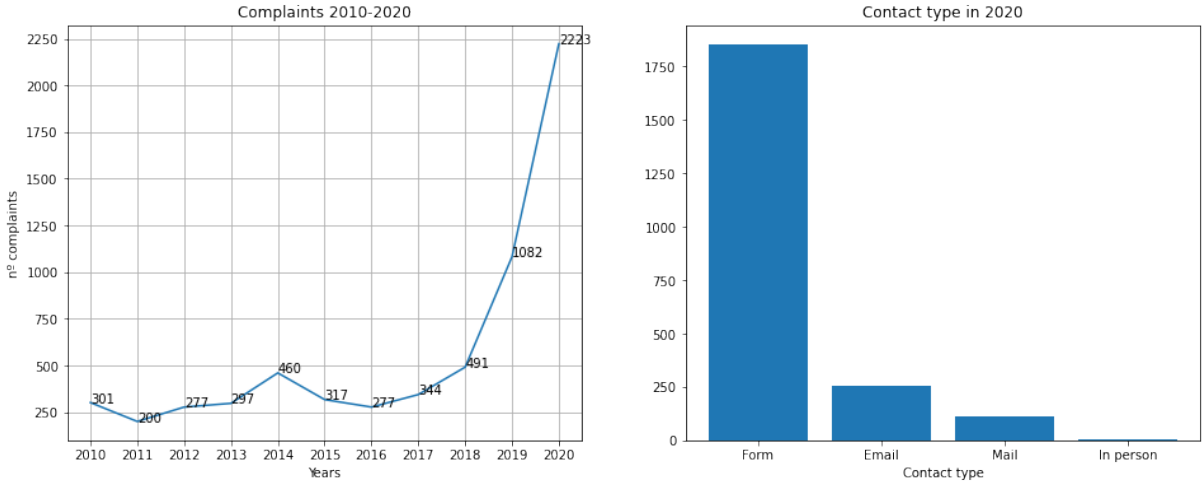

FIGURE 1.1. Complaints evolution from 2010 to 2020 and 2020 overview

1.2. Objective

Initially the goal of this dissertation was building a process to automatically summarize and classify the complaints received by IGSJ. After the initial analysis of the dataset, it was found that it only included the previously summarized complaints. Thus the goal shifted to the classification of the summarized complaints by the institution. Although the raw complaints were later available to analyse, they were used to further evaluate the trained classification models.

All the complaints received by IGSJ were summarized by a worker and labeled with the corresponding action. Then, analyzed and it is decided if a response can be created and be sent back to the complainee or if it is forwarded entity it is being target of a complaint, for handling the matter of the complaint. A visual example can be found in Figure 1.2.

The main goal of this work is to develop a classifier that will be able to process the text data and decide if the text body is related to any of the 14 entities, Table 1.1. All the experiments will also be evaluated in order to understand which is the best approach for the classification task. This work will try to answer the following questions.

- How to handle the imbalance in the datasets, and what methods are more suitable for summarized datasets?
- How do classification methods perform when using summarized data?

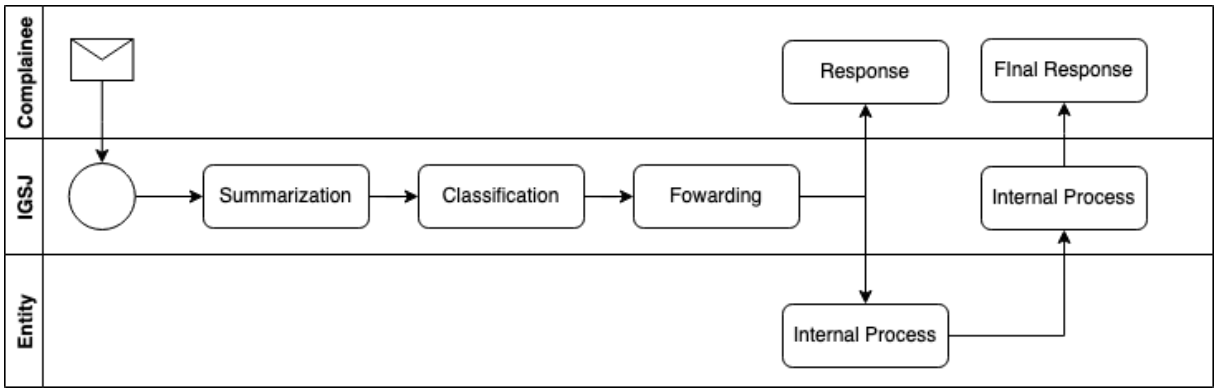

FIGURE 1.2. Flow of the complaints handling

- Are models trained with summarized data useful for classification of the original dataset?

Regarding the first question label imbalance is common issue found in real world data and for this issue two techniques are considered, a multistage classification route, and artificially augmenting the documents. Both techniques are used individually and combined. To conclude, the performance is compared to evaluate the outcome of the different approaches used to handle imbalance. For the second question, we evaluate the usage of shorter texts, like summaries, as the input of classification models. For the data under analysis the performance of the will provide insights over this matter. As for the final question we investigate if training the models with texts summaries can be used for classifying the full texts. The classification models will be trained with summaries and will be used for classifying the full complaints.

1.3. Methodology

This thesis development will adopt Cross-Industry Standard Process for Data Mining (CRISP-DM) methodologies guidelines. CRISP-DM "provides a framework for carrying out data mining projects, the process model is useful for planning, documentation and communication" [1].

- **Business Understanding** - Understand the workflow of IGSJ and correlated with possible outcomes of the classification models.
- **Data Understanding** - Getting familiar with the data and discovering features of documents, in this case the summaries and later the full complaints.
- **Data Preparation and Modeling** - The transformations required in the texts removing frequent words and augmenting the complaints using the translated techniques. Aggregating in new classes to deal with the imbalance and the multistage experiment.
- **Evaluation** - Reviewing the performance of the models and understanding the results. With the new insights iterate over the preparation and modeling to test new results.

- **Deployment** - The final product is this document with the results of the experiments.

1.4. Document Outline

This document will proceed with Chapter 2.1: a review of the some fundamental concepts of text mining and text classification. It is followed by Chapter 2 where the state of art related to classification models, Portuguese text processing and dealing with imbalanced data set, is presented. Then, in Chapter 3, the available data is presented and explored, providing some initial insights. Afterwards, in Chapter 4 and Chapter 5, the steps taken for processing the data and the models used for classification are addressed, and the results of the experients are detailed and analyzed. Some additional experiments were also done with the raw complaints, classification using the models traing with the summarized texts. Finally in Chapter 6 presents the final remarks and future work.

CHAPTER 2

Text Mining Overview

In this chapter we present a brief overview of text mining in portugues and a review of the literature explored for dissertation. For the research of the current state the art relating to natural language processing and text classification the following databases of articles were used: Web of Science[1], Scopus[2] and Google Scholar[3]. For searching the databases the title, abstract, and keywords of the articles were queried with *"text classification, natural language processing, complaint, user generated data, Portuguese"*.

2.1. Fundamentals of Text Mining

In this section a brief overview of text mining fundamentals inherent to the document objective. A short review of text representation techniques is introduced along side a brief explanation of the models used for classification. The metrics to evaluate each output of the model are also briefly explained.

2.1.1. Text Representation

Text data can be regarded as unstructured, weakly-structured or semi-structured and can have any number of features. The usual initial pre-processing is removing special characters, numbers and stop words and representing the words with is lemma or stem. Representing the text in a different method can also be useful, using a vector to represent each document. With the collection of documents being then represented in a sparse matrix. Choosing the correct heuristic to transform the text into a vector is critical to ensure the performance of text mining models. Some models will have its representation form in the form of huge sparse matrices however depending on the task at hand the best representation will vary greatly.

2.1.1.1. *TF and TF-IDF* Term Frequency, Inverse Document Frequency, and their combination TF-IDF are the most popular weighting algorithms to transform text into vector. Term Frequency is in general, defined as a function of number of times a token appears in a document and Inverse Document Frequency is defined in Equation 2.1, where D is the number of documents in the data set. IDF can explain how relevant a word is in a document in regards to the data set.

$$IDF(t) = log\left(\frac{D}{df}\right) \tag{2.1}$$

[1]https://clarivate.com/webofsciencegroup/solutions/web-of-science/
[2]https://www.scopus.com/
[3]https://scholar.google.com/

The main value of TF-IDF is that words with high term frequency should receive high weight unless they also have high document frequency.

TF-IDF is the combination of both TF and IDF and the definition can be seen in Equation 2.2

$$TF\text{-}IDF(t,d) = TF(t,d) \times DF(t) \tag{2.2}$$

A word that is present in few documents will present a weight much higher than a word that appears in almost all. The lowest weight of 1 is assigned to terms that occur in all the documents. Using this method for transforming text into vector will make use of the entire corpus tokens and in general leads to huge sparse matrices.

2.1.1.2. *Word Embeddings* Another model of text representatin is Word Embeddings. in contrast to TF-IDF using word embeddings for representation will lead to dense matrices representations. Words that occur in similar documents tend to have similar meanings. In this representation technique the words are represented using an embedding that evaluates its context in the document, captures the word meaning in the different contexts. This method will consider all words of a document as vectors of a multidimensional semantic spaces, were the coefficients are derived from the context (other words) of a given word. In the end a sentence will be a combination of vectors in a matrix.

This model of representation captures the properties of each word in a real valued vector (dense). This vectorsare also very important for representing word similarity.

2.1.2. Classification Models

In this section a brief description of the classification models used in this dissertation task can be found.

2.1.2.1. *Classic Models*

Naïve Bayes This model is based on Bayes Theorem.

$$P(y|x_1, x_2, \ldots x_n) = \frac{P(y) * P(x_1, x_2, \ldots x_n|y)}{P(x_1, x_2, \ldots x_n)}$$

To evaluate a document and predict the class it belongs, the probability of each class, $P(y)$, is pre-calculated. Then using a maximum likehood estimator the class the maximizes the function is then obtained.

$$\hat{y} = \arg\max_y \Pi_{i=1}^{n} P(x_i|y)$$

SVM Support Vector Machines (SVM) is a supervised learning model that can used for classification tasks. SVM algorithm finds the maximum marginal hyperplane that will separate the data into classes of similiar content in a multi dimensional space. It separates data points into groups with similar properties.

In any given dataset several possible hyperplane separations can be found. SVM finds the optimal separation that best fits the data. Finding the optimal solution is an optimization problem and can be solved using know techniques.

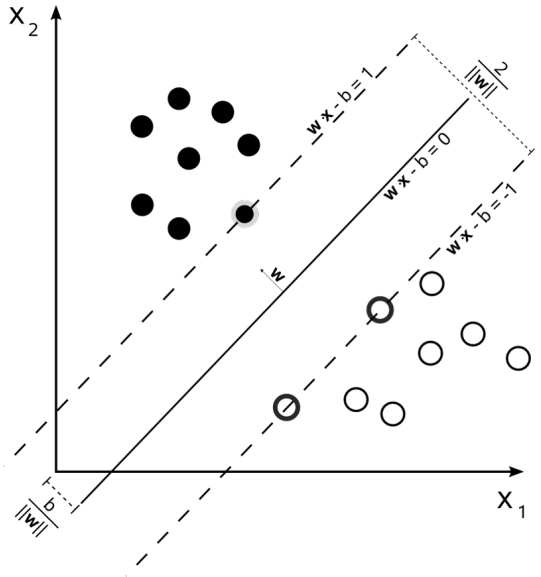

FIGURE 2.1. Example of a SVM hyperplane

k-Nearest Neighbors The k-Nearest Neighbors (KNN) is an algorithm that classifies data points by finding the closest k nearest data points in the training data and attributing the same label as the closest k data points. The distance metric for finding the distance between data point can be any measure that is valid in a given topological space. Some examples are Euclidean distance, Manhattan distance or the L^p norm. The data point is given the label of the highest scoring classes among the k neighbors. A variation of this algorithm was adapted [2] by attributing weights to larger groups of data points to avoid skewing the classification.

2.1.2.2. *Transformer Based Models* A transformer model is defined as a neural network that is able to detect relations between elements in a given data structure [3], an example of can be a sentence represented using embeddings. BERT is the name given to the pretrained model develop at Google for Bidirectional Encoder Representations from Transformers. It is a bidirectional pretrained transformer model used on NLP tasks trained on the Toronto Book Corpus and Wikipedia. It is a model used in the most common language processing task. Since the data for this work is only available in Portuguese the model used was Multilangual-BERT(mBERT for short). This model was trained over 104 different articles available on Wikipedia. BERT difference from other models is the approach to analyze words from both direction in a text in contrast to the other models that were available only evaluated in a single direction.

BERTs key technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modelling.

