FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

# Learning to classify a subject-line quality for email marketing using Data Mining techniques

**Maria João dos Santos Aguiar e Mira Paulo**

MASTER'S DISSERTATION

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Vera Lucia Miguéis Oliveira e Silva

Second Supervisor: Ivo Pereira

July 23, 2019

# Learning to classify a subject-line quality for email marketing using Data Mining techniques

**Maria João dos Santos Aguiar e Mira Paulo**

Mestrado Integrado em Engenharia Informática e Computação

July 23, 2019

# Abstract

With the accelerated advances in technology, companies started to replace traditional marketing methods with technology-intensive techniques, known as the digital marketing. Among the technology-intensive procedures, email marketing arose as one of the most adopted strategies by firms. Although being considered one of the most preferred methods to reach customers, email marketing remains a complex field resulting in companies struggling to achieve a high rate of opened emails and a low rate of unsubscribing users.

E-goi customers send millions and millions of email campaigns every day through the platform. Like E-goi, also other platforms and companies send a massive amount of email campaigns and that is the reason why email marketing started to be such a challenging process. Consumers are confronted with a continuously growing direct email volume in the mailbox that brings increased competition for their limited attention. Naturally, with the overwhelming amount of emails in the mailbox, the recipient needs to filter the most relevant content to read.

In responding to this competition, companies are looking to increase the percentage of recipients that open the direct mail (Open Rate) while boosting the number of subscribers that click at least one link in the email marketing campaign (Click Rate). An email campaign with a high open rate, meaning the majority of the recipients opened and read the message, is deemed as a successful campaign by business standards. Of necessity, it is important to be aware of which elements increase the chance of a customer to open an email. The sender name and the subject-line, both play a big part in getting emails opened and read. In fact, the curiosity and interest in examining the content of an email depend directly on those characteristics.

The main goal of this project consists in using predictive methods to build a program capable of classifying an email subject regarding the respective quality, evaluated in a 1 to 5 range. Based on the big amount of data collected by the company and available for use, this project intends to help email campaigns editors predicting customer behavior when facing the campaigns they are designing. A data set of 140.000 subjects was used to validate the proposed model. Different techniques such as Random Forest, Decision Trees, Neural Networks, Naive Bayes, Support Vector Machines, and Gradient Boosting were applied. The Random Forest technique revealed to be the one generating significantly better classification results.

In conclusion, this thesis provides a prosperous and valuable tool to help email campaigns' editors on overcoming the daily obstacles during the creation of their campaigns.

# Resumo

Com a rápida evolução do mundo tecnológico, o *marketing* tradicional teve necessidade de se adaptar a uma nova realidade de consumo *online*, através da qual surge uma nova área de trabalho denominada de *marketing* digital. Atualmente, apesar da grande variedade de canais de comunicação existentes *online*, como redes sociais ou *blogs*, o *email marketing* continua a destacar-se por ser uma das formas mais eficazes de aproximação com o cliente. Apesar de todas as vantagens inerentes ao *email marketing*, as empresas continuam a enfrentar dificuldades devido às reduzidas taxas de aberturas e elevadas taxas de cancelamento de subscrição.

Tal como os utilizadores da plataforma E-goi, também outras plataformas e empresas enviam milhares de campanhas de *email*. Com o crescente número de campanhas que são enviados diariamente, os consumidores defrontam-se com uma imensa quantidade de *emails* nas suas caixas de correio. Sabendo que um cliente regular não possui nem tempo nem interesse em abrir e ler todo o conteúdo que recebe, torna-se importante refletir sobre o que torna, à primeira vista, um *email* mais interessante que outro e como captar a reduzida atenção dos leitores.

De facto, para um *email* ser aberto e consequentemente lido, é necessário despertar curiosidade e interesse no leitor. Uma vez que o assunto e o remetente são as únicas informações disponibilizadas ao cliente quando acede à sua caixa de correio, são consideradas como grandes influenciadores das taxas de abertura (OR).

No sentido de melhorar as taxas de abertura dos *emails* enviados, pretende-se criar uma ferramenta de suporte aos editores de campanhas de *email*, que prevê a qualidade de um determinado assunto. Com base no histórico de *emails* enviados pela plataforma E-goi e disponibilizado para uso, procura-se criar um modelo que classifique um assunto de *email* quanto à sua qualidade, numa escala de 1 a 5. Foram usadas 140.000 campanhas para avaliação do modelo e testados diferentes algoritmos de classificação, tais como *Random Forest*, *Decision Trees*, *Neural Networks*, *Naive Bayes*, *Support Vector Machines* e *Gradient Boosting*. Contudo, a técnica que gerou melhor resultados foi o *Random Forest*.

Concluindo, esta dissertação contribuiu para o desenvolvimento de uma ferramenta que dará suporte aos editores de campanhas de *email*, apoiando-os nos obstáculos diários que enfrentam durante o processo de criação.

# Acknowledgements

First, I would like to thank my thesis supervisor Vera Miguéis, for the unfailing support, availability, and assistance. For consistently encouraged me to do better and for steered me in the right direction whenever was needed. I am sincerely grateful for the knowledge, ideas and, finally, for the friendly and kind personality.

I would also like to thank E-goi for all the support, guidance and for the facilities, providing me a place to work during my internship with excellent conditions. In particular, to Ivo Pereira and Duarte Coelho. Ivo Pereira for the endless interest in knowing what I was doing and how I was proceeding, always taking care of my well-being. Duarte Coelho for his passionate participation and input and for the huge contribution to the success of this project.

I must express my very profound gratitude to my family for supporting and encouraging me throughout my years of study. If I am who I am, I owe to them. A special thank you to my parents, for believing in me and for supporting me emotionally. To my sister for being my inspiration.

Finally, I would like to thank my friends for helping me become a better person, for the friendship, for listening and offering me advice and for supporting me through this entire process.

This accomplishment would not have been possible without them. Thank you.

Maria João Mira Paulo

# Agradecimentos

Em primeiro lugar, gostava de agradecer à Professora Vera Miguéis, pelo apoio e pela disponibilidade incondicional. Pela motivação, pelo interesse, por me desafiar a fazer melhor e por me direcionar para o caminho certo. Pelas suas ideias, pelas suas críticas e opiniões, pela forma simpática e carinhosa com que sempre lidou comigo.

Gostava de agradecer à E-goi pelo suporte e pelas excelentes condições de trabalho que me proporcionaram durante a realização deste estágio. Em especial, ao Ivo Pereira e ao Duarte Coelho. Ao Ivo Pereira pela constante preocupação e acompanhamento, por ter estado sempre presente em todas as fases de desenvolvimento deste projeto. Ao Duarte Coelho pela sua paixão pela área, pelos seus intermináveis conhecimentos e por todo o apoio e motivação que levaram ao sucesso deste trabalho.

Um agradecimento muito especial à minha família por me ter encorajado a ser melhor e por me ter visto crescer nestes anos de estudo. Se sou o que sou, a eles o devo. Em particular aos meus pais, pela incansável dedicação ao longo de todo o meu percurso académico, por sempre acreditarem em mim e nas minhas capacidades. À minha irmã, por ser quem me inspira a ser mais e melhor.

Por último, um agradecimento aos meus amigos por terem estado sempre presentes quando mais precisei, por me terem ajudado a crescer e a ser uma pessoa melhor.

Nada disto teria sido possível sem todas estas pessoas. O meu muito obrigada.

Maria João Mira Paulo

*“At the end of the day,*
*we can endure much more than we think we can.”*

***Frida Kahlo***

# Contents

# List of Figures

# List of Tables

# Abbreviations

| | |
|---|---|
| ROI | Return on Investment |
| OR | Open Rate |
| BR | Bounce Rate |
| CTR | Click Through Rate |
| KR | Keeping Rate |
| MAT | Marketing Automation Tools |
| SVM | Support-vector Machine |
| CRISP-DM | Cross Industry Standard Process for Data Mining |
| ANN | Artificial Neural Network |
| HTML | HyperText Markup Language |
| CSS | Cascading Style Sheets |
| TF-IDF | Term Frequency-Inverse Document Frequency |
| IP | Internet Protocol |
| REST | Representational State Transfer |
| UI | User Interface |
| PCA | Principal component Analysis |
| NCV | Nested Cross Validation |

# Chapter 1

# Introduction

## 1.1 Context

In today's technology-driven world, with so many people connected to the Internet, there is no more effective way to reach customers than through digital marketing channels. Indeed, it is understandable why digital marketing is overpowering traditional marketing and the reason why online marketing became an integral part of any modernized organization [Apo15].

When talking about digital marketing, email marketing arises as one of the most preferred methods of contact by firms [Dan17]. The capability to reach customers in a very short time, the easy recovery of investment and the quickness to get accurate statistics regarding a specific campaign make email marketing one of the most powerful fields among digital marketing techniques.

With email marketing acting as a starting-point, marketing automation is considered the next generation of digital marketing "for users focused on orchestrating the customer journey" [Dan17]. Marketing automation refers to the use of software with the goal of automating marketing actions, such as campaign management and customer segmentation. With the use of marketing automation, processes that otherwise would be done manually, can be executed automatically and much more efficiently [Dan17]. Companies using marketing automation tools often observe more web traffic and business transactions, achieving greater customer retention and loyalty.

In order to overcome the repetitiveness of tasks and automate marketing processes by easing the process of creating, managing and distributing campaigns over different digital channels, organizations like E-goi were created. E-goi is a Multichannel Marketing Automation Platform, providing marketing automation tools through different communication channels such as email, SMS, push, voice, and social media. E-goi offers marketing automation solutions to all types of organizations and, by helping companies to grow leads, automate their marketing strategies, control costs and increase their ROI (Return on Investment), E-goi has become a suitable solution for several well-renowned companies such as AKI, Continente, Pingo Doce, Salsa, Farfetch, Millenium and BNP Paribas.

Figure 1.1: E-goi logotype

Headquartered in Porto, Matosinhos, the E-goi company already owns 390.000 accounts spread around different countries, with the big ones being Portugal, Brazil, Colombia, Spain and most of Latin America.

## 1.2 Motivation and Objectives

According to VentureBeat [VB 16], email marketing remains to be the channel generating the highest ROI for marketers, also being considered the most effective channel for customer retention, consumer awareness, and customer conversion. However, this technology effectiveness is being challenged, since every modernized company is adopting it, resulting in an overwhelming volume of emails.

In fact, consumers receive, each day, a massive amount of messages, which increases the competition for their limited attention [FFK+13], resulting in emails which remain in the inbox forever unopened. In responding to this competition, companies are searching for a way of increasing customer engagement, more specifically, increase the percentage of recipients that open the direct email and additionally, a way of boosting the number of subscribers that have clicked on at least one link in your email marketing campaign.

Email engagement metrics can be optimized through better customer segmentation, a better knowledge of customer-relevant content, better email sending time or by the envelope of the email. The envelope means the external characteristics of the email, right before opening it and, in fact, the curiosity and interest in examining the content of an email depend directly on those characteristics. As stated by Feld et al. [FFK+13], "the envelope and its design create a certain degree of curiosity and interest in further investigating the content of the mail item". The two unique pieces of information recipients have about the email at first glance are the sender and the subject-line. Therefore, a consumer's decision to open the email critically depends on these two factors.

E-goi customers send millions and millions of email campaigns every day through the platform. Like E-goi, also other platforms and companies send a massive amount of email campaigns. With email overload being a reality in today's fast-paced business environment, customers need to filter out unnecessary or irrelevant emails by the subject field and that is the reason why it plays such a significant role on achieving high email engagement levels.

Every email campaign is built around very specific objectives that a business intends to accomplish. However, a necessary precondition for any of those distinct emails to be considered successful emails is to be opened and read by the user.