2.1.3. Text Classification Literature Review

Text representation and feature extraction are the basis of some models of text classification. Textual data has a huge number of features and not all token have information that can be used for classification. To have more efficient results it is useful to reduce the

number of inputs for a given model. The more classical approaches for feature selection are Information Gain and χ^2 filter. Other techniques use the entire corpus to identify the best subset of features using Principal Component Analysis and the Latent Semantic Analysis.

The authors of [4] proposed Document Class Distance (DCDistance) a method that outputs a vector representation based on the distance between texts and classes. In [5] a modified version is used, Multi View DCD, by combining DCDistance with a genetic algorithm for feature selection. In both cases the number of features is considerably reduced while also improving the results of the classifiers.

Classification of imbalanced data is a known issue in the classification task. Especially when handling with real world data most data sets will not have a uniform distribution among the labels. For handling this problem the author of [2] used a variation of K-Nearest Neighbor algorithm, proposing the Neighbor-Weighted K-Nearest Neighbor (NWKNN). This algorithm assigns larger weights to class neighbours with lower occurrence and for big classes it assigns smaller weights. For validation the author compared KNN and NWKNN with TF-IDF for feature representation in Reuter-21578[4] and TDT2[5] datasets and was able to show that NWKNN outperformed KNN.

Some deep learning models have been used with success in the area of text classification. Using neural networks provided classifications with alternative tools for the task. By using modified versions of recurrent neural networks, Long Short Term Memory (LSTM), its Birecursive LSTM(BLSTM) variant and Gated Recurrent Unit (GRU), the researchers [6] stated this models perform much better than classical recursive networks. A LSTM

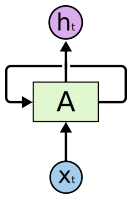

FIGURE 2.2. Recurrent Neural Network

features a mechanism to remember previous node of the network, a LSTM recursively call the same cell every time, only updating the state of its internal structure. The most important features in a LSTM are the cell state, the forget gate (decides what information should be removed), and the input gate (decides what should be forwarded to the next cell). A BLSTM features two LSTM, the researchers state that the experiments with the use of two LSTM layers provided better results in some cases. Finally the GRU is a modified version of a LSTM where the forget and input gate are merged in a single action.

In the Table 2.1 the comparison between different models of RNN in a classification task.

[4]https://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection
[5]https://catalog.ldc.upenn.edu/LDC2001T57

TABLE 2.1. Accuracy, in percentage, on document classification tasks using RNN

Model	Spam Collection dataset	Farm Ads dataset
LSTM	99.798	94.497
BLSTM	99.834	**99.834**
GRU	**99.945**	87.521
AdaBoosting	98.98	84.03
FNN	98.58	94.34

In [7] the authors explore *"Label-Embedding Attentive Model"* (LEAM) by proposing a word embedding approach in which both words and labels are joined in the same latent space in order to measure the compatibility of word-label used as document representations.

TABLE 2.2. Accuracy, in percentage, on document classification tasks

Model	AGNews	Yahoo	DBPedia	Yelp Polarity	Yelp Full
Deep CNN	73.43	98.71	91.27	**95.72**	**64.26**
fastText	72.30	98.60	92.50	95.70	63.90
LSTM	70.84	98.55	86.06	94.74	58.17
LEAM	**77.42**	**99.02**	92.45	95.31	64.09

While in some cases the LEAM model was unable to outperform the other models, the authors stated the algorithm is much less demanding in comparison to other state-of-the-art algorithms.

Some classification algorithms make use of sparse representation techniques in the data analysis and image processing. Although research in this matter regarding text classification remains largely unexplored, the authors of [8] experimented with Sparse Methods for text classification. Sparse classification relays on a collection of dictionaries $D = D_1, D_2, ..., D_c$ with c representing the number of classes and with each D_j representing a class dictionary. The reasoning supporting sparse representation based classifier is that a vector belonging to a class i will fall in the subspace spanned by the corresponding dictionary D_i. They also refer the model "Sparse Representation Classifier based on Minimum Reconstruction Error and class-wise Representation(SRC-4)" works well with imbalanced data

TABLE 2.3. Performance Comparison of SRC-4 with k-NN, Naive Bayes classifier, and SVM (the given values are Macro F1-score of classification)

Dataset	SRC-4	SVM	k-NN	Naive Bayes
Reuters Data	0.8879	0.7867	0.8508	0.8663
web KB	0.8358	0.7545	0.8668	0.8590
20 news group	0.8063	0.73	0.7961	0.7959

For the experiments, they proposed four different approaches for sparse representation based classifiers and tested against the more well known methods SVM, k-NN and Naive Bayes outperforming some of those model 2.3.

To tackle imbalanced complaint data the researchers of [9] proposed the usage *"BERT model to determine shallow labels and use the word2vec model to derive deep labels, thus taking full advantage of the hierarchical characteristics of customer complaint labels"*. The framework proposed consists of a four stage processed.

(1) Data Preprocessing with text enhancement
(2) BERT-based text classification
(3) Word2vec-based semantic similarity matching
(4) Label confirmation

The authors refer to text enhancement with oversampling techniques, for under represented categories the text was translated from the original Chinese to English and back again to Chinese. The resulting text contain new features and doubles the occurrence of the label. *"This model uses text enhancement to mitigate the problem of small category sizes without changing the semantic"*. For their training data set the author refer their BERT model with text enhancements outperformed both BERT without text enhancements and Logistic Regression.

Table 2.4. Results from [9]

Model	Overall accuracy	Average accuracy	Recall	F1 score
Text-enhanced BERT model	0.9175	0.9279	0.9175	0.9168
Unenhanced BERT model	0.9073	0.8672	0.8642	0.8548
TF-IDF-LR model	0.8960	0.9011	0.8960	0.8899

For building artificial data points a wordnet provides the some synonyms and alternatives. The researchers in [10] augmented an existing Portuguese wordnet *PULO (Portuguese Unified Lexical Onthology)*, using monolingual dictionaries and translation dictionaries. The authors successfully increased the size of the wordnet and proved the methods used could be replicated in other wordnets. The word databases are useful for extracting synonyms to create more artificial entries in the data set.

2.1.4. Text Mining in Portuguese

Portuguese orthography has some properties that make the non-language specific techniques under perform when compared to the English language. For example stemming the words *soldado* (soldier) and *soldador* (welder) return the same representation "*sold*". While lemmatization could provide better results, the algorithm needs to finely tuned for each language it is used, with portuguese being a difficult language to lemmatize.

A similar problem was tackled by the researchers in [11] related to complaint data from ASAE. Each complaint is classified in three different dimensions: the type of economical activity, the infractions (each complaint can have one or more infractions), and the competent entity to address the complaint. Since each complaint could feature more than

one type of infractions the authors decided to only select the highest infraction present on the complaint, thus simplifying the problem from a multi label classification problem, while also tackling the imbalanced data distribution. As stated by the authors, they acknowledge the problem but their preliminary experiments did not succeed when using over/under sampling. Their initial experiments used Stanza [12] for both tokenization and lemmatization, since it features state-of-the-art neural models pretrained for Portuguese language and used TF-IDF for feature representation. Then, the following classifiers were used:

- SVM with a linear kernel;
- SGD (SVM with a linear kernel and stochastic gradient descent learning);
- Random Forest.

For analysis and comparison of the results they resorted to accuracy, F1-measure, and, in some cases Receiver Operating Characteristic (ROC) curves. ROC curves can be used to assess *how the true positive rate and false-positive rate of a given class vary by manipulating the decision threshold of the classifier* [11], how can changes made in the decision algorithm that affect recall (positive predictions made out of all positive predictions) impact false-positive rates.

TABLE 2.5. Results obtained using the base techniques for competence prediction from [11]

Classifier	Acc	Macro-F1
Random (stratified)	0.5355	0.5076
SVM	**0.7885**	**0.7748**
SGD	0.7629	0.7339
Random Forest	0.7588	0.7222

After the initial experiments, the data was subject to feature extraction methods such as spell checking, synonym substitution, removal of accentuation, numerical data removal, and stemming. Using these techniques the number of features was significantly reduced. The findings of the experiments are described below, the highlighted methods were kept.

- **Spell Checking**, only replace if the Levenshtein distance is lower or equal to 3. Although it decrease the accuracy, it greatly reduced the number of features.
- Synonym Substitution, low feature reduction and reduced accuracy and F1-measure.
- Removal of Accentuation, low feature reduction.
- **Numerical Data Removal**, high feature reduction.
- **Stemming**, high feature reduction although further experiments could improve the result.

For this experiment they only used SVM, the results can be seen in the Table 2.6.

The author of [13] was also faced with the problem of text classification of public administration data, in his case emails received by a public entity. In this work, the

TABLE 2.6. Results obtained with further feature engineering for competence classification, using SVM. (BT = Base techniques; SC = Spell checking; SS = Synonym substitution; RA = Removal of accentuation; ND = Numerical data removal; St = Stemming) [11]

Experiment	Acc	Macro-F1
BT	0.7885	0.7748
BT + SC	0.7798	0.7656
BT + SC + SS	0.7731	0.7583
BT + SC + RA	0.7796	0.7655
BT + SC + ND	0.7784	0.7643
BT + SC + RD + St	0.7739	0.7597

objective was the classification of emails that can be regarded as a form of user-generated content. Similarly to the previous work, data was also imbalanced. While the work largely focused on data exploration and cleaning the data set was used in a large set of classifiers. To address the issue of imbalanced data the author used several techniques, from which we highlight: deleting classes with an extremely low percentage of data, merging classes with an extremely low percentage of data. The techniques were used together with the goal of increasing classification performance.

Another relevant work was done in [14] relating to text classification of correspondence of the Portuguese navy. Similar to previous works using real world data, the data was also imbalanced. Due to the nature of the data the author was able to experiment with multi stage classification. After discarding labels with less 10 occurrences the first level of classification had 14 labels, the same as the expected number of labels for IGSJ. For the preprocessing of the data and feature reduction the author removed words with frequency less than 4, used NLTK to remove stop words, and also removed Numerical Data , that similarly to [11] also significantly reduced the number of features.

For feature representation the author used TF-IDF. For comparison of the performance of the models the metrics used were accuracy, precision, recall, F1-score, weighted Precision, weighted Recall, and weighted F1-score. Of the models used for classification both Linear Support Vector Classification and BERT achieved the same level of accuracy and F1-score, 0.92 and 0.85 respectively.

Sentiment analysis can be regarded as form of classification. In [15], the authors present an analysis of user commentaries from a Portuguese telecommunication company for a sentiment analysis task. For preprocessing the data the traditional methods like stemming, removing stop words, and punctuation were used, however in stemming for Portuguese is not very good. For example, *fala* (speak) and *falido* (bankrupt) are both represented by the same stem "*fal*", but they represent different words. Also some words have variations that depend on the orthographic convention used, for example "*desativar*" and "*desactivar*" (deactivate). To tackle the issue, the authors resorted to a custom built set of rules to handle these cases: transforming all the words from the old orthographic convention and not stemming the words in the sentiment lexicon. They also used a

personalized list of domain specific stopwords that were considered unfit for the sentiment analysis task. In the conclusion of their work, the authors noted that using off-the-shelf solutions resulted in poor results. The custom rules for preprocessing were highly effective although time consuming.

In [16], a curated corpus of frequently asked questions in Portuguese was developed. The work focused on frequently asked questions of *Balcão do Empreendedor*[6] (Entrepreneur Counter), each set of questions related to any 3 areas of business. In order to create additional data the authors used Google API to translate back and forth to Portuguese and English to develop new data. As well as using native speakers to create alternative constructions of the questions. For benchmark they used the set of the generated question to match the original ones and a classification based of the area of business.

For the classification task they resort to Linear SVM, a Naïve Bayes and, a Random Forest model. To select the features of the questions, TF-IDF-weighted vectors with up to 200 bag-of-words features were used while only considering words that occurred in 50% of the questions and also included random question from film subtitles to diversify the training set. The best performing model was SVM but concluded that *"performance differences when of assigning the correct source to different kinds of variation were not so clear"*. They also noted that using a fine tuned BERT model could have improved the results at cost of higher train time.

The authors of [17] compared SVM with Universal Language Model Fine-tuning (ULMFiT), for classification of official Brazilian Government data. The concluded that, even though ULMFiT is a state-of-the-art technique for classification, it only corresponded to a small increase in classification accuracy when compared to the SVM model, especially considering the training times, simpler training procedure and easier parameter tuning of SVM. The text classification provided by SVM is still competitive with modern deep learning models, also noting the time needed to train a single ULMFiT model can used to fine tune the hyper parameters of SVM.

Machine learning approaches have been used for text processing with varying degrees of success. With this in mind, some researchers adapted the well known BERT model for Portuguese text. Even noting that *"large pretrained learning models can be valuable assets especially for languages that have few annotated resources but abundant unlabeled data, such as Portuguese"* [18]. BERTimbau [7] is a BERT model trained specifically using Brazilian Portuguese data. The authors compared the performance of BERTimbau against BERT for Named Entity Recognition tasks and Sentence Textual Similarity and concluded that the their model outperformed the existing one.

2.1.5. Summary

Table 2.7 aggregates the articles selected for the literature review.

[6] https://eportugal.gov.pt/inicio/espaco-empresa/balcao-do-empreendedor
[7] https://github.com/neuralmind-ai/portuguese-bert

TABLE 2.7. Summary of the article reviewed in this section

Article	Preprocessing	Classifier
[11]	Tokenization, Stemming, Spell checking, Synonym substitution, Numerical Data Removal, TF-IDF, 1-grams	SVM, SGD, Random Forest, CNN, FastText, **Bert**
[13]	Tokenization, Stemming, Lematization, IG, Word Embeddings	Multinomial Naive Bayes, Decision Tree Classifier, K-NN, Support Vector Classifier , Stochastic Gradient Descent , Random Forest Classifier, Gradient Boosting , AdaBoost , **XGBoost**, HDLTex DNN, HDLTex CNN, HDLTex RNN, Random Model Deep Learning, LSTM, BERT
[14]	Tokenization, TF-IDF, Grid Search CV, Word Embeddings	Multinomial Naive Bayes, K-NN, Logistic Regression, **Linear Support Vector**, Stochastic Gradient Descent, CNN, **BERT**
[2]	*NA*	Neighbor-weighted K-NN
[9]	TF-IDF, word2vec	Text enhancement Bert
[17]	SentencePiece, TF-IDF	Naive Bayes, ULMFiT, SVM
[16]	Weigthed TD-IDF	Random Forest, Naive Bayes, SVM
[8]	Stemming	Enhanced sparse representation classifier
[4]	DCDistance and MVDCD	Logistic Regression, SVM, k-NN
[6]	NA	LSTM, BLSTM and GRU

CHAPTER 3

Data Understanding

Getting to know the data is an important step in data analysis. This chapter features an analysis of the data set used in for this work. The corpus and the distribution of the labels are summarized in the following sections.

The workflow for processing the complaints comprehends several steps, the text analysis followed by the assignment of the entity to which the complaint is being directed.

The complaint data had already been processed by a worker of IGSJ and in this work we focus on the summary that was made available along with some labeling annotations. After the analysis the summarized complaints, IGSJ provided some raw complaints to further extend the analysis and the comparison the results of the different datasets.

3.1. Corpus Description

The data was constituted by complaints from 2020 and 2021, in a total of 4459 complaints spread over 18 different labels, the categories are available in Table 3.1. The different labels are related to the institutions the Portuguese Ministry of Justice oversees, however not all entities were targeted by complaints and some labels refer to complaints that must be reported to other entities, outside of IGSJ competence. The data is characterized by the variables *Subject* (summarized complaints), *Entity* and 14 additional variables with details regarding the pipeline process and complaint outcome.

While processing the complaints the worker will respond to some complaints without being redirect to the complained entity. Either because some of them are not of the responsibility of IGSJ or simply because they refer to a known issue of some entities and thus can be answered without being redirected. For example delays regarding *"Cartão do Cidadão"* (Portuguese citizenship card), *"Nacionalidade"*(Nationality), or *"Transferências de reclusos"* (Inmate transfer) are common complaints.

3.2. Corpus Analysis

Of the 4459 complaint summaries, it was extracted a lexicon of 3705 different tokens, with 1805 tokens only appearing once. An example of the complaints can be found in Table 3.2.

Some of those tokens are an abridgement of words or acronyms other entities, for example *"dto"* - *"direito"* (right hand side) , *"jf"* - *"junta de freguesia"* (Council Office), or *"iva"* - *"imposto valor acrescentado"* (Value Added Tax).

The complaint summaries have an average of 16 words per complaint with the shortest having only 2 words and longest 38 words. The shortest summaries were similar with the smallest (two words) repeated two times containing the text *"Atrasos processuais"*

TABLE 3.1. Label Distribution

Entity	Count
IRN	2747
"*Alheia à comp. IGSJ*" (Outside of IGSJ competence)	761
DGRSP	732
Tribunais	109
DGAJ	22
IGFEJ	21
CAAJ	19
INMLCF	17
PJ	10
CPVC	5
INPI	5
-	3
DGPJ	2
SGMJ	2
CEJ	1
DGPJ/IRN	1
ON	1
Alheia à comp. IGSJ - DN da PJ	1

TABLE 3.2. Data example, as featured in the data set

Complaint	Entity
"Cartão do Cidadão - Exposição apresentada por ter o CC a caducar e não conseguir marcar a renovação." (Citizen's Card - Exhibition presented for having the CC expiring and not being able to schedule the renewal)	IRN
"Falta de competência da IGSJ - Exposição apresentada referente a uma mensagem de burla suspeita." (IGSJ's lack of competence - Exposure presented referring to a suspicious fraud message.)	"*Alheia à comp. IGSJ*" (Outside of IGSJ competence)
"*Outros - Exposição anónima a denunciar guarda prisional reformado que continua a usufruir de casa de função*" (Others - Anonymous exhibition denouncing a retired prison guard who continues to use a function house)	DGRSP

(Process delays). Other small summaries of with three and four words were similar with the added words giving more detail regarding the actual complaint for example "*Atraso processual IRN*" (Nationality process delays).

The text data features a very small number of spelling errors.

Looking at the most frequent words Table 3.3 it is expected that *"exposicao"* and *"apresentada"* and, *"por"* have a large quantity of appearances since most of summaries started with *"exposição apresentada por"* (compliant submited for), almost every document features either one, the other or even both. It is also worth mentioning the abbreviation *"cc"* refering to the Portuguese identity card *"Cartão do Cidadão"*.

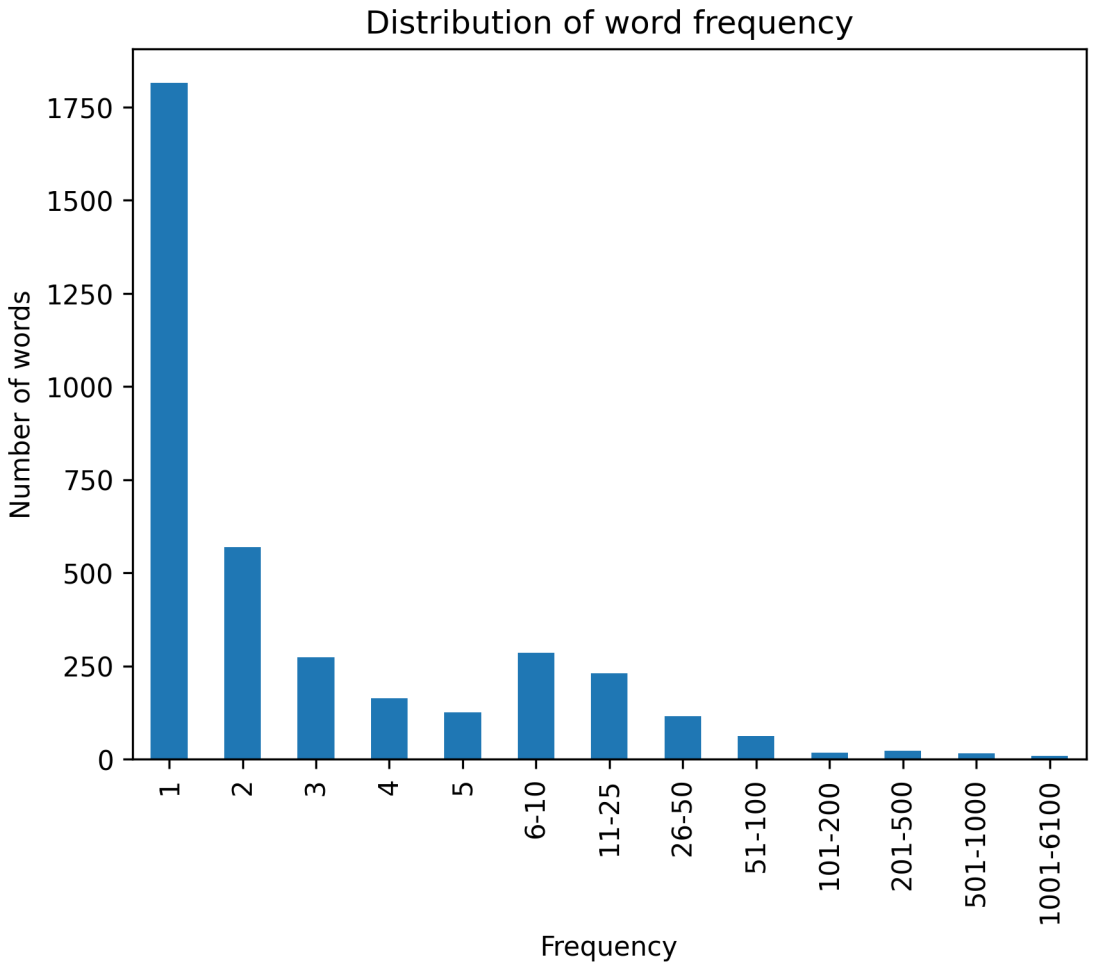

FIGURE 3.1. Word frequency

TABLE 3.3. Most frequent words

Words	Count
cc	876
outros	1327
da	1406
no	1420
referente	1658
do	3408
apresentada	3503
por	3782
exposicao	3886
de	6001

3.3. Label Distribution

As mentioned in Section 1.1 IGSJ oversees 14 other different entities, Table 1.1, but the form allows for 17 different entities having three additional fields in the form "*Estabelecimento prisional*" (Prison Establishment), *"Centro Educativo"* (Educational Center), and "*Secretaria de Tribunal*" (Court Scretary). However the data featured 18 different

labels, Table 3.1 the distribution of the labels in the data set is represented. The labels present in the form the following were missing from the dataset: CPES, IGSJ, *"Centro Educativo"* (Educational Center), and "*Secretaria de Tribunal*" (Court Scretary). Some documents had an new labels: "*Alheia à comp. IGSJ*" (Outside of IGSJ competence), "-", and "*ON*".

It is also important to notice that two entities, IGSJ and CPES, were not assigned any complaint. Thus the models used in this work will not be able to take into account those entities.

CHAPTER 4

Data Preparation

This chapter presents the overall data processing, both text and labels were subject to an initial analysis, the process can be seen in Figure 4.1.

4.1. Text Preprocessing

For the initial preparation of the documents all characters were converted to lowercase and the special characters and accentuation were removed from all words. The numbers were also removed.

Some tokens that were short forms or even acronyms of other words some were replaced.

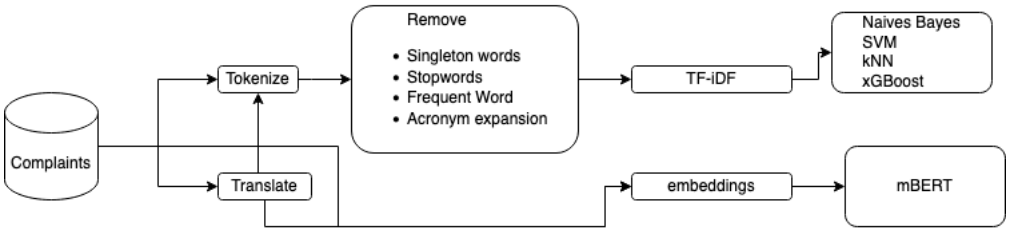

FIGURE 4.1. Text processing workflow

4.2. Corpus Optimization

Referring to Table 3.3, it possible to see that the words *"exposição"* and *"apresentada"* have a high frequency, appearing in almost all of the documents due to this they were candidates to removed from the corpus.

In contrast some words only appear once in the corpus, this words are also undesired for the more traditional models. In total 1815 words with frequency one were removed.

After analyzing some of the complaints some acronyms were expanded to provide extra information for the all the classification models. All of the entities tutored by IGSJ were always represented in such form and were replaced by the full name. Also similarly referred in the previous Section 3.2 some other acronyms were also present that were substituted by their complete words. For example: EP was transformed into *"Estabelecimento Prisional"* (prison establishment). CC was transformed into *"Cartão do Cidadão"* (identity card). MJ was transformed into *"Ministério da Justiça"* (Ministry of Justice).

4.3. Label Imbalance

In regards to the labels available in the dataset, we opted to merge some of categories in order to form bigger sets. Labels with *"-"*, *"Alheia à comp. IGSJ"*, and *"Alheia à*

comp. IGSJ - DN da PJ" were all assigned to the same label, *Unrelated*, since all were referring to complaints out of the jurisdiction of IGSJ. The label with *"ON"* was changed to *"IRN"* since it was directed to that Institute. Lastly, a complaint was attributed two labels, *"DGPJ/IRN"* and it was categorized to *"DGPJ"*, this choice made since this class had considerably fewer complaints than *"IRN"*.