Therefore, having high open rates becomes critical to business success [BP15]. The main goal of this project is to develop a predictive model capable of classifying a subject field of an email, according to the respective potential to engage customers. Using the big amount of data collected and available by the company E-goi, this project intends to support and facilitate the editor's decision task of choosing a subject-line for a specific email campaign.

## 1.3 Dissertation structure

Apart from the Introduction, this document contains five additional chapters. Chapter 2 provides an overview of the related work on this field, including a literature review and relevant existing technologies. Additionally, this chapter contains some relevant concepts about Marketing Engagement Metrics and Marketing Automation Tools. Chapter 3 gives a summary of the existing methods that could be employed to solve the above-stated problem, in this particular case, Data Mining multi-class classification techniques. Chapter 4 explains, step by step, the project implementation, focusing on business understanding, data understanding, exploration and preparation, modeling, evaluation, and deployment. Chapter 5 describes the different experiences made in order to achieve the most accurate model. Finally, Chapter 6 points out some final conclusions and highlights potential future work for the continuation of this project.

# Chapter 2

# Related work

This chapter provides a review of the previous work accomplished on Email Marketing, particularly, on topics related to the improvement of customer engagement metrics. Since an email campaign is only considered a successful email if it is open and read by the receiver, big efforts have been made in order to explain and increase the number of recipients who opened and read the email.

Firstly, relevant marketing concepts are highlighted to a better understanding of the problem scope, and a background of the company and the effort made in order to help email editors on improving the quality of the campaigns is given. Subsequent to the overview of the previous research on the area of increasing email marketing engagement metrics, related frameworks available at the time are described, scrutinized and compared.

## 2.1 Engagement Metrics

Email marketing engagement metrics are a measurement of how customers interact with the email campaign, if they open it, read the content or click on links. The engagement metrics help on refining the contacts list, deciding what type of content is more valuable and can lead to more efficient email campaigns. The main engagement metrics are summarized below:

- **Open Rate (OR)** is the percentage of the total number of subscribers who opened an email campaign. Low open rates could be directly related to, for example, having uninteresting subject-lines, unqualified, outdated and inactive subscribers or unsegmented email lists.

- **Click Through Rate (CTR)** is the number of recipients that click on any given link within your email. The CTR is quite an imprecise measure since a vast variety of factors can influence it. Specifically, it can depend on the subject-line, email content, content design, link positioning, time of the day or email length.

- **Bounce Rate (BR)** is the percentage of email addresses in the subscriber list that did not receive the email because of being returned by a recipient mail server. The BR can be directly related to holding invalid email addresses in the subscribers list.

- **Keeping Rate (KR)** is the percentage of recipients that keep the mailing after opening the envelope.

## 2.2 Marketing Automation Platforms tools for improving engagement metrics

Companies have been adopting marketing automation platforms because they are considered truly valuable tools in measuring the emotional connection between the customer and the brand, and because they enable an easy estimation and evaluation of the marketing engagement levels. Since high engaged customers are acknowledged to be more loyal to the brand, to buy more and more frequently, it is crucial to provide high-quality campaigns with positive customer engagement results.

The most common strategy to measure the effectiveness of an email message is the well-known A/B test, provided by almost all marketing automation platforms.

### 2.2.1 A/B tests

Nowadays, most marketing automation platforms provide the possibility of using A/B tests, which is one of the most well-known and meaningful ways to measure the quality and customer response to a campaign.

The aforementioned technique examines how small changes in the same email impact the results of a campaign. Hence, after creating an email campaign, the editor creates another version, option B, with a single variation. Later, the editor can analyze the performance of both campaigns and, for example, pick the winning version to send to the rest of the subscribers list. An A/B test allows measuring, for example, the impact of two different subjects on the engagement levels.

Multivariate testing is an extension of A/B testing, differing from each other in the number of versions tested at the same time. Whereas the first one only compares two email campaigns, the last one supports more than two.

### 2.2.2 E-goi Content Checker

Besides the well-known A/B tests, the E-goi platform provides a distinct tool, named Content Checker. This tool is composed of two distinct modules: Source Code Module and SpamAssassin Module. Whereas the first one provides HTML and CSS analysis by checking, for example, broken links and CSS good programming practices, the second one is a popular email spam filter used by many corporate networks named Apache SpamAssassin program. Content Checker lists some tips to the editor, in order to improve the quality of the content of the email and security tips in order to avoid the campaign to be considered spam by an email provider.

The subject analyzer tool intended to be developed during this thesis should be, afterward, incorporated as a new module to the specified Content Check Tool.

## 2.3 Literature Review

To get a better understanding of the following studies, this section was divided into Primary Analysis and Secondary Analysis. Whereas a Primary Analysis involves primary data collected for the purpose of addressing the specific research problem, a Secondary Analysis handles secondary data, which is data not necessarily collected to tackle the current research problem.

A Primary Analysis usually involves focus groups, surveys or interviews, while a Secondary Analysis usually includes previously completed studies or data collected to support firm operations.

### 2.3.1 Primary Analysis

According to prior researches, when scanning the inbox, people prioritize some emails over others [WDK11]. With the overwhelming volume of emails that people receive each day, it becomes impractical to pay attention to all that content. Wainer et al. [WDK11] attempt to reveal the reasons why people choose to open certain emails over others, suggesting that driving attention to an email is a function of the inferred utility of message content and curiosity. A think-aloud study concluded that people read emails that are directly related to their work or other important content and also that messages with moderate levels of uncertainty are more likely to be open because of the curiosity and desire created to open the email.

Sahni et al. [SWC16] empirically studied the content of an email, specifically, quantifying the role of personalized content as customer-specific information. As firms usually have information about consumers, such as the customer's name, it can potentially be incorporated into the marketing message to personalize it. Non-informative content was proved to be valuable in garnering customer attention and interest, increasing the probability of the recipient to opening and reading it.

Feld et al. [FFK$^+$13] developed empirical studies to evaluate the effect of direct mail design characteristics such as visual design, color, illustrations, sender identity, sender's name, logo and personalization on the OR and KR. The study concludes that the design characteristics that potentially generate curiosity can widely influence the OR, offering specific guidelines for email campaigns editors, such as using colors with caution, using sender identity with care and using personalization as a differentiation factor. Both Feld et al. [FFK$^+$13] and Sahni et al. [SWC16] concluded that personalization can contribute to increasing campaigns success.

### 2.3.2 Secondary Analysis

The secondary analysis was divided into Non-linguistic and Linguistic-analysis. While in the first one, only the morphology of the words was studied, in the second one, the meaning of words was also taken into consideration.

#### 2.3.2.1 Non-linguistic analysis of subject-lines

The Apostolopoulos [Apo17a]'s approach toward predicting the OR was based on patterns found on the subject-lines of the most opened emails and disregarding the meaning of words. Therefore, the author attempts to foretell the number of openings through a Random Forest Regressor algorithm, considering the subject-line morphology, the day of the week and also the importance and value of the email sender based on the performance history. Regarding the morphology, the author considered the effects of emoji, exclamation marks and the number of words.

#### 2.3.2.2 Linguistic analysis of subject-lines

Balakrishnan and Parekh [BP15] also intended to predict the open rate of an email based on the subject-line characteristics. This time focusing not only on syntactical and historical features but also on the influence of different keywords in the subject-line. In fact, it is not right to affirm that all the keywords in a subject-line have the same influence on the open rate. Using a Random Forest Regressor algorithm, the authors could see how different keywords on the same subject were performing, assigning a score to each of them.

Apostolopoulos, on his second study [Apo17b], proposes a different approach to the prediction of the open rate. This time, measuring the impact of each word and also the impact of patterns, basically the position of the word inside the subject-line through the application of a Gradient Boosting Regressor. An entity recognition on the subject-line was run in order to remove stop words, numbers, punctuation, currency, percentages, and emojis, and then, the remaining words were stemmed. Therefore, having only the root words[1], without the prefixes and suffixes, and a short wordlist, each one of the root words was scored.

According to Miller and Charles [MC17], the subject-line and email address of the sender are the main deciding factors for one to open the email or leave it. The author analyzed the email subject-lines in a psychological point of view, studying their "effect in a person when he/she reads it and the decision he/she makes to open that email or neglect it" [MC17]. Hence, the authors studied the emotional effect of the subject-line on the OR, through a lexicon-based approach, where the adjectives in the subject-line were classified in one of nine emotion categories between trust, joy, surprise, anticipation, peace, sad, anger, fear and disgust. Furthermore, a Support Vector Machine classifier was used to classify the subjectiveness of the email subject-lines as a fact or opinion, and the opinion induced when reading an email as positive, negative or neutral. The presence of adjectives, verbs, localization or an organization name, the occurrence or non-occurrence of certain

[1] For example, the root word of waiting and waited, is wait.

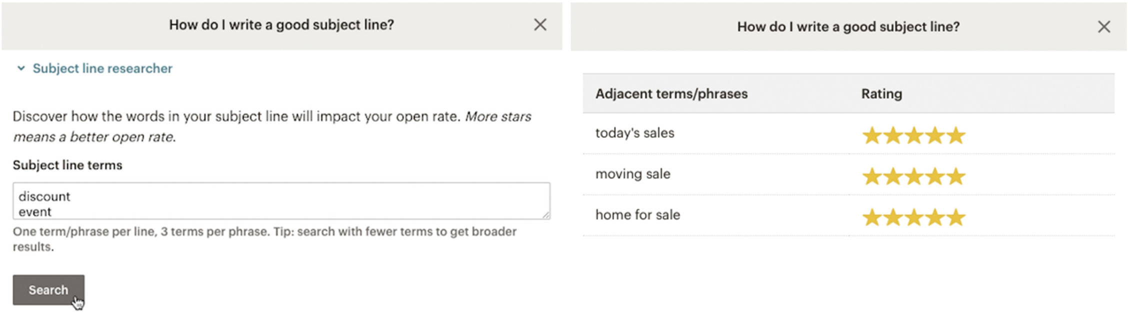
terms, as well as, the number of words and characters were analyzed regarding the act of opening an email or rejecting it.

## 2.4 Related methodologies

Additionally to the above-described research, newly technologies in this field are introduced and described in the following section.

### 2.4.1 MailChimp subject-line Researcher

Given that a successful email campaign starts with a subject-line that grabs the attention of the subscribers, the famous and well known MailChimp[2], a marketing automation platform, developed a subject-line researcher tool that is able to show the effectiveness of different keywords. As shown in Figure 2.1, when searching for a keyword, MailChimp will compare that term to all subject-lines ever sent through MailChimp, listing related terms and phrases with the respective effectiveness of the word, in a 5-star rating system.



How do I write a good subject line?
×
Subject line researcher

Discover how the words in your subject line will impact your open rate. *More stars means a better open rate.*

Subject line terms

discount
event

One term/phrase per line, 3 terms per phrase. Tip: search with fewer terms to get broader results.

Search
How do I write a good subject line?
×

| Adjacent terms/phrases | Rating |
|---|---|
| today's sales | ★★★★★ |
| moving sale | ★★★★★ |
| home for sale | ★★★★★ |

Figure 2.1: MailChimp subject-line researcher tool

This tool does not support a subject analysis. Instead, it receives a list of words and returns, not the quality of each of the given words but a list of related terms and the respective effectiveness.

### 2.4.2 CoSchedule Email subject-line Tester

Another existing tool is the CoSchedule Email subject-line Tester[3] which is able to classify a subject quality, from 1 to 100, also providing clear feedback in order to optimize the subject-line even further.

Taking into consideration the words that increase openings, the words that decrease openings, the effects of the case, the presence of numbers, character count, word count, and emoji count, this tool helps editors optimizing subjects line before sending it to the subscriber list. Although being an innovative tool, it does not interpret the words or the context of the words contained in the subject. Indeed, it checks if the subject contains any word of a pre-defined list of words. For

[2] https://mailchimp.com/

[3] https://coschedule.com/email-subject-line-tester

example, the words "% off" or "voucher" belong to the list of words that increase openings and, the words "cash" and "www" are examples of words that decrease openings.

## 2.5 Conclusion

To sum up, even though the efforts made in the investigation field to influence customer behavior when receiving an email campaign, the literature on this topic is very incipient.

Regarding the literature that uses secondary data, the available studies do not evaluate the impact of using personalized messages, as addressing the subscriber by his/her name or email. Moreover, these studies lack on investigating the influence of the industry to which the campaign is sent. For example, if whether an email talking about Education could impact differently from an email about Hotels offers. Furthermore, fail on undervaluing the impact of the country to where the email is sent. Indeed, the median of the OR is absolutely different depending on the country [Wat18]. Finally, the utilization of machine learning techniques to support these studies is still very limited to a small set of techniques.

Solutions in the market continue to be scarce. Most of the websites that are seeking to help customers on choosing the best subject-line offer merely descriptive guidelines and, hence, most of them, end up easily outdated. Moreover, none of the actual technologies incorporate Machine Learning and there is where this project distinguishes itself. To operate in the fast-changing industry, this technology should be automatically updated, with new data and, hence, with the latest trends.

# Chapter 3

# Data Mining Projects and Classification Algorithms

This chapter explains major concepts in Data Mining field. For a better understanding of the thesis implementation phase, some key points on the development of Data Mining projects are highlighted. Also, this chapter bears on identifying and describing some Data Mining algorithms that could be used along with this project.

## 3.1 Data Mining

With the fast-moving advances in data collection and data storage technology, organizations have been accumulating huge amounts of data. Although recognizing the importance and value of that data, obtaining useful information continues to be extremely challenging [TSK$^+$06]. The information, patterns, and relationships that are hidden on this data were proven to be absolutely valuable since they can be used to make predictions that support businesses such as Medicine, Science, and Engineering. Data mining is the process of automatically discovering useful information in large data repositories, identifying patterns and association rules, that otherwise would remain unknown [TSK$^+$06]. Data mining techniques can, for example, contribute to customer profiling, targeted marketing, workflow management, store layout, and fraud detection.

### 3.1.1 Data Mining Process and Tasks

The CRISP-DM, Cross-Industry Standard Process for data mining projects, provides an explicit structure and clear guidelines for the data science journey from data to wisdom [Joh]. According to the CRISP-DM, the data mining process is a cyclical process and consists of six distinct transformation phases that are responsible for converting raw data into useful data. It should be considered that the sequences of the phases are not strict and the arrows could represent loops depending on the outcome of the specific phase.

This standard process intends to make large data mining projects less costly, more reliable, repeatable, manageable, and faster, being independent of both the industry sector and the technology [Wir00]. As shown in Figure 3.1, the process consists of six phases, namely:

1. **Business Understanding:** The first phase focuses on getting a clear understanding of the problem, scope and respective goals. Also, in this phase, the requirements, from a business perspective, should be considered.
2. **Data Understanding:** Starts by collecting data and, then, getting to know the data characteristics. In this phase, it is important to discover the first insights into the data, being aware of the data quality issues.
3. **Data Preparation:** Construction of the final data set, ready for the modeling. This phase could involve tasks such as data cleaning, data clustering or data formatting.
4. **Modeling:** Various modeling techniques and methods are selected and applied in order to identify patterns in the data.
5. **Evaluation:** After applied multiple models, the results should be evaluated and balanced depending on the business goals. At the end of this phase, a decision on the use of the data mining results should be reached.
6. **Deployment:** Reorganize the created model, presenting it in a way that the customer can easily use it.

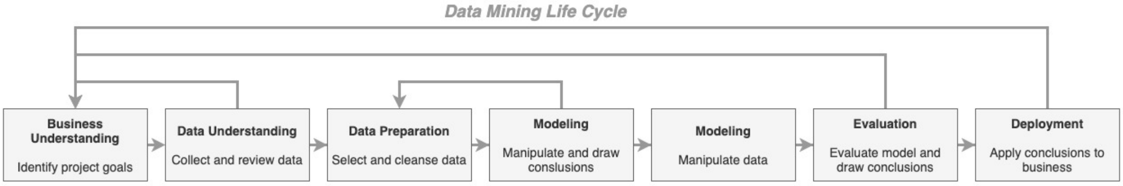


Figure 3.1: Data Mining Process

There are two main data mining modeling types that could be applied to the Modeling phase, i.e Predictive and Descriptive. While the descriptive model recognizes the designs or relationships in data, discovering the properties of the data studied, the predictive model makes authoritative predictions about the future [Sen15]. The most common techniques associated with the mentioned models are:

- **Association Rule Discovery**, a descriptive method which detects dependency rules in order to identify an occurrence of an item based on the occurrences of other items.
- **Clustering**, a descriptive method which groups individual pieces of data together to form a structured opinion based on one or more attributes or classes. At a simple level, clustering creates a meaningful cluster of objects which have similar characteristics.

- **Classification**, a predictive method which assigns one object to one of several predefined categories or groups.

- **Regression**, a predictive method which predicts the value of a given feature based on the values of other features in the data, being able to forecast these attribute values for new cases. The difference between regression and classification is that regression deals with numerical or continuous target attributes, whereas classification deals with discrete or categorical target attributes.

### 3.1.2 Supervised Learning Methods

Data Mining algorithms can also be classified as Supervised and Unsupervised Algorithms. Supervised algorithms include Classification and Regression. In Supervised Learning, a function should be learned so that it would be capable of mapping an input to an output based on example input-output pairs, i.e. training examples. By analyzing labeled training data, a function is inferred, so that it can be used for mapping new examples.

In this section, supervised learning algorithms that could potentially fit the problem at hand, in specific, predictive models for classification, are described.

#### 3.1.2.1 Naive Bayes

The Naive Bayes [Lew98] is a probabilistic machine learning algorithm, based on the Bayes' Theorem conditions on the independent and equal contribution of each feature to the outcome. The Bayes' Theorem determines the probability of an event occurring given the probability of another event has already occurred. Mathematically, the Bayes theorem is stated by the following equation:

$$P(Y|X) = \frac{P(X|Y) * P(Y)}{P(X)} \tag{3.1}$$

In equation 3.1, Y acts as the outcome to predict, and X as the evidence or features. The Bayes Rule is a way of predicting P(Y|X), in other words, the probability of Y given X, from P(X|Y), known from the training data set. In the case of dealing with a multi-classification problem, for each class Y, P(Y|X) is calculated and the class with the highest probability wins.

#### 3.1.2.2 Decision Tree

Based on the concept of divide and conquer and, through a set of if-then-else operations, a problem is solved through a tree representation where, in specific for classification problems, the leaves represent the output class and the branches correspond to the features. The basic idea of a decision tree [Qui86] is to go from the root, including all observations, to conclusions about the item's target value, represented in the leaves, through the satisfaction of the conditions expressed in the branches.

A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values, usually obtained through the entropy and information gain calculation. The entropy measures the heterogeneity of a specific sample and the information gain estimates the reduction on entropy resulting from the division of the sample based on one attribute. Constructing a decision tree is all about continuously finding the attributes which provide the highest information gain until all the samples belong to the same class.

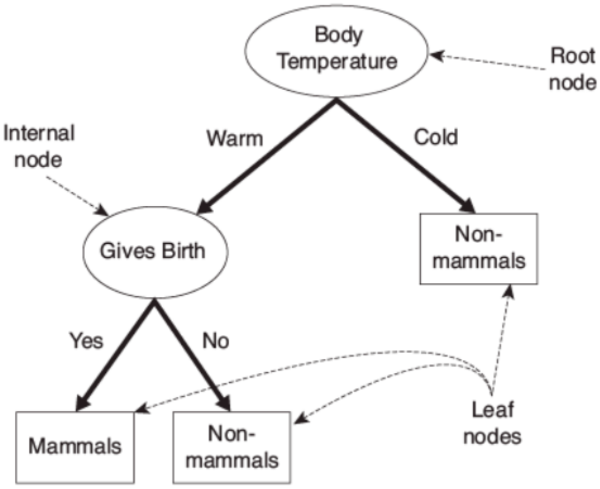


Figure 3.2: Example of a decision tree for the mammal classification problem [TSK$^+$06]

#### 3.1.2.3 Ensemble Methods

Ensemble methods [OG10] are known for improving prediction accuracy by aggregating the predictions of multiple predictors and can be classified into bagging or boosting algorithms.

Even though ensemble methods can be used with any classifier, this section focuses on ensemble methods applied to Decision Trees. Although a single decision tree may not be a very good predictor, very powerful models can be obtained by combining the results of several decision trees classifiers.

Before getting into the following algorithms, it is recommended to understand clearly the Decision Tree Algorithm, explained in Section 3.1.2.2.

- **Bagging** [Bre04] models aspire to reduce the variance of predictions by creating subsets of data from the training sample. Each collection of subset data is used to train a specific decision tree. The final estimation consists of the average of the multiple estimations calculated by each decision tree.

  The **Random Forest Algorithm** [Bre01] is an example of ensemble methods based on bagging, working for both classification and regression. As shown in Figure 3.3, the Random Forest learning method operates with a bag of decision trees, each one considering a random subset of features and having access to a random set of training data points. The output value becomes the mode of the classes, regarding classification, and the mean prediction in the case of regression, of all the individual trees. This algorithm achieves robust overall predictions because of the diversity of results and since the output is not swayed by a single atypical data source. Furthermore, the Random forest algorithm fixes the decision tree algorithm problem of overfitting to their training set.

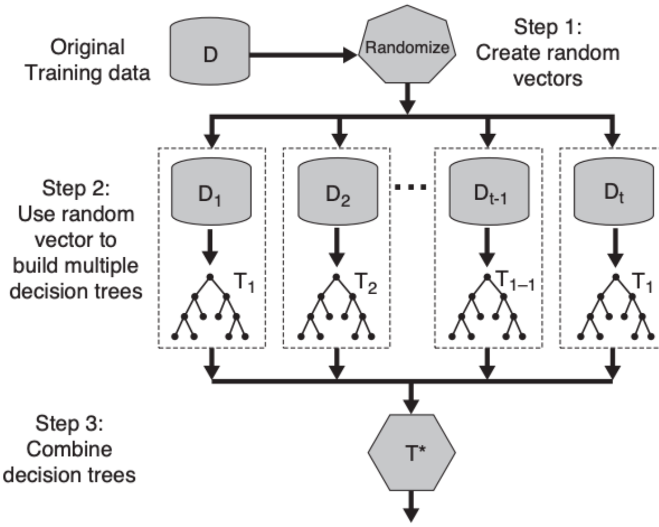

Original Training data
D
Randomize
Step 1: Create random vectors
Step 2: Use random vector to build multiple decision trees
D1
D2
Dt-1
Dt
T1
T2
T1−1
T1
Step 3: Combine decision trees
T*

Figure 3.3: Random Forests [TSK$^{+}$06]

- **Boosting or Gradient Boosting** [Fri02] is a boosting ensemble method. Whereas in bagging algorithms, independent predictors are combined using some model averaging techniques, in boosting algorithms, predictors are not grouped independently, but sequentially. This technique employs the logic in which the subsequent predictors learn from the mistakes of the previous predictors. An example of a boosting model is the AdaBoost technique.

#### 3.1.2.4 Support Vector Machine

Support Vector Machine (SVM) [Ste18] is a supervised machine learning algorithm which can be used for both classification and regression problems, although being mostly used for classification projects.

This model is based on the concept of decision planes that define decision boundaries called hyperplanes. One unique aspect of this technique is that the decision boundary is defined using only a subset of the training examples, known as the support vectors [TSK$^{+}$06].

Consider the two classes, i.e. *square* and *circle*, represented in Figure 3.4. A hyperplane is a frontier such that all the squares reside on one side of the hyperplane and all the circles reside on the other side, taking into account that the selected hyperplane is the one that maximizes the margin between the two classes with the help of support vectors. Looking at Figure 3.4, and considering each data item as a point in an n-dimensional space, where n is the number of features, and the value of each feature being the value of the particular coordinate, we can perform classification by finding the hyperplane that separates clearly the two classes.