As it is possible to observe in Figure 4.2 almost more than half of the complaints are targeted to *"IRN"*.

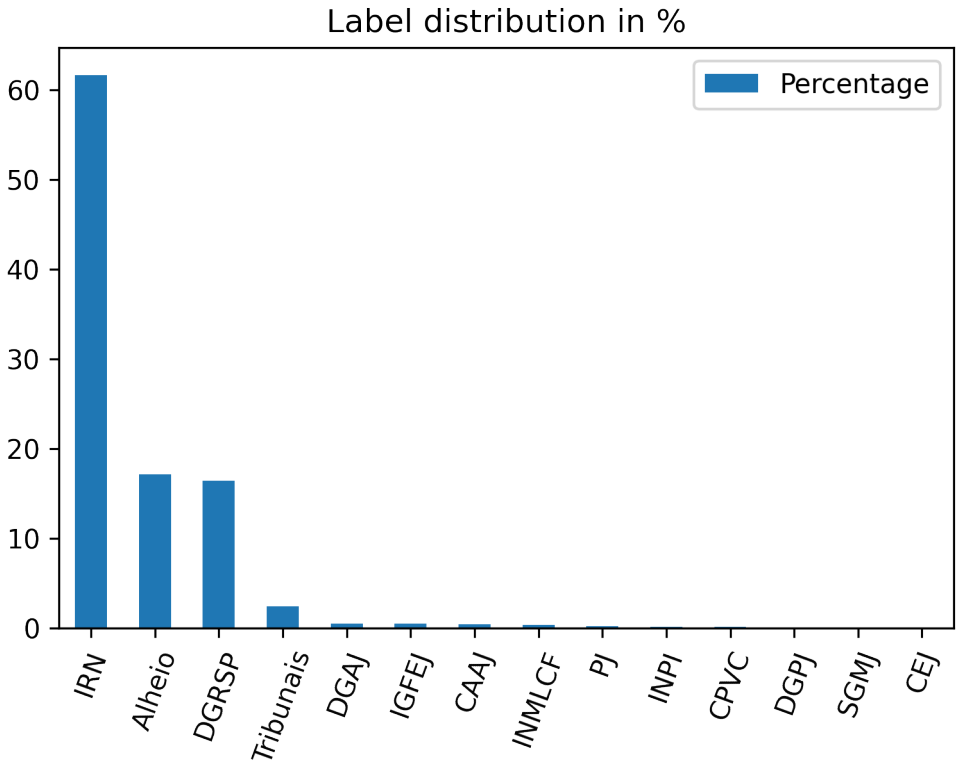

FIGURE 4.2. Label distribution

The data was critically imbalanced, more than 90% of the data was targeted to only three classes.

In order to tackle this issue the same strategy as in [9] was applied. Translating to other languages and back to the original in order to increase the number of examples of the least represented classes, an example of the translation can be found in Figure 4.1.

TABLE 4.1. Translation example

Summary	Translation
Outros - Exposição apresentada por pelo facto do pedido de indemnização requerida pelo seu filho ainda não estar concluído.	Original
utros - Exposição apresentada pelo fato de que o pedido de compensação exigido por seu filho ainda não está concluído.	English

To increase the number of class representatives, the documents of the classes that had between 10 and 30 elements were translated into several languages (English, Spanish,

Italian, Polish, and German) by using a Python package to acess Google translate API[1] and back to Portuguese. This strategy increased the number of representatives by sixfold (Table 4.2). By using this method an additional 303 words were added to the lexicon. These words were provided by the translation technique and are synonyms of some words present in the dataset. It is important to highlight that the fabricated complaints were only used for model training while for validation of the models only real world data was used.

TABLE 4.2. Distribution of complaints per label, the first 3 labels contain almost 80% of the data set. Using the translation technique made the classes more equal in regards to their size

	Class	Class with translations
IRN	2748	2748
Alheio	765	765
DGRSP	732	732
Courts	109	109
DGAJ	22	132
IGFEJ	21	123
CAAJ	19	114
INMLCF	17	102
PJ	10	60
INPI	5	30
CPVC	5	30
DGPJ	3	18
SGMJ	2	12
CEJ	1	6

4.4. Handling Label Imbalance

Another approach to mitigate the observed imbalance was based on [14], having multiple stages of classification. The main idea for this approach was to first classify the top 3 labels, that are approximately 95% of the dataset and on a second classification only classify the classes with the least amount of representatives, a diagram is available on Figure 4.3.

The documents were then regrouped into different labels. Firstly, the complaints were separated into the top 3 classes and the remaining classes. For the final split, the classes ranking from 4 to 8 were given the respective label and labels that had less than 10 example were combined into a single class, named "*Others*". Figure 4.3 illustrates this process. From this point onwards, the first stage classification will refer the classification of the top 3 classes along with the "Others" (new label for the remaining classes) and the second stage classification will refer to classification of the others to the remaining considered classes. The full data classification will refer to the top 8 classes along with the remaining grouped into the class "Others" of the second stage. For all the experiments,

[1]https://pypi.org/project/translate-api/

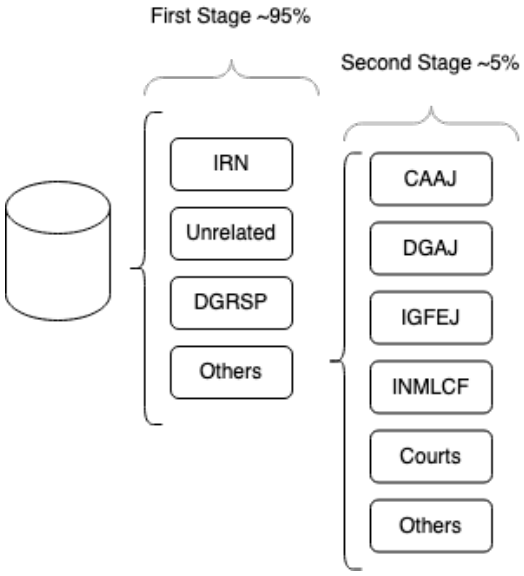

FIGURE 4.3. To handle the label imbalance issue a multistage classification method was analyzed, the top 3 classes were initially classified and the remaining top 5 classes were also considered, class with lower representation were not considered for the task.

the classes with a cardinality lower than 15 were not considered as a full class: they were assigned the label of "Others".

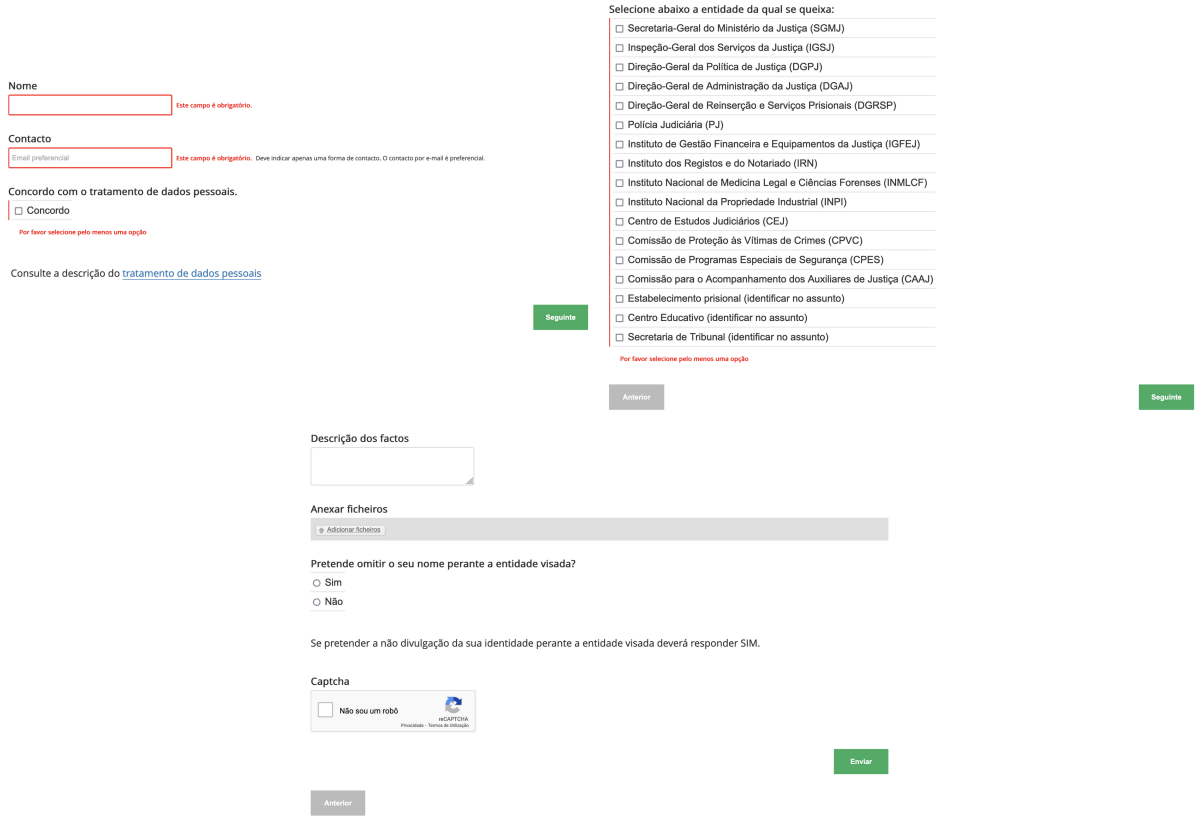

FIGURE 4.4. Complaint input form

4.5. Raw Data

The raw data data, in pdf format, featured complaints from the years 2020 and 2021 and only contained data that was submitted using the website form, and example of the form can be found in Figure 4.4.

The raw complaints contained sensitive information related the author that were previously anonimized, by replacing all sensitive information with non-retraceable placeholders. Data was then converted into CSV format so that it could processed. In the end the csv contained the ID of the complaint, the targeted institution and the complaint. Unfortunately this data did not feature any identifier to match with the main data source used in this work.

This collection featured a total of 142 complaints from 2020 and 2021 and was only used to test and validate the models produced from the curated complaints.

4.6. Data Description

The complaint input form, Figure 4.4, provided several fields for additional detail regarding the complaint and the complainee. The name and contact of the complainee(anonimized and not used), the complaint target and the text.

For the entity, it was possible to select more than one and in some cases it was possible to write additional text regarding the entity, for example to specify a prison establishment or a court. Since the entity was picked by the complainee and even several

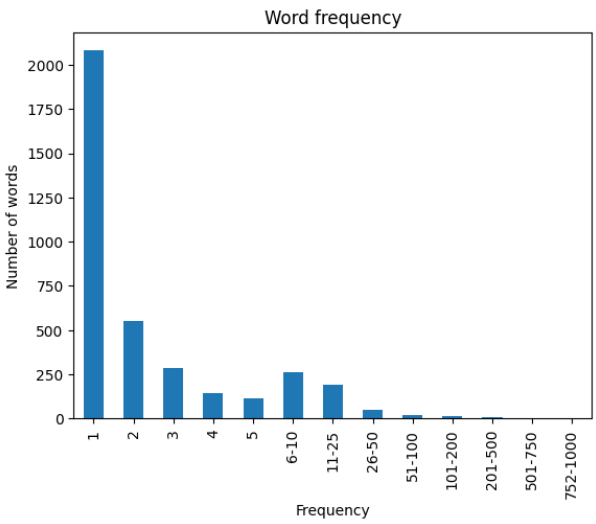

FIGURE 4.5. Raw data word frequency

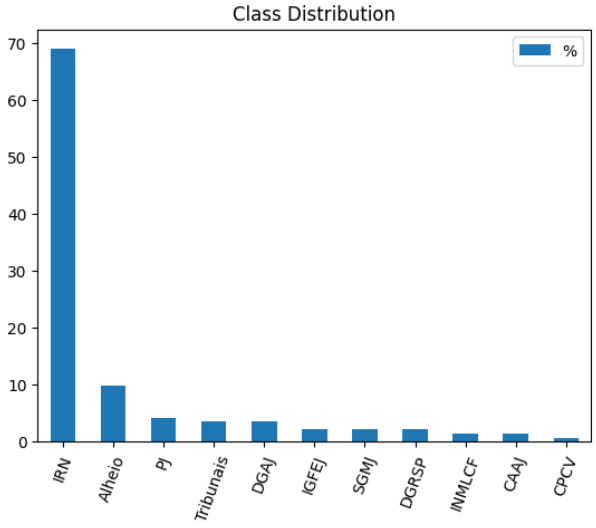

FIGURE 4.6. Raw data label distribution

could be picked, instead of being validated by a worker from the institute. The entity could not be as accurate as the curated and summarized complaints.