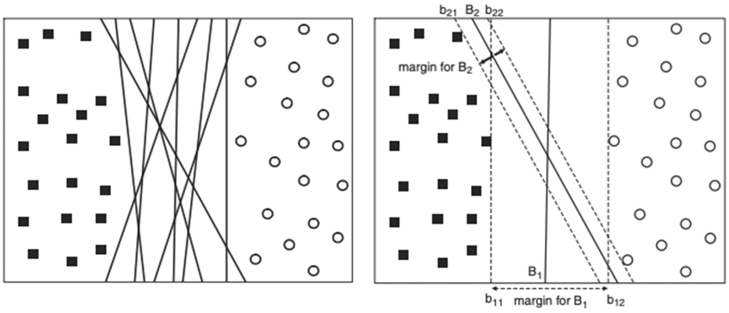


Figure 3.4: Support Vector Machine hyperplan decision [TSK+06]

#### 3.1.2.5 Artificial Neural Networks

An Artificial Neuron Network (ANN) is a computational model that imitates the biological Neural Networks of the human body. As represented in Figure 3.5, there are three different layers in a ANN:

- **Input Layer**, responsible for bringing the initial data into the system, to be processed by the subsequent layers of the artificial network.
- **Hidden layer**, a layer where artificial neurons take in a set of weighted inputs and produce an output through an activation function.
- **Output layer**, responsible for producing given outputs for the program.

Although there are different types of ANN, this section is focused on the Multilayer ANN because of being often applied to supervised learning, due to its capability for solving complex classification and regression problems.

A Multilayer ANN, or Feed Forward Neural Network, contains one or more hidden layers and can learn nonlinear functions, using a backpropagation method to train the network. Training involves adjusting the parameters, or the weights, of the model in order to minimize the overall error. It is also called feed-forward because, in this specific network, the nodes in the first layer are connected only to the nodes in the next layer.

Initially, with the backpropagation method, all the edge weights are randomly assigned. For every input in the training data set, the ANN is activated and its output is observed. The given output is compared with the desired one, and the error is propagated back to the previous layer. This error is recorded and the weights are adjusted accordingly. This process is repeated until the output error is below a predetermined threshold.

The activation function of a node defines the output of that node given a specific input or a set of inputs. There exist several activation functions that can be applied depending on the problem at hand, for example, the sigmoid function, Tan-h, Softmax, ReLU or Leaky ReLU. An activation function allows the model to produce a result (target variable, class label, or score) that varies non-linearly with its explanatory variables [Zer18].

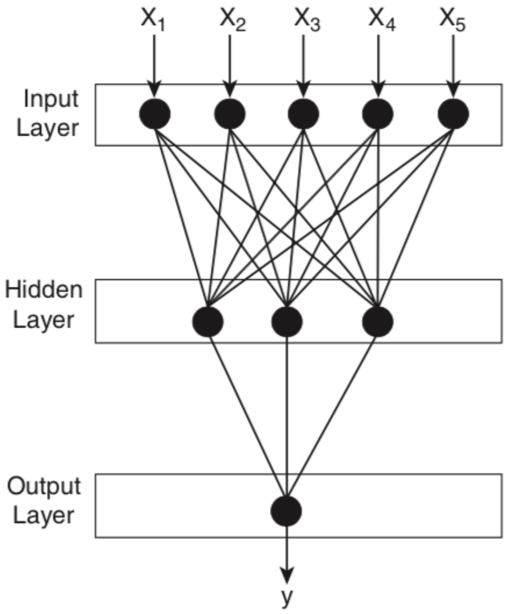


Figure 3.5: Example of a Multilayer feed-forward artificial neural network [TSK$^+$06]

Figure 3.5 illustrates a Multilayer ANN where $X_n$ represents a node in the input layer, corresponding to the input feature, and $y$ is the predicted value.

### 3.1.3 Model Evaluation

The model evaluation is the process of choosing the most suitable algorithm for a particular business problem. Regarding the project at hand, this section is specifically focused on how to evaluate multi-class classification models.

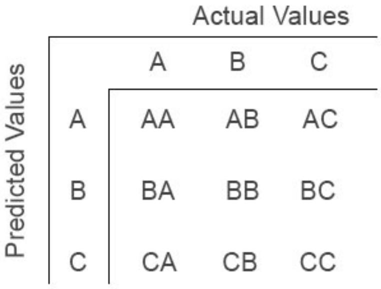
#### 3.1.3.1 Confusion Matrix

A confusion matrix [Tin10] is a specific table layout that summarizes the classification performance of a classifier with respect to some test data. Almost all of the performance metrics are based on a confusion matrix and on the numbers inside it. Considering the multi-class confusion matrix, on Figure 3.6, each row of the matrix represents the instances in a predicted class and each column corresponds to the instances in an actual class.

In Figure 3.6, {A, B, C} refer to the three existent classes. For example, AA refers to the number of observations classified correctly, where A was classified as A, whereas the AB refers to the number of observations belonging to B and classified as A.

The following terminology is often used together with the confusion matrix:

- **True positives (*tp*)** which corresponds to the number of positive examples correctly predicted by the classification model. When analyzing, for example, class A, it would correspond to the value of AA.



Actual Values

| Predicted Values | A | B | C |
|---|---|---|---|
| A | AA | AB | AC |
| B | BA | BB | BC |
| C | CA | CB | CC |

Figure 3.6: Confusion Matrix

- **False negatives (*fn*)** which corresponds to the number of positive examples wrongly predicted as negative by the classification model. Considering the class A as an example, the value of the *fn* would be the sum of BA and CA.

- **False positives (*fp*)** which corresponds to the number of negative examples wrongly predicted as positive by the classification model. For example, with respect to the class A, the value of the *fp* would be the sum of AB and AC.

- **True negatives (*tn*)** which corresponds to the number of negative examples correctly predicted by the classification model. For class A the value of *tn* would be the sum of BB, BC, CB, and CC.

The **average accuracy** is the ratio of correct predictions to the total number of predictions, strictly speaking, how often is the classifier correct.

$$AverageAccuracy = \frac{\sum_{i=1}^{l} \frac{tp_i+tn_i}{tp_i+fn_i+fp_i+tn_i}}{l}$$

The **sensitivity**, or **recall**, is the ratio of correct positive predictions to the total number of positive predictions, in other words, that is, how sensitive the classifier is for detecting positive instances.

$$Recall = \frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l} (tp_i + fn_i)}$$

The **precision** is the proportion of positive identifications that were actually correct, how precise the classifier is when predicting positive instances.

$$Precision = \frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l} (tp_i + fp_i)}$$

The **F1 Score** is defined as the harmonic average between precision and recall.

$$F1\_Score = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)}$$

Where $tp_i$ are true positive, $fp_i$ are false positive, $fn_i$ are false negative and $tn_i$ are true negative counts for the class $C_i$, among $l$ classes.

#### 3.1.3.2 Cross Validation

Cross-validation [LJ09] is a commonly used technique to validate the performance of machine learning models, examining how well the model generalizes when dealing with unseen data.

Given that when the training data is reduced there is a risk of losing important patterns and trends, removing part of the data for validation can lead to a problem of underfitting. In order to avoid that, arises the cross-validation technique, which assures ample data for training the model and also leaves enough data for validation. In the K-Fold Cross Validation, the data is split into K number of subsets, known as folds. K-1 folds are used for training, with the remaining fold applied for the evaluation of the trained model. The final error is obtained by combining the errors coming from the K performed rounds, for example, using the average. In this method, K iterations are employed with a different subset reserved for the testing purposes each time.

Employing this method leads to more efficient use of data, as every observation is used for both training and testing.

#### 3.1.3.3 Nested Cross Validation

Hyperparameter optimization emerged to support the process of choosing the parameters which maximize the overall performance of the model. This is a costly process due to the plenty of options available for each parameter, in each model.

It is wrongly common to try achieving hyperparameter tuning using Cross Validation. However, since using the test set to both select the values of the hyperparameters and evaluate the model can result in optimistic and biased results, the right way to do the tuning is through Nested Cross-Validation (NCV).

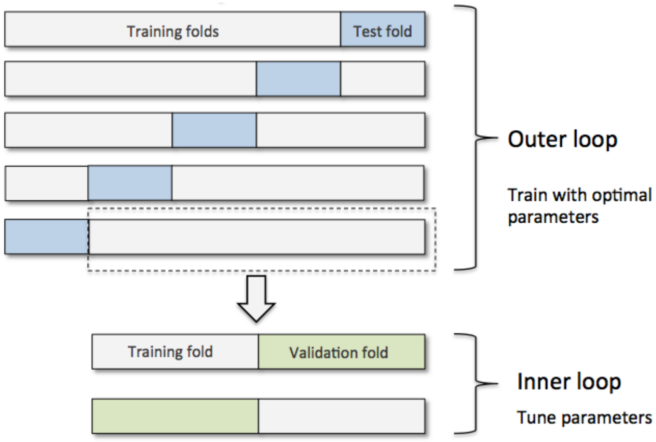


Figure 3.7: Nested Cross Validation Approach [Ras17]

As shown in Figure 3.7, at first, in NCV, an inner cross-validation loop is employed to tune the hyperparameters. Subsequent, an outer cross-validation loop is used to evaluate the model selected by the inner cross-validation.

## 3.2 Conclusion

In this chapter, key concepts in the field of Data Mining are gathered, in order to provide sufficient information to a clear understanding of the subsequent chapter, which bears on the tackled solution. The CRISP-DM methodology, usually applied to the development of data mining projects is explained, several supervised learning classification techniques that could potentially be employed in the project at hand are clarified, and existing model evaluation metrics are described.

# Chapter 4

# Implementation

This chapter describes the implemented solution and the required steps to tackle the problem at hand. This chapter is structured according to CRISP-DM methodology which, as explained in Section 3.1.1, can be broken down in six different phases named Business Understanding, Data Understanding, Data preparation, Modeling, Evaluation, and Deployment. In addition to the stated phases, another section was added at the beginning of the chapter, providing some guidelines about the conducted methodology.

## 4.1 Methodology

E-goi provides a Marketing Automation Platform solution for individual users, small or big companies. Users take advantage of this service in order to reach out their customers through automation marketing tools. Henceforth, the customers who use E-goi service will be referred to as *company or E-goi customers*, and the subsequent customers targeted by the email campaign will be mentioned as *recipients or customers*.

Aiming to classify a subject according to the respective quality, different strategies were carried out:

1. **Structure** analysis, under which only the subject structure was inspected.

2. **Content** analysis, focused on words that are part of the subject. In order to analyze the subject content, two different techniques were tested, i.e words past performance and Bag of Words (BOW).

The following sections describe the performed steps to achieve the proposed model, according to the CRISP-DM phases.

## 4.2 Business Understanding

With the growing number of companies adopting and considering email marketing as one of the most powerful and effective marketing channels to reach customers, and the consequent absurd and abundant number of emails received by each individual, each day, it has become impracticable for the recipient to pay attention to all the received content.

Given that the subject field acts as the deciding factor for a customer to open a specific campaign, this project intends to create a predictive model for classifying a subject field regarding its quality. Therefore, this study arose with the following objectives:

- To predict the subject quality for new email campaigns to be sent through the platform, taking into account not only structural but also subject content features which could impact the overall subject quality. Hence, by improving the subject quality, we seek to achieve higher open rates.

- To compare performance results regarding distinct data mining techniques, in specific Naive Bayes, Random Forest, Decision Trees, Support Vector Machines, Gradient Boosting, and Neural Networks.

- To increase E-goi customers engagement with the E-goi platform, through an innovative and helpful service capable of analyzing a subject-line quality.

This project is meant to work as a support tool for campaigns editors who need advice regarding the subject quality. Although endless website pages continue to teach users on how to improve subject-lines, they are merely based on a descriptive analysis. The need for a tool which evaluates the subject, instantly, is the reason why this project arose.

Unfortunately, factors such as time of the year, day of the week, customer mailing list and company's segment can seriously influence the outcome of this project. The subjectiveness when speaking about a good quality subject, since the same subject can be considered a good subject to an individual and a poor subject to another user, is an additional shortcoming which can contribute to the complexity and risk of this project.

Thanks to the data collected and available by E-goi, a data set with real customer information and real sent email campaigns was used in this project. As mentioned in Section 2.2.2, E-goi already offers a Content Check Tool which enables companies to improve the quality of their campaigns. This tool is intended to be added to the Content Check Tool as a new module named Subject analyzer, contributing to an even more complete and useful solution.

Finally, the study will be judged a success if E-goi customers start perceiving an increase in the OR of their email campaigns and, consequently, continue choosing E-goi in preference to the market competition.

## 4.3 Data Understanding

Making use of the data collected and available by E-goi, a new data set was created comprising email campaigns with, at least, 100 subscribers and, because campaigns take time to work and reach the recipients, only campaigns sent over one week before data collection were considered. This way, it is guaranteed that the data set includes exclusively campaigns with viable performance reports.

Due to the necessity of fast reading and high scalability and availability, it was chosen a NoSQL database system, in specific, an Apache Cassandra Database[1]. NoSQL databases are non-relational databases, capable of dealing with sheer volumes of data.

Table 4.1: Available data set

| **Variable** | **Description** | |
|---|---|---|
| campaign_hash | Campaign's unique and uniform identifier | - |
| client_id | Company's id | |
| country | Company's country | |
| sector | Company's business sector | |
| subject | Campaign's subject field | |
| unique_click_rate | No. of emails clicks for the first time by individual users | |
| unique_open_rate | No. of emails opened for the first time by individual users | |
| delivery_rate | No. of emails sent that were successfully delivered to recipients | |

The data set was composed of 140.000 sent email campaigns and, for each one, the information contained in Table 4.1 was provided. As introduced in Chapter 2, the CR depends directly on the campaign content and on the user's curiosity in knowing more about the content of the email. The DR depends on whether the email address is valid, the domain exists or the Internet Protocol (IP) address is being blocked. Therefore, from the listed available data, only the OR, the subject, the country, and the sector were picked to analyze the subject quality.

Concerning the diverse languages across the data set and, since it could be challenging to address efficiently so many different vocabularies and so many distinct countries, by developing a simple application, the most prevalent languages were filtered out. Employing the langdetect[2] python library to detect the subject language and, studying all the available data, we could sustain the original idea that the company's most significant marketplaces are Portugal, Brazil, Colombia, Spain and most of Latin America.

As shown in Figure 4.1, the most widespread languages among the subjects are Portuguese (pt), English (en) and Spanish (es). In fact, 80% of the subjects were written in Portuguese, 7% in English and 5% in Spanish. The other 8% corresponds to 28 different languages, including, for example, Dutch, French, Italian, and Catalan. Therefore, and as recommended by E-goi, only the three most relevant languages were considered to tackle this problem.

[1] http://cassandra.apache.org/

[2] https://pypi.org/project/langdetect/

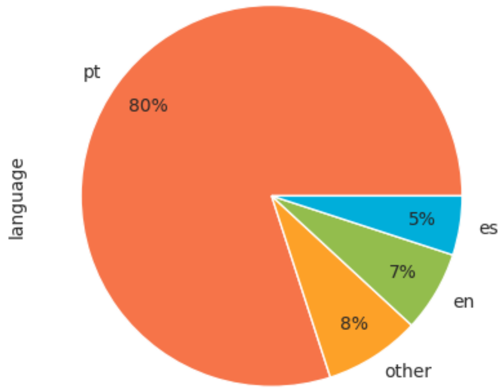


Figure 4.1: Most common email subject languages

### 4.3.1 Data Preparation

The Data Preparation phase encompasses feature engineering, as a way to extract knowledge from the data, creating features that make machine learning algorithms work.

The construction of the chosen classes and the construction of the new features are detailed in the following sections. Furthermore, concretely in Section 4.3.1.6, the data issues such as outliers and the lack of attributes of interest are identified.

#### 4.3.1.1 Classes creation

Considering the continuous value for the OR and the classification problem at hand, the first step was to convert the OR into a discrete value. Therefore, a binning algorithm (equal-width binning) was employed to establish 5 levels of quality, each one with a balanced number of samples.

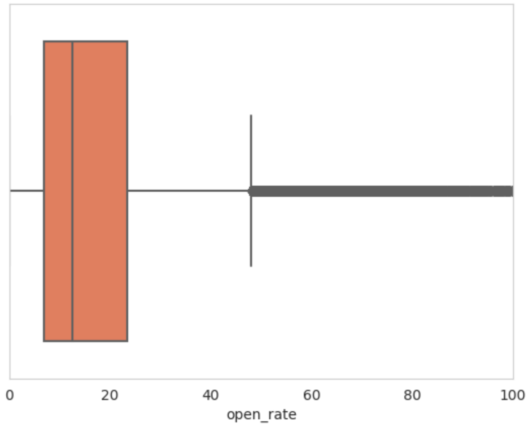


Figure 4.2: Open Rate Distribution

Figure 4.21 represents the distribution of the OR variable.

$$quality\_class = \begin{cases} if & OR \geq 22.4 & ,5 \\ if & 22.4 > OR \geq 13.7 & ,4 \\ if & 13.7 > OR \geq 9.37 & ,3 \\ if & 9.37 > OR \geq 5.41 & ,2 \\ else, & 1 \end{cases} \tag{4.1}$$

Equation 4.1 lists the five classes which were created through the binning technique, representing a 5 stars rating system.

#### 4.3.1.2 Structural Features

The structural features assembler cares about the structure and size of the subject-line, considering the number of words, number of characters, the use of upper case, punctuation, special characters, numbers currency, emojis and the personalization of the message.

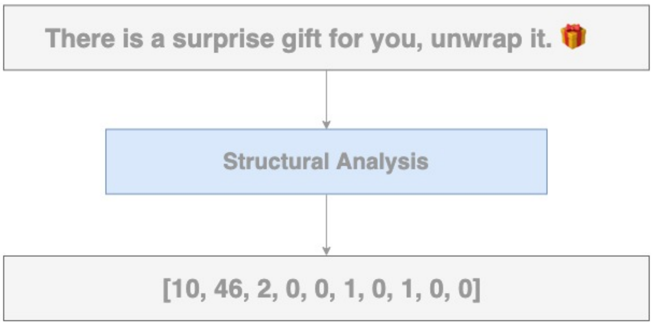


Figure 4.3: Structural Analysis

As shown in Figure 4.3, the structural assembler receives a subject and outputs a list with:

*[number of words **(integer)**, number of characters **(integer)**, case percentage **(integer)**, presence of punctuaction**(boolean)**, presence of prefixes **(boolean)**, presence of emojis **(boolean)**, presence of personalization **(boolean)**, presence of special characters **(boolean)**, presence of numbers **(boolean)**, presence of currency **(boolean)**]*

Please note that emojis are treated as words, as they represent a set of characters. Personalizing the subject-line means adding subscriber unique information such as the name or email address, providing messages perfectly tailored to subscribers' needs. The E-goi platform allows addressing the subscribers by their first name, e.g. *"Hi John"* or by any other information.

Table 4.2 contains some of the codes that should be used in order to achieve personalization.

Regarding the currency analysis, the structural assembler is capable of identifying currency by the symbol (€, $, £, R$) or by the abbreviation, for euro, real, dollar or pound sterling. With respect to the case percentage, it gives a percentage of the subject in upper case. For example, the word "John" has 25% of upper case percentage.

Table 4.2: E-goi personalization codes

| Personalization Code | Meaning |
|---|---|
| !fname | Full Name |
| !lname | Last name |
| !email | Email Address |
| !telephone | Phone Number |
| !birth_date | Birth date |

The prefixes are abbreviations which are added by the author to the subject-line, in order to improve the ability to prioritize, review, and process new inbox messages.

Table 4.3: List of common prefixes

| Prefix | Meaning |
|---|---|
| RE | Followed by the subject of a previous message indicates a "reply" to that message |
| FWD | A forwarded message |
| FW | A forwarded message |
| WAS | The subject was changed |
| FYI | Meaning For your information |
| NRN | Meaning No Reply Necessary |
| OT | Off topic |
| EOM | End of message |
| 1L | One Liner, when the subject is the only text contained in the email |
| NONB | Non-business |
| ASAP | Meaning As Soon As Possible |

Table 4.3 contains some of the most helpful acronyms and the respective meaning.

#### 4.3.1.3 Content Features

The content features assembler create features with information regarding the quality of the set of words contained in the subject. In order to achieve this, two distinct approaches were employed: Past Performance and BOW.

Before explaining the abovementioned approaches, and considering that the preprocessing phase was transversal across both strategies, there is the sequence of operations taken to perform the cleaning and subsequently subject analysis:

- **Clean Subject:** Converts a subject to lower case, removes personalization codes, mentioned in Table 4.2, removes punctuation marks, special characters and emojis, keeping the accent marks, also known as Diacritic[3].
- **Identify language:** The subject language was identified through the referred Python Library called langdetect[4]. Accordingly to the aforementioned research results on finding the most

[3] Diacritic is a glyph added to a letter, or basic glyph.

[4] https://pypi.org/project/langdetect/

common languages across the past subject-lines, only the Portuguese, English and Spanish packages were made available and, hence, all the other languages were discarded. In case of being impossible to identify the language, the respective country language was used. Taking into account the available packages (English, Portuguese and Spanish), the following countries were accepted: Portugal, Honduras, Angola, Curaçao, Spain, Brazil, Colombia, Peru, Mozambique, United States Of America, Australia, Mexico, Malta, United Kingdom, Argentina, and Chile.

- **Discard Stop Words:** Depending on the language, different stop words were discarded. To identify them, the Spacy [5] library was used. Filtering out stop words before or after the processing of natural language data is a very common strategy to save time and space in these type of projects. As stop words correspond to common and irrelevant words of the vocabulary [6], they can be ignored.

- **Lemmatizing:** Depending on the detected language, different lemmatizers were applied. Concerning lemmatization, the Spacy[7] library was used. A lemma is the canonical form or dictionary form of a set of words[8].

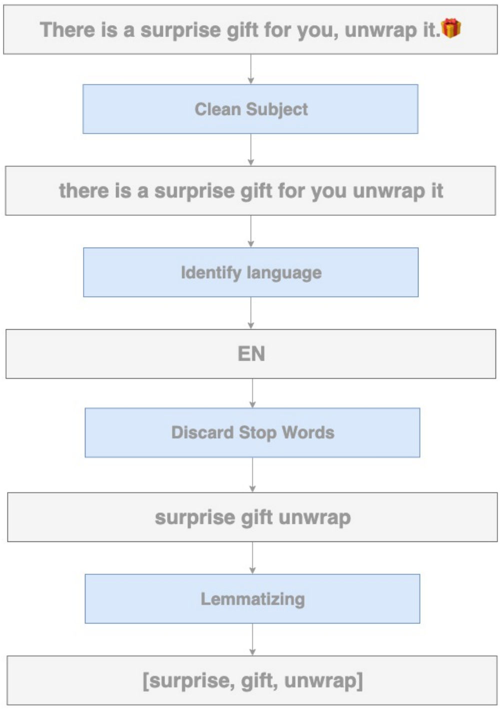


Figure 4.4: Example of subject pre-processing

Figure 4.4 illustrates the pre-processing steps for a random subject, explained above.

[5] https://spacy.io/

[6] For example, the word "the" and "a".