The raw complaints were then further transformed from the csv into a dataset that featured the following attributes *Text* and *Entity*. The entity label also needed to be manually processed in order to match the labels the models were trained on.

4.7. Raw Data Analysis

The raw data set also presented the same imbalance issue with "*IRN*" having most of the examples with 98 of the 142 related to the same label. Comparing Figure 4.6 with Figure 4.2 shows the same imbalance for the most common label, although unlike the summarized documents dataset only the label *IRN* is more prevalent.

Comparing also the corpus of the raw dataset, it featured more words than the summarized one, 3724 different words, with over 2000 only appearing once. Of those around 1263 words appeared in both corpus. In Figure 4.5 the word frequency of the corpus is presented.

CHAPTER 5

Classification

Having a deeper insight of the data and the respective preparation sorted the ensuing phase is the classification. In this chapter, we focus on the classification experiments, performed with both the original dataset and the augmented dataset with the translated complaints. In order to better assess our proposal an initial baseline is created. Additionally we will experiment with classic machine learning models. Afterwards the classification task is done with pre-processed data set and using classic models and deep learning algorithms.

5.1. Initial Experiments

The first experiments were initially executed with SVM and Naïves Bayes, without extra processing done in the data and for the full dataset. After analyzing the output, most of the accuracy was attributed to the classes with greater representation in the dataset. After these initial experiments, and based on the insights obtained from the outcome of the experimetns, another experiment-cycle was planned (as is common in the CRISP-DM methodology). These results motivated the usage of some of the techniques explained in the previous chapters to enhance the classification capabilities of the trained models.

5.2. Baseline

For creating a baseline the unprocessed text data was used. The stopwords, numbers and all the words were tokenized and TD-IDF was used as document representation.

Initially experiments were carried out without model fine-tuning and no data balancing using artificial texts.

Due to the aforementioned label imbalance some results were subpar, having some labels withount any predicted labels, as seen in Figure 5.1 and in Table 5.1. In the referenced table and figure it is possible to understand most of the performance is attributed to the most frequent labels, *"Alheio"*, *"IRN"* and *"DGRSP"*, with all the performance metrics having a score greater than 70%.

This was the main motivation for the two step classification approach. Following the CRISP-DM methodology, after this iteration some of the described preprocessing in Chapter 4 was made. The experiments were carried over using Naïve Bayes Classifier, k-NN, SVM, Gradient Bosting, and BERT.

	Precision	Recall	F-Score	Support
Alheio	0.81	0.70	0.75	247
CAAJ	0.08	0.83	0.14	6
DGAJ	0.03	0.14	0.05	7
DGRSP	0.95	0.91	0.93	228
IGFEJ	0.11	0.50	0.19	8
INMLCF	0.11	0.67	0.19	6
IRN	0.98	0.74	0.84	802
Others	0.11	0.43	0.18	7
Tribunais	0.15	0.59	0.24	27

TABLE 5.1. Metrics by class for Naïve Bayes for classification of all the labels, the baseline used for evaluating the perfomance of the techniques and other models

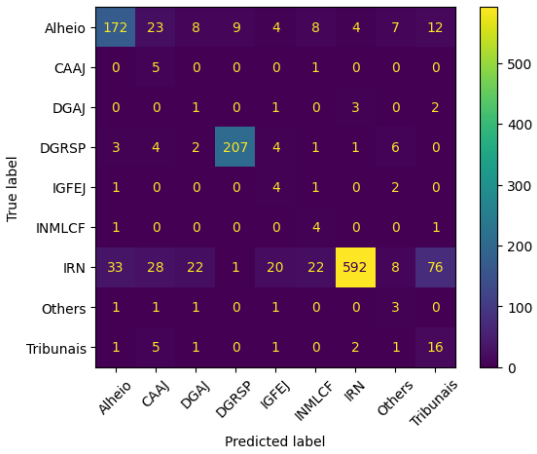

FIGURE 5.1. Confusion matrix Metrics by class for Naïve Bayes for classification of all labels

5.3. Machine Learning Models

The classic models used for classification are based on a sparse data representation, using TF-IDF weighting.

Following TF-IDF representation, each complaint was transformed into a vector of numbers, with each entry representing the score of the word for the document. Since the models are not capable of processing words they needed this step to transform into number to used as input for the models to evaluate.

Naïves Bayes was used as the baseline for all the experiments performed in this dissertation. The model yielded better results for most tests, except for the second stage classification. Regarding the first classification the experiments ended with accuracy of 89%, in Table A.2 the second stage classification with 31% accuracy, and in Table 5.4 with 75% accuracy, for further detail in classifying all the labels in Figure 5.1 and in Table 5.1. While only comparing the first stage classification Naïve Bayes model present good results, in some cases even better than BERT based model.

For k-NN when compararing the full data set classification this model is on par with the best performing ones. When using a multistage classification k-NN was outperformed by other traditional models.

SVM was one of the best performing models for this task and after augmenting the training set it showed an improvement for all tasks but specially in the second stage classification. This can be seen in the confusion matrices in Figure 5.2 and in Figure 5.3 and in Table A.2, the original data set results, and in Table A.5 for the augmented dataset.

Using Extreme Gradient Boosting (XGBoost) for classification lead to the worst results, specially when dealing with the low data of the second stage classification.

Finally we also experimented different representation of the complaints: word embeddings. The complaints were transformed using BERT multilingual model pretrained

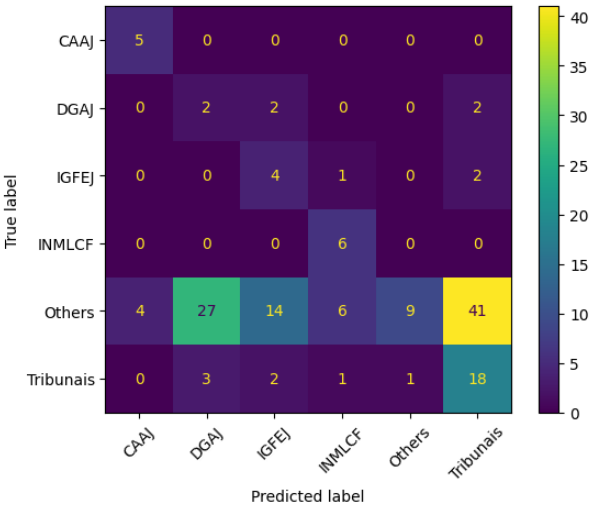

FIGURE 5.2. Confusion matrix for the second stage classification using the original dataset

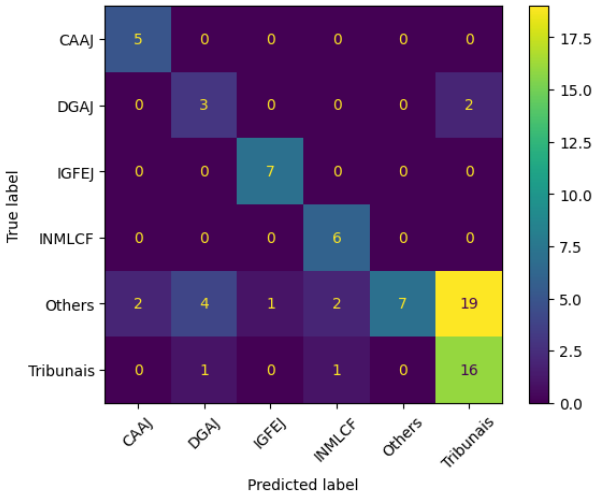

FIGURE 5.3. Confusion matrix for the second stage classification using the augmented data set

embbedings. Figures from Figure 5.4 to Figure 5.9 represent the accuracy gained during the training epochs. Since the training set is relatively small in the first epochs the accuracy gain is larger and stabilizes in the later ones. The figures represent the training loss and accuracy plotted for each trained epoch.

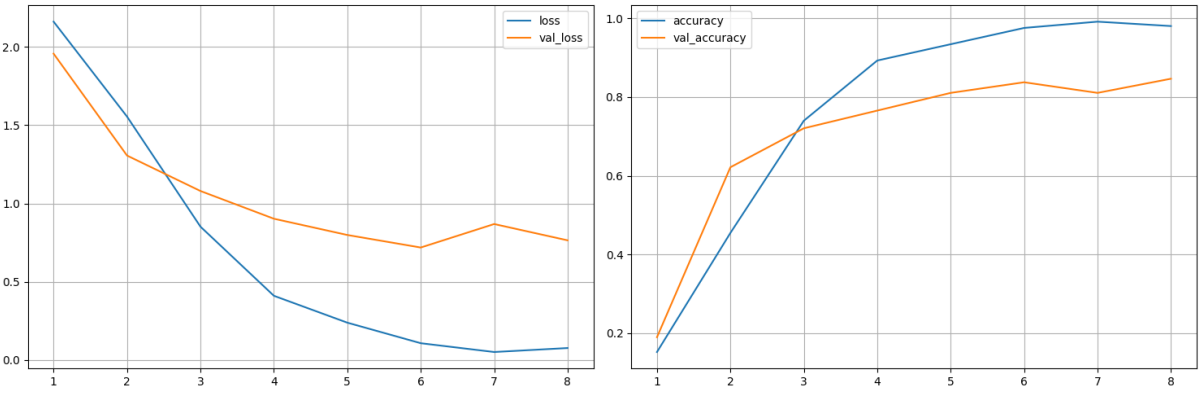

FIGURE 5.4. Mean loss and mean accuracy results from training set using the original dataset and for classifying all labels

Comparing the results for classifying all labels, in Figure 5.4, Figure 5.5, and Figure 5.6, using BERT the translated dataset presented an improvement over the original dataset, leading to a better end accuracy. It is also worth noticing that when using the expanded dataset the training lasted more epochs than when using the original.

Similar to the full classification, for the first stage the increase the accuracy is also noticeable.

Due to the lack of training examples, the model was unable to be trained using the original dataset. The only analysis to be done is over the extra processing used. Both

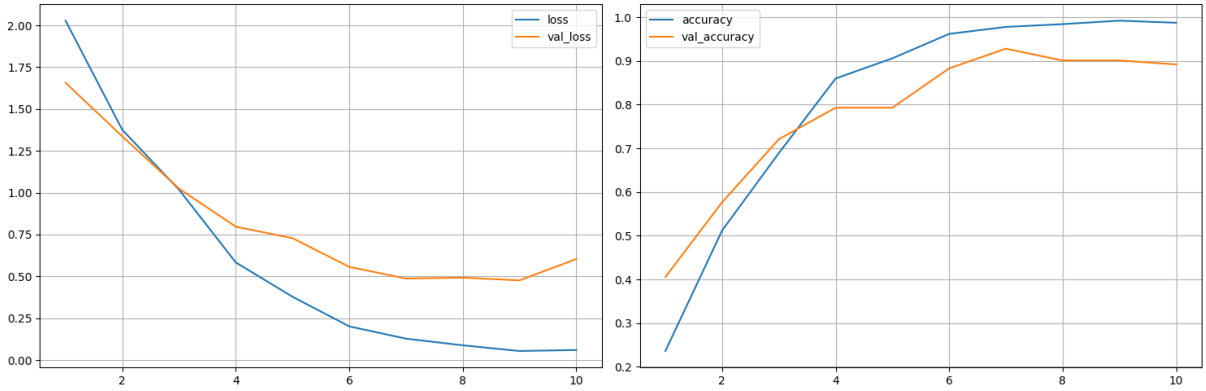

FIGURE 5.5. Mean loss and mean accuracy results from training set using the translated dataset with extra processing for classifying all labels

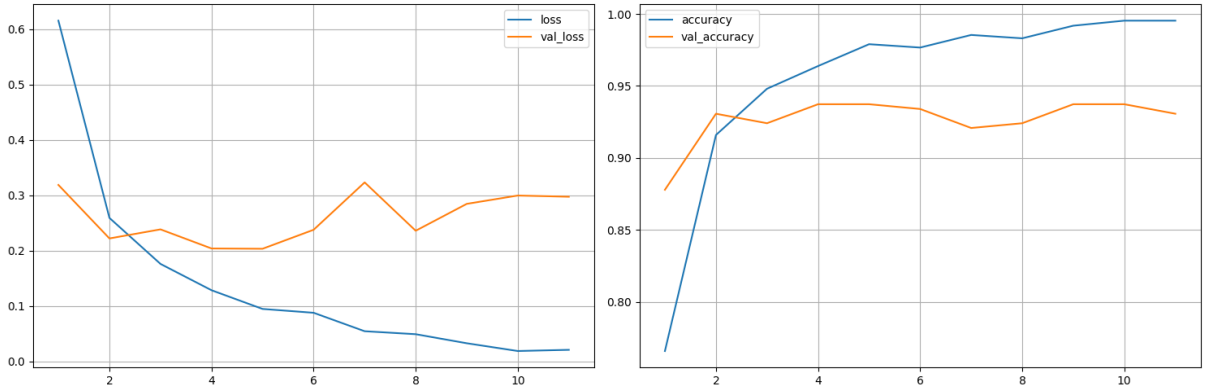

FIGURE 5.6. Mean loss and mean accuracy results from training set using the translated dataset for first stage classification