[7] https://spacy.io/

[8] For example, the words running, runs and ran all have run as the lemma.

#### 4.3.1.4 Past Performance Approach

Two new variables were created containing information regarding the subject's lemmas past performance. Considering a lemma as the dictionary form of a set of words, the first feature, *lemmas_past_performance*, represents the quality of the lemmas contained in the subject, predicated on the past sent subjects, and contains values between 1 and 5. The second feature, called *number_lemmas*, as the name suggests, holds information regarding the number of identified lemmas.

In order to do so, a dictionary was created, in which, for each lemma, the following information was available: *[lemma, count, average of quality from all past sent subjects]*.

**Dictionary creation**

Hence, towards generating those new variables, a dictionary was created containing past sent lemmas, the respective number of appearances and the average of the quality calculated through the unique open rate.

```

'''Pre Processing'''
cleaned_subject = clean_subject(subject)
lang_code = identify_language(subject)
transformed_subject = remove_stop_words(lang_code, cleaned_subject)
lemmas = lemmatizing(lang_code, transformed_subject)

'''Updating dictionary'''

for lemma in lemmas:

    '''Lemma is already contained in the dictionary -> updates quality'''
    if lemma in dictionary:
        update_count(lemma)
        update_average_quality(lemma)
    else:
        add_lemma_to_dict(lemma)
```

Listing 4.1: Dictionary creation

The Listing 4.1 synthesizes the applied steps for each sent subject in order to fill out the dictionary. Basically, the steps are the same as to do the subject preprocessing, explained in Section 4.3.1.3.

**Features Builder**

Subsequently to the dictionary creation, the *lemmas_past_performance* variable was added. In order to do so, different approaches were experimented:

1. At first, the **weighted average** was employed. Hence, it was calculated with the past quality of each of the lemmas contained in the subject and also with the number of appearances

of each of these lemmas in emails history. The information regarding the number of appearances of each lemma is contained in the dictionary and here acts as the weight. This approach guarantees that, for example, if a word was used only once, it will impact differently compared to frequently used words.

$$weighted\_average = \frac{\sum_{i=1}^{n}(x_i * w_i)}{\sum_{i=1}^{n} w_i} \tag{4.2}$$

For $n$ lemmas filtered from the subject, the weighted average is the quality of the lemma, $x_i$, multiplied by the weight of the lemma $w_i$, here known as the number of appearances of the lemma, divided by the sum of all the $n$ lemmas weights.

2. Secondly, the naive **average** was employed, discarding the number of appearances of each of the lemmas.

$$average = \frac{1}{n} * \sum_{i=1}^{n} x_i \tag{4.3}$$

For $n$ lemmas filtered from the subject, the average is the sum of the quality of each lemma $w_i$, divided by the number of lemmas.

3. Faced with the email overload, when opening the mailbox, recipients choose to ignore some email subjects or not to read them fully [SL16]. For this reason, subject-lines need to quickly grab the recipient's attention.

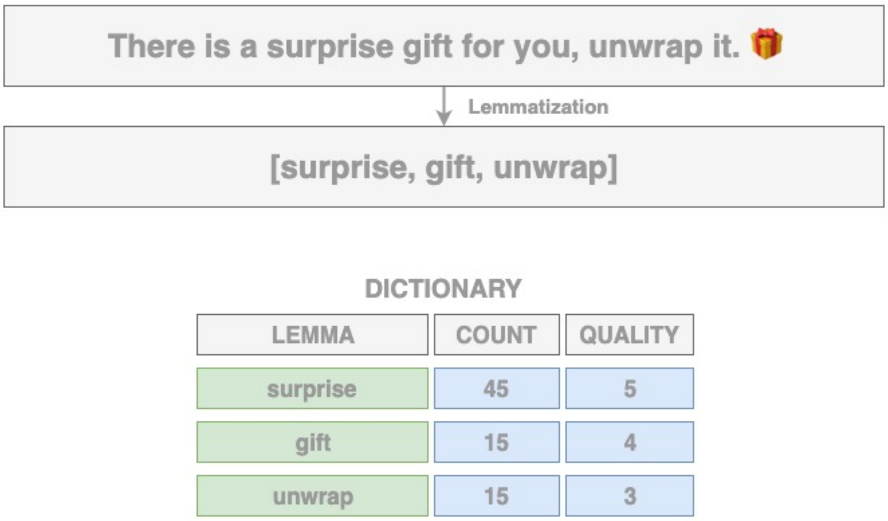
According to Duggan et al. [DP06], "skimming" a text means reading a text quickly to get a general idea of meaning and is increasingly common in our information-rich time-limited society. The author concluded that readers focus on important information when skimming.

Hence, given the small fraction of time spent in analyzing the subject-line, considering that a customer perceives only one word of the subject instead of all words, a different approach was tested, but this time considering just the **word that stands out**.

Thus, the value of the variable *lemmas_past_performance* was calculated by the maximum quality value of the lemmas. Accordingly, if a subject contains a lemma of 5 stars quality, it is believed that the subject semantic quality is also 5 because the reader, subconsciously, ignores all the other lemmas.

Table 4.4: Example of content analysis for each different approach

| Approach | lemmas_past_performance |
|---|---|
| Weighted Average | $\frac{(45*5)+(15*4)+(15*3)}{45+15+15} = 4.4$ |
| Average | $\frac{(5+3+4)}{3} = 4$ |
| Maximum Value | $Max(5,3,4) = 5$ |



There is a surprise gift for you, unwrap it. 🎁

[surprise, gift, unwrap]

DICTIONARY

| LEMMA | COUNT | QUALITY |
|---|---|---|
| surprise | 45 | 5 |
| gift | 15 | 4 |
| unwrap | 15 | 3 |

Figure 4.5: Example of subject content analysis

Figure 4.5 is an example of the content analysis for a random subject. Table 4.4 illustrates how is the *lemmas_past_performance* calculated regarding each of the mentioned approaches.

#### 4.3.1.5 Bag of Words Approach

In substitution to the Past Performance strategy and because of being commonly used in natural language processing and document classification [FSGS14], a BOW approach was attempted. The BOW technique extracts features from text documents, that can be used in machine learning algorithms. Essentially, is responsible for creating a vocabulary of unique words contained in all documents, completely disregarding the order in which those words appear.

After pre-processing the subject, the BOW method can be achieved through the following steps:

- **Build Vocabulary:** Taking as an example the subject in Figure 4.4, after cleaning, removing the stop words and stemming, the generated vocabulary would be *[surprise, gift, unwrap].*

- **Generate Vectors:** Following the vocabulary creation, as a new input appears, the count vectors are generated. Considering the following new subject *"I have a gift for you"* and the vocabulary *[surprise, gift, unwrap]*, the generated vector would be *[0, 1, 0]*, since only the word "gift" is contained in the vocabulary.

  In this particular case, instead of the simple count of the words within the subject, and because it is essential to take in consideration all the other sent subjects, the Term Frequency-Inverse Document Frequency (TF-IDF) calculation was employed. TF-IDF measures relevance, not frequency. For that reason, word counts are replaced with TF-IDF scores across the whole data set. TF-IDF not only measures the number of times a word appears in a subject but then, it gives more importance to less common words and a lot of value to the rare ones [Abu17].

Although being very simple to understand, in most cases, the BOW technique reveals to be insufficient and limiting because of some shortcomings. Considering that each word of the vocabulary acts as a new feature of the model, this technique requires a big space and time complexity. For that reason, it can be challenging to guarantee sufficient information for the model, in this specific case, enough vocabulary given the space complexity required to ensure that. Additionally, taking account that not only one but multiple languages are being studied, it could be really hard to provide sufficient vocabulary for each of the languages due to space constraints.

#### 4.3.1.6 Data quality

During the data preparation phase, it is also crucial to clean the data. The primary focus of data cleaning phase is to handle missing data, noisy data, and remove outliers, minimizing duplication and computed biases within the data.

While analyzing the data, inconsistencies were detected. Beginning with missing values, many samples were found to have the sector column without any value. Those cases were related to users who do not specify any business sector in their E-goi profile account. In order to fix that issue, samples with empty sector column were filled with the *undefined* value.

Moreover, when inspecting the *lemmas past performance* and *number of lemmas*, some subjects were found to be assigned with zero number of lemmas. Doubting of the existence of subjects with zero lemmas, after some research, they were found to be related to subjects with miswritten words, extremely short words or words with more than 16 characters. Since those subjects were creating noise in data and do not help in explaining the feature itself or the relationship between feature and target, they were discarded.

Lastly, outliers were found regarding the unique open rate value. Some data points differed significantly from other observations, containing fraudulent open rates, with values greater than 100% and, hence, were dropped.

## 4.4 Data Exploration

In this section, deep data analysis is done using data visualization techniques whereby it is possible to characterize data, find correlations, identify potential relationships, patterns or insights that may be hidden in the data. For data visualization, the Seaborn[9] and the Matplotlib[10] python libraries were used.

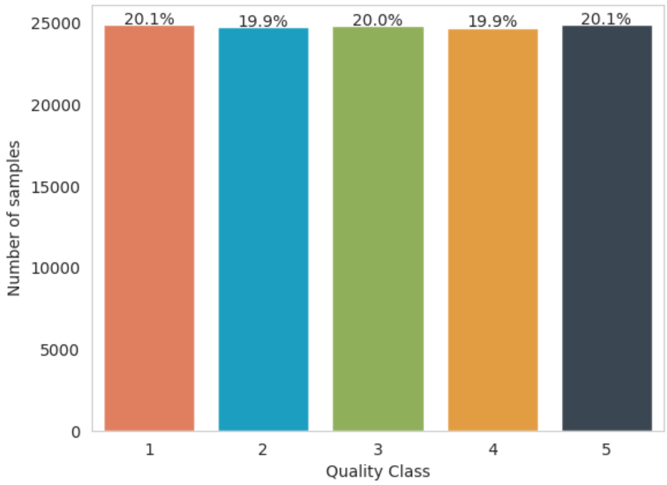


Figure 4.6: Samples distribution per class

Figure 4.6 illustrates the samples distribution per class, from 1 to 5 stars, with 1 being the lower quality. We can infer that there exists a balanced share of subjects in each quality class.

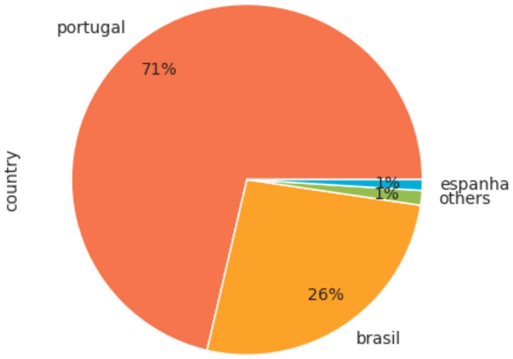


Figure 4.7: Data set country distribution

Figure 4.7 reveals how unbalanced is the distribution of countries in the data set. Portugal and Brazil are the countries sending more email campaigns, followed by Spain.

[9] https://seaborn.pydata.org

[10] https://matplotlib.org

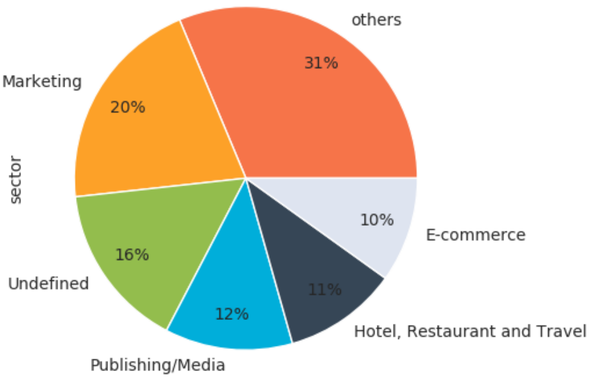


Figure 4.8: Data set sector distribution

Figure 4.8 illustrates the sector distribution in the data set. This distribution seems most balanced, with 20% of the email campaigns sent from customers with the business sector being *Marketing*, 16% without any selected sector, tagged with *Undefined*, and the big slice of email campaigns being associated to *others* business sector.

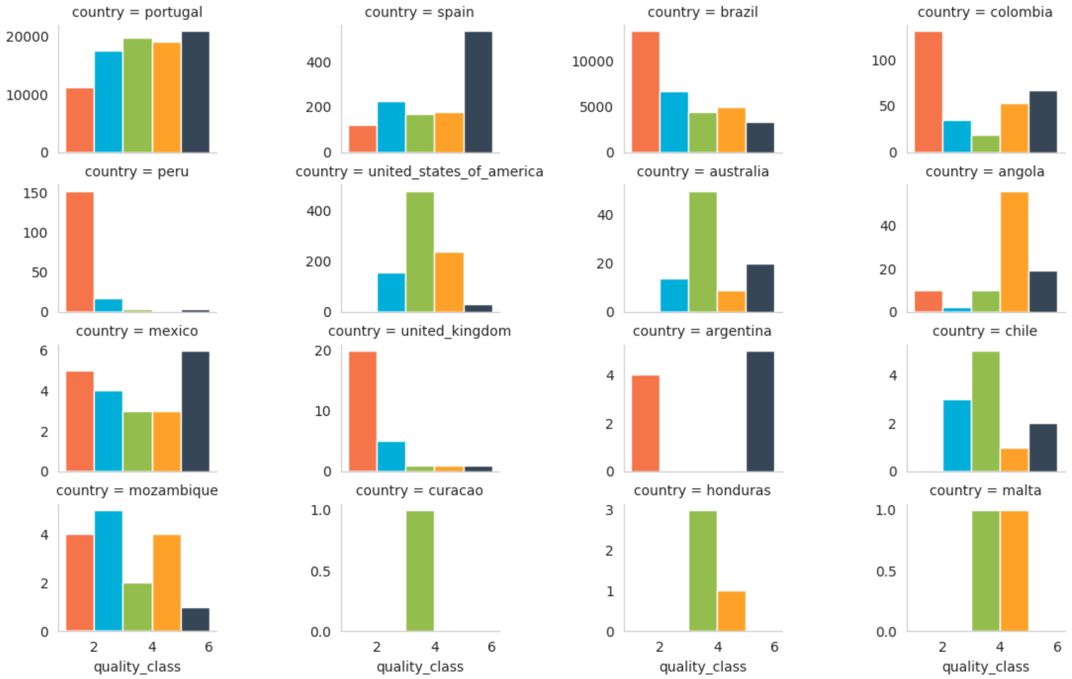


Figure 4.9: Subject-line quality per country

Figure 4.9 reveals how good is the open rate regarding the country. It is reasonable to conclude that Portugal, Spain, Argentina, and Mexico send campaigns with high-quality subjects. On the other side, even though Brazil is still the country which sends more campaigns, very few campaigns were sent with good subjects, leading to low OR.

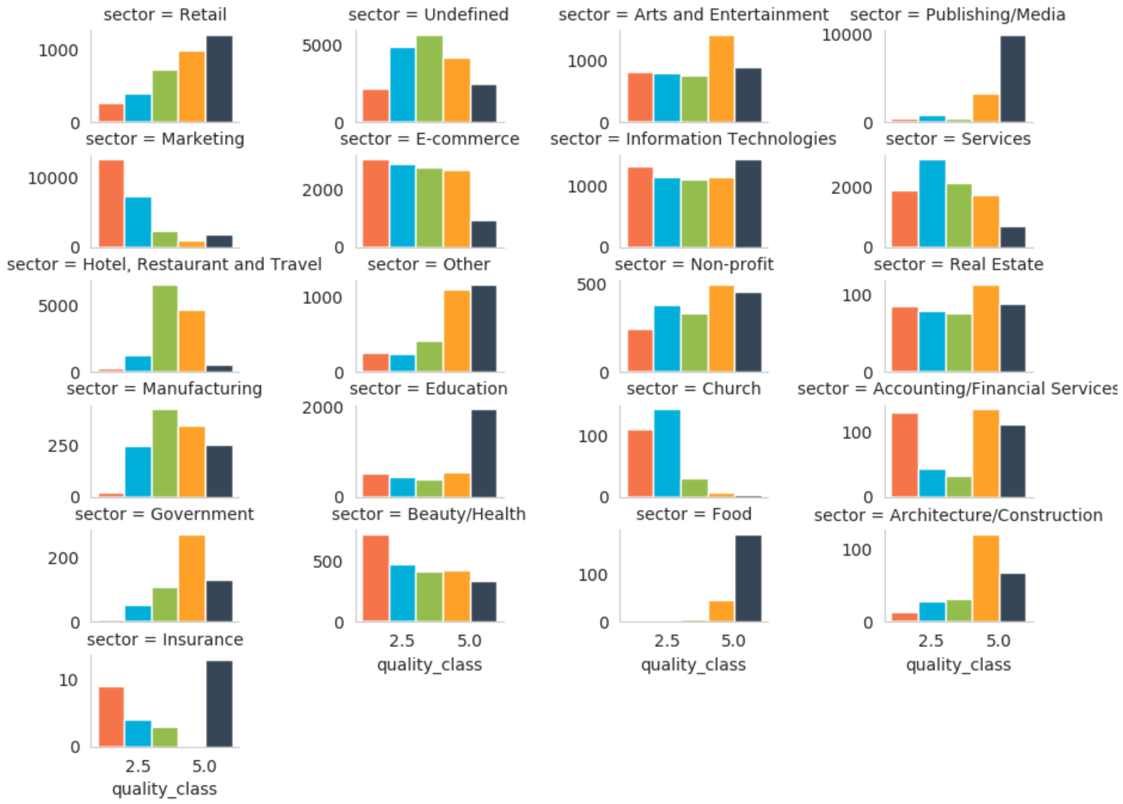


Figure 4.10: Subject-line quality per sector

Figure 4.10 reveals how good are the campaigns' subject regarding the sector. Indeed, for Education, Publishing/Media, Food, Insurance, and Retail sector, the likelihood of a subject being of high quality is high. On the other hand, *Marketing*, *Hotel, Restaurant and Travel*, and *Church* tend to have subjects with low quality.

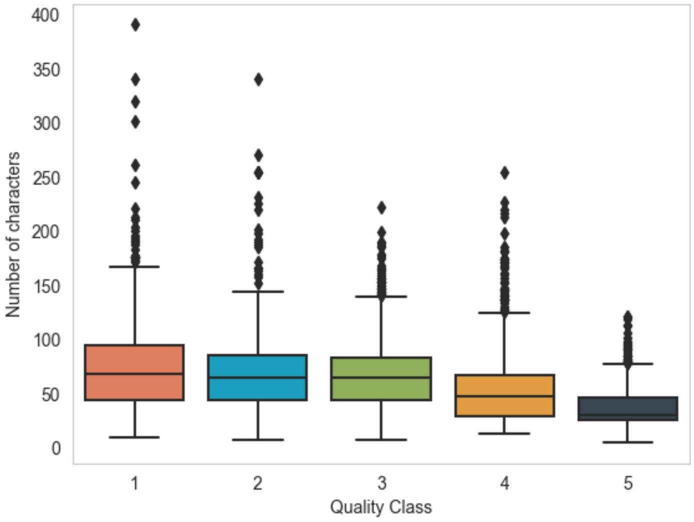


Figure 4.11: How the number of characters affect the subject quality

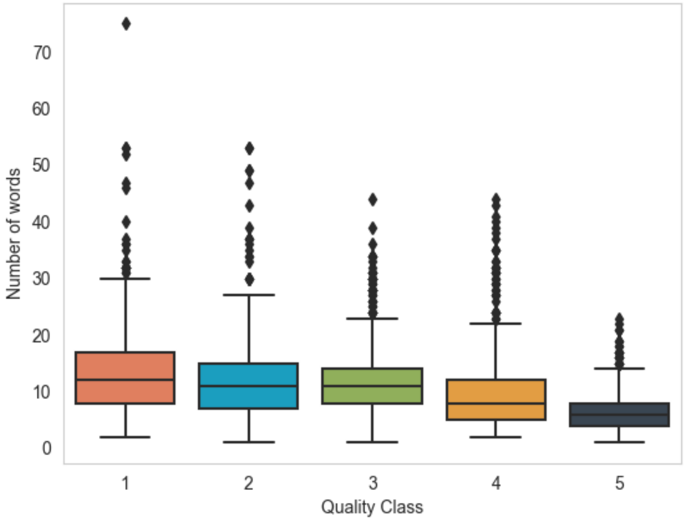


Figure 4.12: How the number of words affect the subject quality

Figure 4.11 summarizes the correlation between the subject quality and the number of characters, and Figure 4.12 reveals how is the subject quality affected by the number of words. In fact, we can conclude that the median (represented by the line that divides the box into two parts) for the number of words in class 5 is extremely low, increasing as the quality of the subject decreases, behaving similarly to the number of characters. Therefore, it is plausible to declare that the shorter the subject, the better the quality.

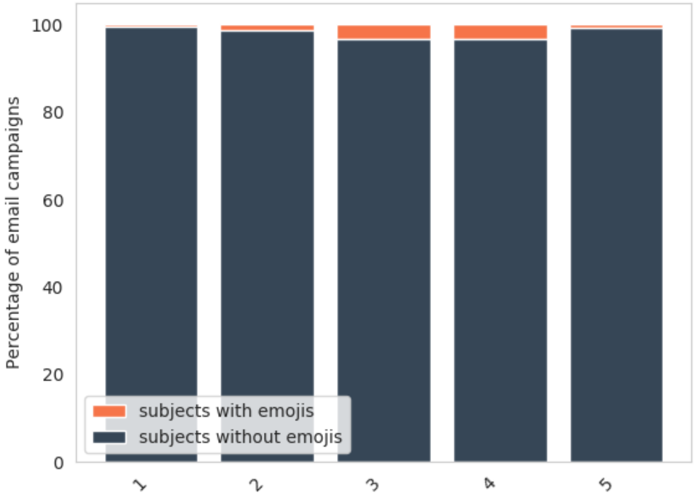


Figure 4.13: How the use of emojis affects the subject quality

Figure 4.13 clarifies how the use of emojis affects the subject quality. Surprisingly, adding emojis to subject-lines is not an obvious advantage and, in fact, for subjects with quality higher than 3, the use of emojis does not seem to be worth.

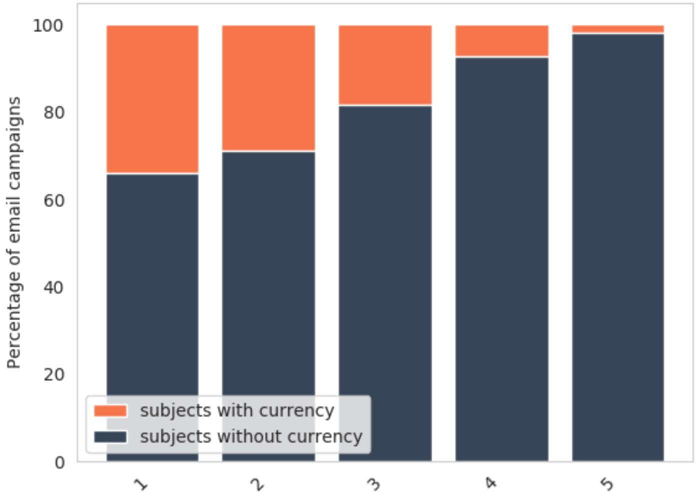


Figure 4.14: How the use of currency affects the subject quality

Figure 4.14 demonstrates how the use of currency affects the subject quality and, contrarily to what people may think, data uncovers that there is no benefit in adding currency symbols. As the quality of the subjects increase, the number of subjects with currency abbreviations or symbols decreases.