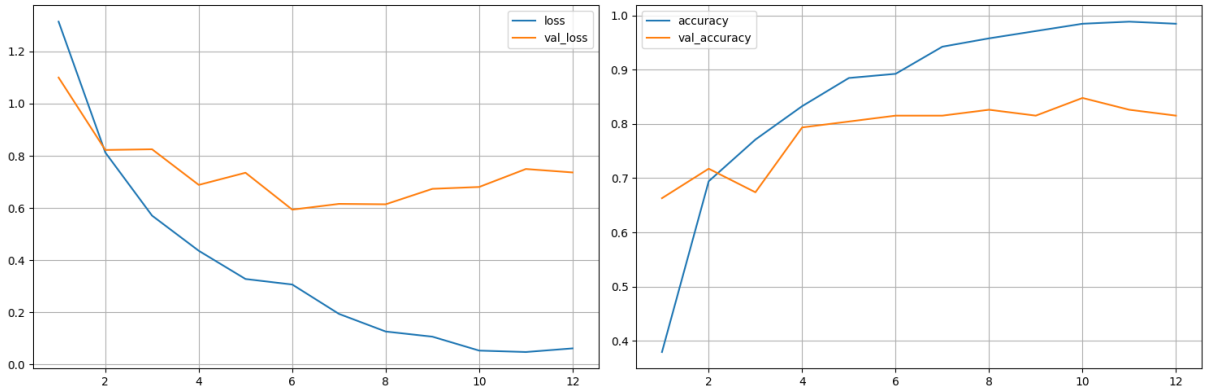

FIGURE 5.7. Mean loss and mean accuracy results from training set using the original dataset first stage classification

experients yielded a similar accuracy, with the unprocessed data set being slightly more accurate.

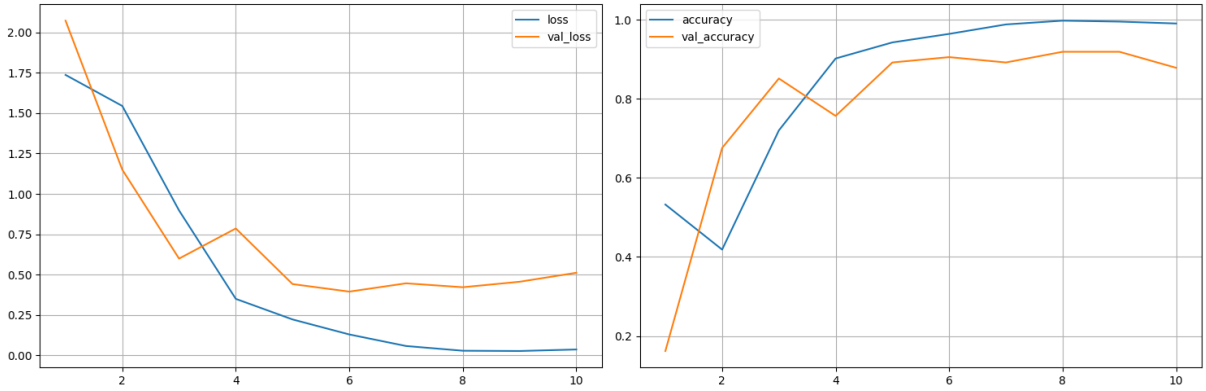

FIGURE 5.8. Mean loss and mean accuracy results from training set using the translated dataset with extra processing for second stage classification

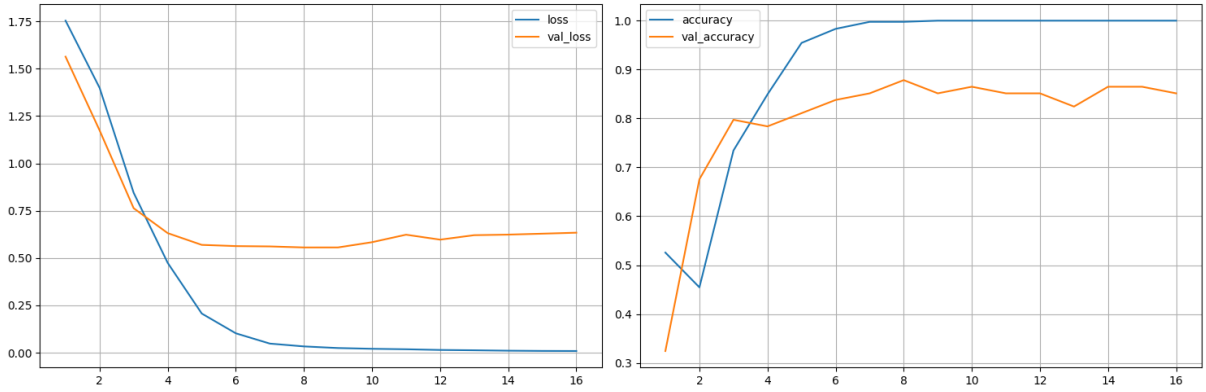

FIGURE 5.9. Mean loss and mean accuracy results from training set using the translated dataset for second stage classification

Table 5.2. Results for the SVM classifier using the summarized dataset

Model	Processing	Accuracy	Precision	Recall	F-score
First stage	No	0.89	0.92	0.89	0.91
	Yes	0.89	0.93	0.89	0.90
Second Stage	No	0.31	0.67	0.31	0.24
	Yes	0.29	0.70	0.29	0.24
All labels	No	0.78	0.91	0.78	0.83
	Yes	0.79	0.92	0.79	0.84

5.4. Summarized Complaints Results

All tables with the results for the experiments are presented in the Appendix, Appendix A for the summarized dataset and Appendix B for the summarized dataset.

The results for the classification and using the original dataset can be viewed in Tables A.1, A.2, and A.3 (Appendix A).

Considering the first stage classification for the original dataset, the BERT based models and the more classic machine learning models yielded similar results to SVM, proving to be the model with best performance in this experiment.

For the second stage classification, the results were the worst from all the experiments. While presenting around 70% precision, the accuracy, and f-score values were extremely low. Due to the low number of class representatives, the BERT model was unable to be fine tuned. The training and validation data would be very short for the complexity of a neural network.

For the full classification when using the original dataset, k-NN and SVM presented good results with an 80% accuracy and 92% precision, respectively. When comparing the results for each class in Table 5.3 it becomes apparent the good results are heavily weighted by the distribution of the testing set. The classes with more representatives have more weight and yield better performance.

Table 5.3. Metrics by class for SVM using the original dataset

	Precision	Recall	F-Score	Support
Alheio	0.88	0.64	0.74	247
CAAJ	0.14	0.83	0.24	6
DGAJ	0.05	0.43	0.10	7
DGRSP	0.97	0.90	0.94	228
IGFEJ	0.11	0.75	0.19	8
INMLCF	0.13	0.83	0.22	6
IRN	0.98	0.82	0.89	802
Others	0.11	0.57	0.18	7
Courts	0.27	0.52	0.35	27

The deep learning models had an accuracy value of 58% when compared to k-NN with 80% accuracy, Table 5.4.

TABLE 5.4. Results for the full classification using the original dataset, k-NN and SVM have similar scores and present decent performance although SVM has a higher precision and f-score

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.75	0.90	0.75	0.81
	Yes	0.77	0.92	0.77	0.83
SVM	No	0.78	0.91	0.78	0.83
	Yes	0.79	**0.92**	0.79	**0.84**
KNN	No	0.79	0.88	0.79	0.833793
	Yes	**0.80**	0.88	**0.80**	0.83
XGBoost	No	0.60	0.84	0.60	0.68
	Yes	0.63	0.89	0.63	0.73
Multilingual-Bert	No	0.58	0.82	0.58	0.67
	Yes	0.55	0.84	0.55	0.65

Considering only the first stage classification task and not removing frequent words, the results were much better results with 89% accuracy using SVM, although only for a limited number of classes. The full classification, while also presenting good results, was skewed from the imbalance of the testing set, most of the accuracy was attributed to three labels. The performance results for SVM can be found in Table 5.2, this model was overall the best performing for the original dataset, both the first stage and the full classification.

The results for the classification using the expanded dataset can be viewed in Tables A.4, A.5, and A.6 (Appendix A). The expanded dataset yielded considerably better results for all the tasks examined in this work.

The first stage classification had an increase of almost 6% from the original data set with SVM outperforming the others, this is more noticeable on Figure 5.10 and Figure 5.11. Surprisingly with the second stage classification almost doubling the performance achieved with the original dataset. Accuracy improved from 29% to 58% and f-score increased from 24% to 52%, when using an SVM. With the augmented dataset it was possible to fine-tune BERT for the second classification and it proved to be the best performing model for this task.

For the full classification using SVM, the accuracy reported was of 89% with 90% f-score, an increase of 10p.p. from the dataset. Using BERT for this task returned similar results to the SVM noting the approximated 20p.p. increase in accuracy from the original dataset, from 58% to 76%. Having a bigger training pool was essential in improving the BERT model.

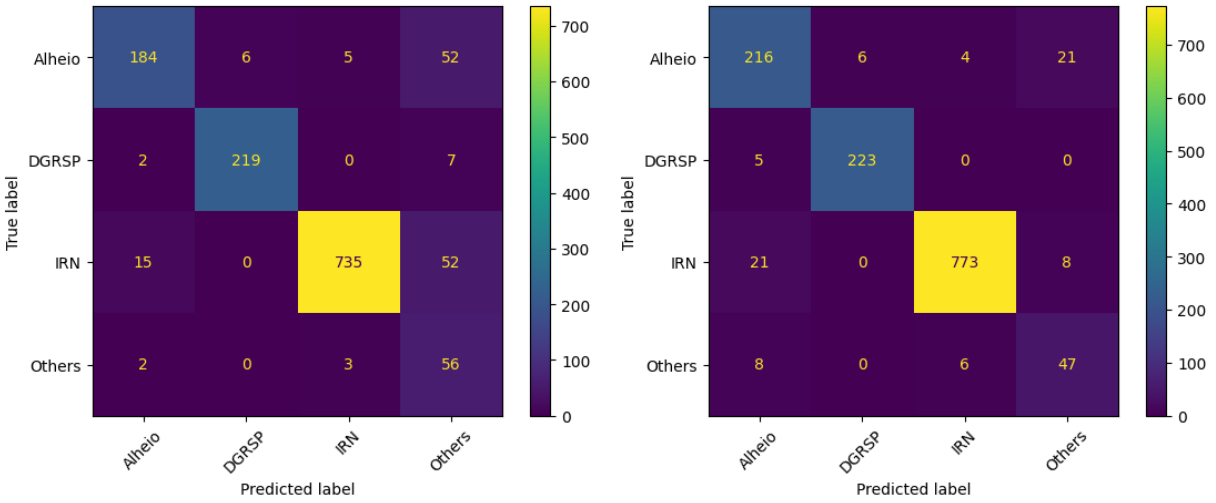

FIGURE 5.10. Confusion matrix for first stage SVM, on the right the model was trained using the expanded dataset

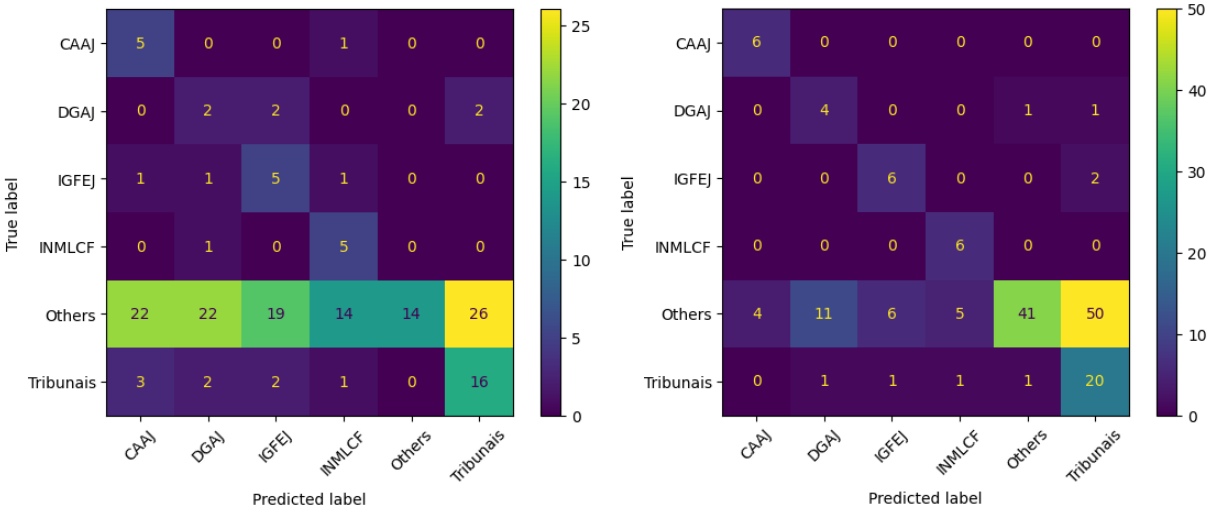

FIGURE 5.11. Confusion matrix for the second stage using SVM, on the right the model was trained using the expanded dataset

5.5. Raw Data Results

The raw data was used as an input for the models trained in Chapter 5 with the summarized data. In order to use the raw as an input, the same processing used for the summarized complaints in Chapter 4 was also repeated for the dataset. Further the same tests(multistage classification and removing stopwords) were experimented for this data set, the multi stage classification and extra processing of the complaint text. The results are available in the following tables. The tables Table B.1, Table B.2 and Table B.6 (Appendix B) were compiled using the original summarized dataset.

For the first stage classification the predictions were less accurate when compared to initial experiments. This can be largely explained due to the imbalance of the raw dataset. Nonetheless most models had an accuracy over 50% with most of the accuracy being explained by the correct predictions of the most common label IRN.

For the second stage the performance drop was in line with the original dataset. Noticeably XGBoost yieled terrible results it was unable to predict any label, check the confusion matrix in Figure 5.12.

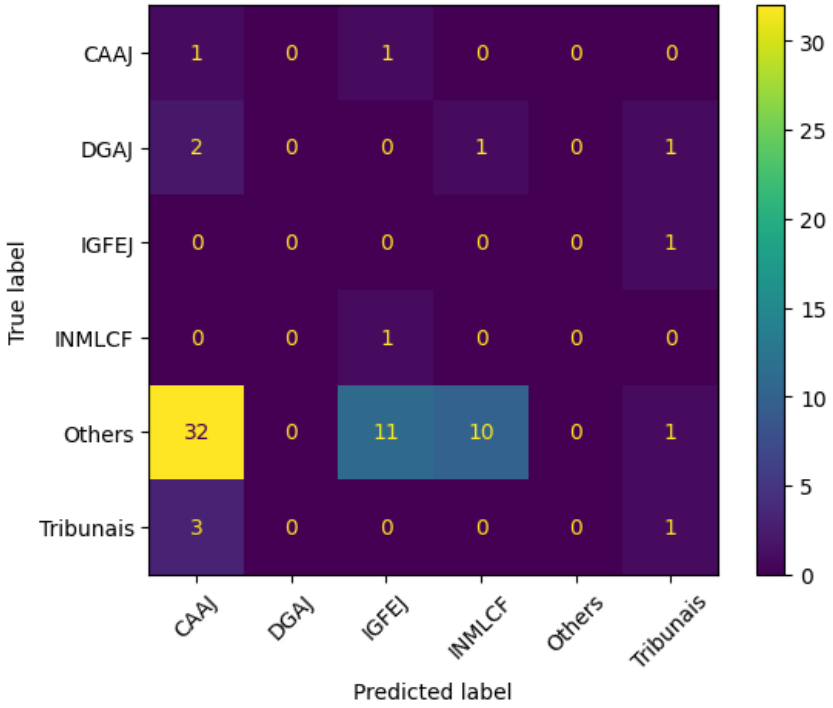

FIGURE 5.12. XGBoosting for second stage classification

Regarding the full classification over the raw dataset it is noticeable the decrease in performance when compared to the results obtained with the summarized dataset, Table 5.4. Most of the accuracy is related to the correct prediction of the most common label, Table 5.5 holds the performance of each class.

Comparing the models trained with the augmented dataset and the base dataset a drop in performance for BERT is noticeable. The model failed to classify all of the most common label, the labels that it correctly guessed were unable to account for the incorrect classifications.

Similar to the results for the second stage classification (Table B.3), using BERT also resulted in terrible performance, for the less common labels the training and the raw examples were very diverse especially when comparing the words used and length of the complaint as well as the lack of classification of the most common label IRN.

When comparing the results of the full classification with BERT with the raw dataset the latter presented higher scores. Comparing the full classification with the first stage classification, especially for the results that are similar, most of the performance can be attributed to the most common label, this can easilly be observed in Figure 5.13.

Table 5.6 presents the detailed results for the XGBoosting model trained with the augmented data over the raw data. As previously mentioned most of the accuracy is

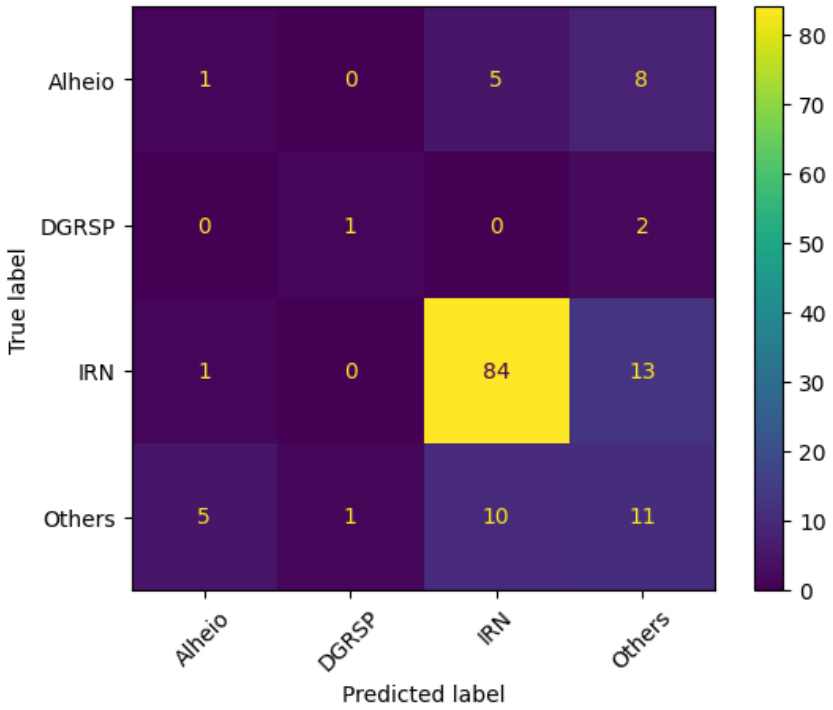

FIGURE 5.13. Confusion matrix for BERT using the Raw Data set, almost all of the classifications are attributed to IRN

TABLE 5.5. Metrics by class using BERT for classification of all the labels,most of the performance can give attributed to the most common label

	Precision	Recall	F-Score	Support
Alheio	0.50	0.14	0.22	14
CAAJ	0.00	0.00	0.00	2
DGAJ	0.09	0.40	0.15	5
DGRSP	0.17	0.33	0.22	3
IGFEJ	0.40	0.67	0.50	3
INMLCF	0.00	0.00	0.00	2
IRN	0.86	0.77	0.81	98
Others	0.50	0.20	0.29	10
Tribunais	0.27	0.60	0.37	5

attributed to the most common label as well other labels that were correctly guessed, in comparison to the other models.

Table 5.6. Metrics by class for XGBoosting using the raw dataset trained with the augmented dataset

	Precision	Recall	F-Score	Support
Alheio	0.33	0.14	0.20	14
CAAJ	1.00	0.50	0.67	2
DGAJ	0.00	0.00	0.00	5
DGRSP	0.50	0.67	0.57	3
IGFEJ	0.14	0.33	0.20	3
INMLCF	0.00	0.00	0.00	2
IRN	0.94	0.78	0.85	98
Others	0.32	0.60	0.41	10
Tribunais	0.20	0.60	0.30	5

CHAPTER 6

Conclusion

The main objective of this dissertation was to classify the summarized complaints of the IGSJ. The first analysis of the data showed a highly imbalanced target classes, Figure 4.2. As such, this work also describes the techniques explored to overcome this issue. The main approaches explored were a multistage classification and augmenting the dataset using artificially made complaints created via translation to other languages. A first attempt of pre-summarization (raw data) classification was attempted to determine the gain of applying classification before or after the summarization step.

The developed classifiers were able to correctly classify some of the summarized complaints. Augmenting the data proved essential to train models(especially BERT) while also improving the outcome. Similar to the results achieved by Silva et al. [19], the concise versions of the complaints proved to be enough for a satisfatory and usable classification results. Removing stopwords and frequent words had low gains: in this case, since the tokens were already part of a short corpus, the additional processing (removing stopwords, the most highest and low frequency words) showed only marginal improvements.

Due to the low number of complaints for some classes, the more traditional models had a better performance than the BERT based models, especially when using the original dataset. When using the expanded dataset, deep learning models yielded similar results to more traditional models. For the second stage classification, BERT was able to outperform all of the others. The expanded dataset led to considerably better results when compared to the original data set, especially for the second stage classification task and for the full classification. Boosting the number of representatives for each class, especially the least represented ones, greatly improved the performance of the models. The new sentences and the tokens introduced were essential to improve the BERT performance. Even though further testing is needed to confirm this, these experiments seem to indicate that balancing the data played a crucial role in the performance gains and could be considered as technique for improving datasets with lower cardinality.

When using the raw dataset for classification it is easily noticeable that the performance is far worse, however for some cases (the full classification and first stage classification) using BERT yielded a score capable of correctly identifying the most common label. Using these models for the raw data could prove essential for picking the most common label and considerably reduce the amount of time a worker spends identifying complaints targets.

6.1. Future Work

For future work the recommended path is the exploration and fine tuning of a multi-stage classification methods, comparing more classification stages and using binary classification [20]. More techniques for expanding the corpus and classes examples could also be explored, as an example wordnets could be used to further diversify the dataset. Creating hand made examples for the lower classes could provide even better representatives for each class as the machine made translations that can only reach a certain limit. It should also be noted that combining different models for the various stages of the classification could provide additional insights.

On the topic of the complaint classification for IGSJ, some other tasks are still open for analysis. For example in this work the data had already been summarized, the process to automatically generate a summary is open to work and could provide new insights into the data. Another interesting work based on the complaints from IGSJ would be to predict the flow of some of the complaints. As mentioned in Chapter 4, some of the complaints were recurrent complaints and triggered a premade response, exploring this classification also poses an open issue for this problem.

References

[1] R. Wirth and J. Hipp, "Crisp-dm: Towards a standard process model for data mining," *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, Jan. 2000.

[2] S. Tan, "Neighbor-weighted k-nearest neighbor for unbalanced text corpus," *Expert Systems with Applications*, vol. 28, no. 4, pp. 667–671, 2005, ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2004.12.023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417404001708.

[3] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. arXiv: 1706.03762. [Online]. Available: http://arxiv.org/abs/1706.03762.

[4] C. H. P. Ferreira, F. O. De Franca, and D. R. Medeiros, "Combining multiple views from a distance based feature extraction for text classification," pp. 1–8, 2018. DOI: 10.1109/CEC.2018.8477772.

[5] C. H. P. Ferreira, F. O. De Franca, and D. R. Medeiros, "Combining multiple views from a distance based feature extraction for text classification," in *2018 IEEE Congress on Evolutionary Computation (CEC)*, 2018, pp. 1–8. DOI: 10.1109/CEC.2018.8477772.

[6] J. Nowak, A. Taspinar, and R. Scherer, "Lstm recurrent neural networks for short text and sentiment classification," in *Artificial Intelligence and Soft Computing*, L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, and J. M. Zurada, Eds., Cham: Springer International Publishing, 2017, pp. 553–562, ISBN: 978-3-319-59060-8.

[7] G. Wang, C. Li, W. Wang, *et al.*, "Joint embedding of words and labels for text classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 2321–2331. DOI: 10.18653/v1/P18-1216. [Online]. Available: https://aclanthology.org/P18-1216.

[8] U. P., V. Govindan, and S. Madhu Kumar, "Enhanced sparse representation classifier for text classification," *Expert Systems with Applications*, vol. 129, pp. 260–272, 2019, ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2019.04.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417419302337.

[9] X. Tang, H. Mou, J. Liu, and X. Du, "Research on automatic labeling of imbalanced texts of customer complaints based on text enhancement and layer-by-layer semantic

matching," *Scientific Reports*, vol. 11, no. 1, p. 11 849, Jun. 2021, ISSN: 2045-2322. DOI: 10.1038/s41598-021-91189-0. [Online]. Available: https://doi.org/10.1038/s41598-021-91189-0.

[10] A. Simões, X. G. Guinovart, and J. J. Almeida, "Enriching a portuguese wordnet using synonyms from a monolingual dictionary," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, N. C. (Chair), K. Choukri, T. Declerck, *et al.*, Eds., Portoro, Slovenia: European Language Resources Association (ELRA), May 2016, ISBN: 978-2-9517408-9-1.

[11] H. Lopes-Cardoso, T. F. Osório, L. V. Barbosa, *et al.*, "Robust complaint processing in portuguese," *Information*, vol. 12, no. 12, 2021, ISSN: 2078-2489. DOI: 10.3390/info12120525. [Online]. Available: https://www.mdpi.com/2078-2489/12/12/525.

[12] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning, "Stanza: A python natural language processing toolkit for many human languages," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Online: Association for Computational Linguistics, Jul. 2020, pp. 101–108. DOI: 10.18653/v1/2020.acl-demos.14. [Online]. Available: https://aclanthology.org/2020.acl-demos.14.

[13] L. Neto, "Cia: Citizen contact center agent assistant," M.S. thesis, Instituto Superior Técnico, Jan. 2021.

[14] V. Neves, "Automatic classification of correspondence from a public institution," M.S. thesis, Instituto Superior Técnico, Jan. 2021.

[15] A. C. Forte and P. B. Brazdil, "Determining the level of clients' dissatisfaction from their commentaries," in *Computational Processing of the Portuguese Language*, J. Silva, R. Ribeiro, P. Quaresma, A. Adami, and A. Branco, Eds., Cham: Springer International Publishing, 2016, pp. 74–85, ISBN: 978-3-319-41552-9.

[16] H. Gonçalo Oliveira, J. Ferreira, J. Santos, *et al.*, "AIA-BDE: A corpus of FAQs in Portuguese and their variations," English, in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, Marseille, France: European Language Resources Association, May 2020, pp. 5442–5449, ISBN: 979-10-95546-34-4. [Online]. Available: https://aclanthology.org/2020.lrec-1.669.

[17] P. H. Luz de Araujo, T. E. de Campos, and M. Magalhães Silva de Sousa, "Inferring the source of official texts: Can svm beat ulmfit?" In *Computational Processing of the Portuguese Language*, P. Quaresma, R. Vieira, S. Aluísio, H. Moniz, F. Batista, and T. Gonçalves, Eds., Cham: Springer International Publishing, 2020, pp. 76–86, ISBN: 978-3-030-41505-1.

[18] F. Souza, R. Nogueira, and R. Lotufo, "Bertimbau: Pretrained bert models for brazilian portuguese," in *Intelligent Systems*, R. Cerri and R. C. Prati, Eds., Cham: Springer International Publishing, 2020, pp. 403–417, ISBN: 978-3-030-61377-8.

[19] S. Silva, R. Ribeiro, and R. Pereira, "Less is more in incident categorization," in *7th Symposium on Languages, Applications and Technologies, SLATE 2018, June 21-22, 2018, Guimaraes, Portugal*, P. R. Henriques, J. P. Leal, A. M. Leitão, and X. G. Guinovart, Eds., ser. OASIcs, vol. 62, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018, 17:1–17:7. DOI: 10.4230/OASIcs.SLATE.2018.17. [Online]. Available: https://doi.org/10.4230/OASIcs.SLATE.2018.17.

[20] F. Batista and R. Ribeiro, "Sentiment analysis and topic classification based on binary maximum entropy classifiers," *Proces. del Leng. Natural*, vol. 50, pp. 77–84, 2013. [Online]. Available: http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/4662.

APPENDIX A

Results Summarized Data

TABLE A.1. Results for the first stage using the original dataset, SVM outperform all the other models for this task but Naïve Bayes presented a marginally higher f-score

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.894619	0.921471	0.894619	0.903991
	Yes	0.902093	0.930147	0.902093	**0.911670**
SVM	No	**0.899851**	**0.929783**	**0.899851**	0.910339
	Yes	0.894619	0.934678	0.894619	0.908281
k-NN	No	0.853513	0.887257	0.853513	0.867000
	Yes	0.855007	0.895267	0.855007	0.870560
XGBoost	No	0.826607	0.883714	0.826607	0.848738
	Yes	0.828102	0.903759	0.828102	0.856595
Multilingual-Bert	No	0.869207	0.91050	0.86920	0.883518
	Yes	0.857249	0.9442	0.857249	0.88504

TABLE A.2. Results for the second stage classification using the original dataset, low performance across all models. BERT model was unable to trained for this stage

Model	Processing	Accuracy	Precision	Recall	F-score
SVM	No	0.312977	0.672483	0.312977	0.246029
	Yes	0.293333	0.700265	0.293333	0.242265
Naïve Bayes	No	0.259843	0.577479	0.259843	0.181176
	Yes	0.271318	0.753100	0.271318	0.188976
XGBoost	No	0.166667	0.606909	0.166667	0.184742
	Yes	0.304569	0.791608	0.304569	0.370414
k-NN	No	0.230769	0.738754	0.230769	0.194529
	Yes	0.312500	0.809010	0.312500	0.311697

TABLE A.3. Results for the full classification using the original dataset, k-NN and SVM have similar scores and present decent performance although SVM has a higher precision and f-score

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.750374	0.907025	0.750374	0.810925
	Yes	0.770553	0.92491	0.770553	0.832805
SVM	No	0.786996	0.913812	0.786996	0.837782
	Yes	0.793722	**0.924617**	0.793722	**0.845285**
KNN	No	0.796712	0.887542	0.796712	0.833793
	Yes	**0.803438**	0.883222	**0.803438**	0.837429
XGBoost	No	0.601644	0.842656	0.601644	0.685857
	Yes	0.633782	0.894219	0.633782	0.730709
Multilingual-Bert	No	0.58071	0.82905	0.58071	0.67277
	Yes	0.550822	0.84441	0.550822	0.65165

TABLE A.4. Results for the first stage classification using the augmented dataset, SVM outperform all the other models for this task

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.930493	0.932668	0.930493	0.931215
	Yes	0.933483	0.937128	0.933483	0.934630
SVM	No	0.934230	0.941662	0.934230	0.937099
	Yes	**0.940957**	**0.945378**	**0.940957**	**0.942697**
k-NN	No	0.911809	0.914773	0.911809	0.911943
	Yes	0.904335	0.908641	0.904335	0.905653
XGBoost	No	0.904335	0.918122	0.904335	0.909718
	Yes	0.912556	0.924713	0.912556	0.917385
Multilingual-Bert	No	0.93	0.93025	0.93	0.92831
	Yes	0.936472	0.93789	0.936472	0.93710

TABLE A.5. Results for the second stage classification using the expanded dataset, BERT outperform all the other models for this task

Model	Processing	Accuracy	Precision	Recall	F-score
SVM	Yes	0.578947	0.767832	0.578947	0.525620
	No	0.488095	**0.772963**	0.488095	0.405475
Naïve Bayes	Yes	0.447368	0.480623	0.447368	0.335068
	No	0.472222	0.432330	0.472222	0.370383
XGBoost	Yes	0.534884	0.716058	0.534884	0.535191
	No	0.455556	0.718620	0.455556	0.447290
k-NN	Yes	0.487500	0.751803	0.487500	0.405868
	No	0.432432	0.775594	0.432432	0.330564
Multilingual-Bert	No	**0.65217**	0.64684	**0.65217**	**0.59163**
	Yes	0.55384	0.7719	0.55384	0.483812

TABLE A.6. Results for the full classification using the expanded dataset, SVM outperform all the other models for this task

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.870703	0.931320	0.870703	0.891887
	Yes	0.878924	0.931459	0.878924	0.896761
SVM	No	0.884155	0.921298	0.884155	0.897685
	Yes	**0.890882**	0.924622	**0.890882**	**0.902759**
k-NN	No	0.835575	0.908580	0.835575	0.860282
	Yes	0.837818	0.909102	0.837818	0.863215
XGBoost	No	0.774290	0.877428	0.774290	0.814150
	Yes	0.791480	0.881701	0.791480	0.825520
Multilingual-Bert	No	0.76233	0.9231	0.76233	0.819744
	Yes	0.84679	**0.93258**	0.84679	0.88095

APPENDIX B

Raw Data Results

TABLE B.1. Results for the first stage classification using the raw dataset, models trained with the base dataset.

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.570423	0.681824	0.570423	0.609463
	Yes	0.683099	0.709555	0.683099	0.693986
SVM	No	0.661972	0.717766	0.661972	0.683825
	Yes	0.612676	0.717019	0.612676	0.651171
k-NN	No	0.563380	0.612595	0.563380	0.585015
	Yes	0.563380	0.623250	0.563380	0.586830
XGBoost	No	0.542254	0.716816	0.542254	0.584816
	Yes	0.598592	0.723709	0.598592	0.644556
Multilingual-Bert	No	0.676056	**0.774231**	.676056	0.707960
	Yes	**0.760563**	0.741053	**0.760563**	**0.742202**

TABLE B.2. Results for the second stage classification using the raw dataset. Models trained with the base dataset, BERT model was unable to be trained for this stage.

Model	Processing	Accuracy	Precision	Recall	F-score
SVM	No	0.379310	0.679598	0.379310	0.417400
	Yes	0.193548	0.861290	0.193548	0.198677
Naïve Bayes	No	0.333333	0.771605	0.333333	0.387736
	Yes	0.206897	0.747126	0.206897	0.150739
XGBoost	No	0.030303	0.015949	0.030303	0.016667
	Yes	0.448276	0.712931	0.448276	0.504957
k-NN	No	0.200000	0.496667	0.200000	0.210860
	Yes	0.142857	0.830952	0.142857	0.124868

TABLE B.3. Results for the full classification using the raw dataset, models trained with the base dataset.

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.190141	0.607123	0.190141	0.261915
	Yes	0.295775	0.568331	0.295775	0.375173
SVM	No	0.218310	0.600436	0.218310	0.296846
	Yes	0.295775	0.583975	0.295775	0.375955
KNN	No	0.295775	0.576775	0.295775	0.365618
	Yes	0.401408	0.620604	0.401408	0.464974
XGBoost	No	0.316901	0.529646	0.316901	0.380927
	Yes	0.232394	0.613427	0.232394	0.268572
Multilingual-Bert	No	0.281690	0.580415	.281690	0.355399
	Yes	0.091549	0.547959	0.091549	0.107027

TABLE B.4. Results for the first stage classification using the raw dataset, models trained using the augmented dataset.

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.521127	0.715658	0.521127	0.567160
	Yes	0.704225	0.750431	0.704225	0.719772
SVM	No	0.690141	0.743816	0.690141	0.712144
	Yes	0.676056	0.730931	0.676056	0.699058
k-NN	No	0.619718	0.690142	0.619718	0.644502
	Yes	0.654930	0.696486	0.654930	0.656692
XGBoost	No	0.669014	0.755156	0.669014	0.695326
	Yes	**0.718310**	**0.765323**	**0.718310**	**0.726471**
Multilingual-Bert	No	0.612676	0.704231	0.612676	0.635282
	Yes	0.654929	0.760155	0.654929	0.684413

TABLE B.5. Results for the second stage classification using the expanded dataset, models trained using the augmented dataset.

Model	Processing	Accuracy	Precision	Recall	F-score
SVM	Yes	**0.578947**	0.767832	**0.578947**	**0.525620**
	No	0.488095	**0.772963**	0.488095	0.405475
Naïve Bayes	Yes	0.447368	0.480623	0.447368	0.335068
	No	0.472222	0.432330	0.472222	0.370383
XGBoost	Yes	0.534884	0.716058	0.534884	0.535191
	No	0.455556	0.718620	0.455556	0.447290
k-NN	Yes	0.487500	0.751803	0.487500	0.405868
	No	0.432432	0.775594	0.432432	0.330564
Multilingual-Bert	No	0.029411	0.001960	0.029411	0.003676
	Yes	0.037037	0.001610	0.037037	0.003086

TABLE B.6. Results for the full classification using the expanded dataset, models trained using the augmented dataset.

Model	Processing	Accuracy	Precision	Recall	F-score
Naïve Bayes	No	0.380282	0.662828	0.380282	0.459485
	Yes	0.577465	0.690651	0.577465	0.613549
SVM	No	0.500000	0.654419	0.500000	0.552794
	Yes	0.556338	0.648104	0.556338	0.593348
k-NN	No	0.415493	0.626492	0.415493	0.487297
	Yes	0.514085	0.620943	0.514085	0.556376
XGBoost	No	0.422535	0.658533	0.422535	0.494046
	Yes	0.640845	0.737350	0.640845	0.671151
Multilingual-Bert	No	0.654929	**0.760155**	0.654929	**0.684413**
	Yes	**0.683098**	0.671738	**0.683098**	0.674962