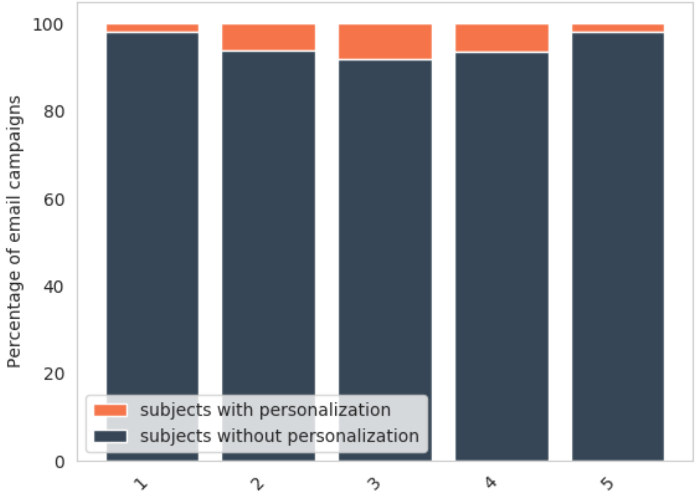


Figure 4.15: How the use of personalization affects the subject quality

As shown in Figure 4.15, similar happens to the use of personalization. Although people may believe personalizing subjects increase the curiosity and interest in opening the email, the data reveals that personalize subjects helps improving subject quality only till quality 3 stars. For subjects with quality higher than 3 stars, the use of personalization is irrelevant.

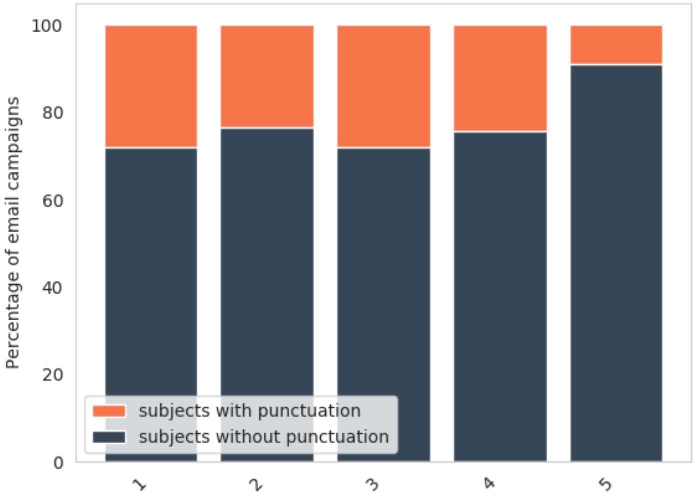


Figure 4.16: How the use of punctuation affects the subject quality

As shown in Figure 4.16, the use of punctuation does not influence linearly the subject quality. Anyway, it is plausible to conclude that there is a small number of subjects containing punctuation characters in the 5 stars quality class.

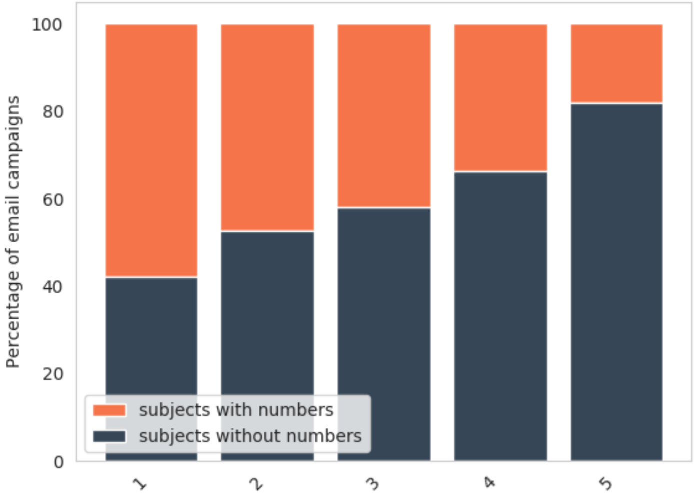


Figure 4.17: How the use of numbers affects the subject quality

With regard to the use of numbers and special characters, as shown in Figure 4.17 and Figure 4.18, as the quality of the subject increases, the number of samples with numbers and special

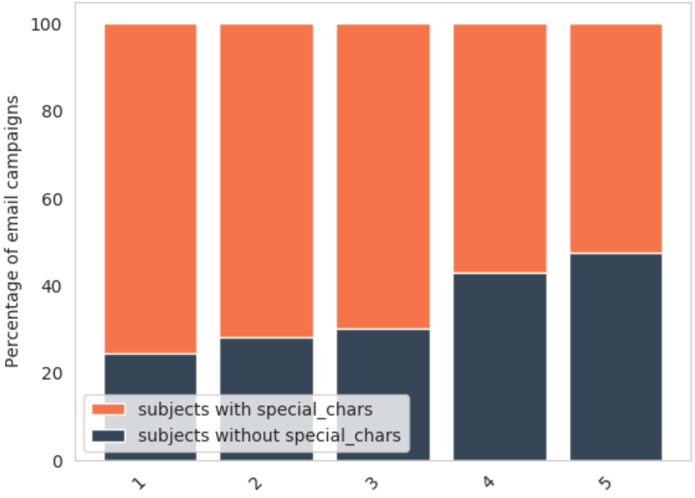


Figure 4.18: How the use of special characters affects the subject quality

characters decreases. Consequently, neither the presence of numbers or special characters increase the subject quality.

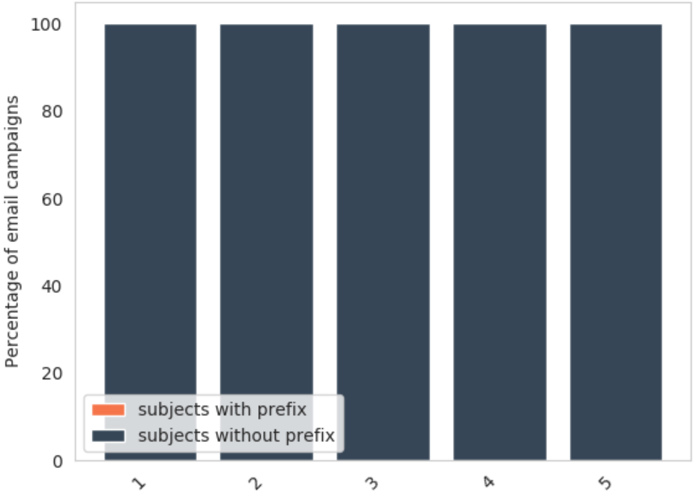


Figure 4.19: How the use of prefixes affects the subject quality

As Figure 4.19 reveals, the number of subjects containing a prefix is so insignificant that it is impossible to analyze the particular impact on the subject quality.

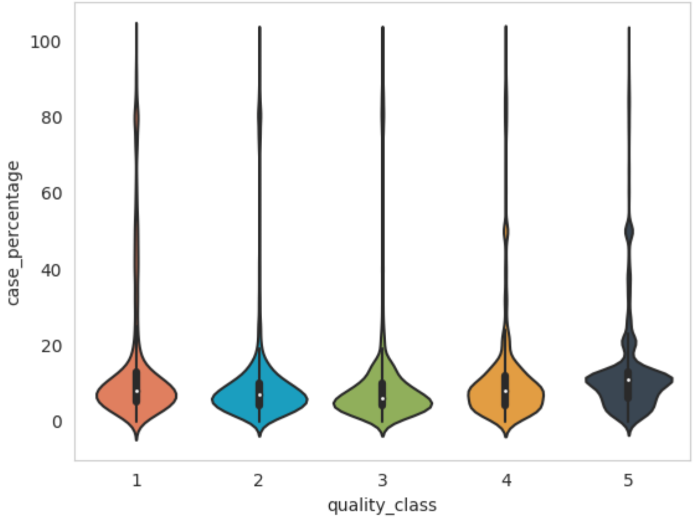


Figure 4.20: How the percentage of upper case affects the subject quality

Looking at Figure 4.20 and regarding the case percentage impact on the subject quality, there exists an insignificant difference between the median for each of the classes, concluding that the presence of characters in upper case does not have any good or bad impact on the overall subject quality.

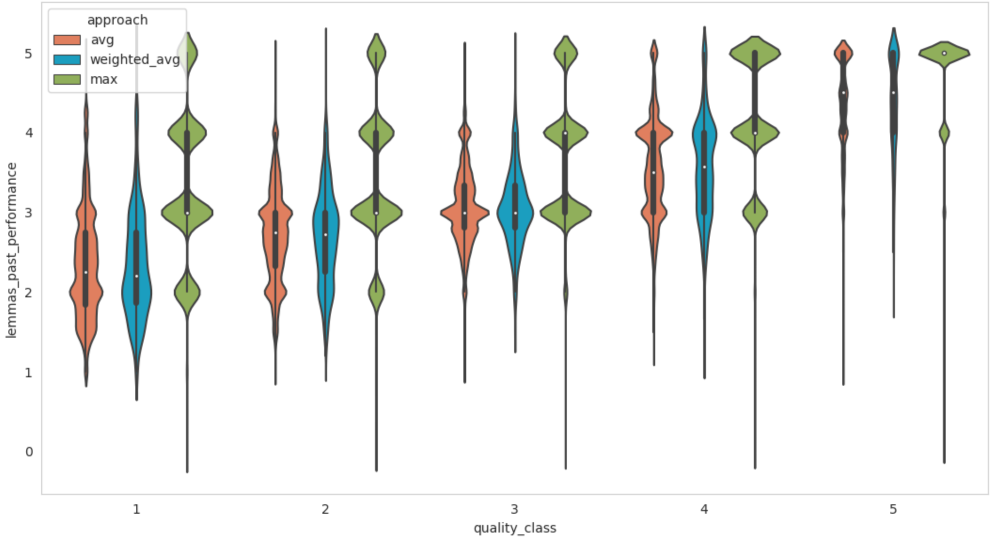


Figure 4.21: Comparing different approaches for lemmas past performance calculation

Reminding the distinct possible ways to calculate the *lemmas_past_performance* feature, explained in Section 4.3.1.4, Figure 4.21 represents how strongly is the quality class influenced by each of the different taken approaches. The average and weighted average approaches seem to be a good indicator of the subject quality contrary to the maximum approach, which does not act as a good indicator of the subject quality.

## 4.5 Data Preparation

The data type of an attribute *(nominal or categorical - numerical or ordinal)* affects the way algorithms identify patterns and relationships between attributes.

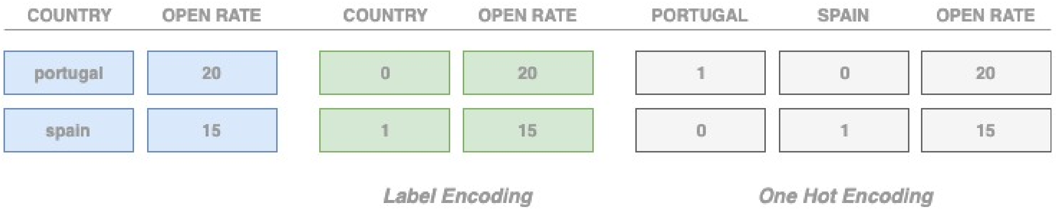
With the creation of new numerical features such as the number of words, number of characters, case percentage, lemmas past performance and number of lemmas, scaling problems arose. Since feature scaling can affect the performance of many data mining algorithms, data normalization was performed in order to bring all the numerical features to the same level of magnitudes. **Min-Max** normalization is one of the most common ways to normalize numerical data [SM13]. Therefore, for each feature, the minimum value gets transformed into a 0 and the maximum value gets transformed into a 1. All the other values get a value between 0 and 1, given by the following equation:

$$MiniMax = \frac{value - min}{max - min} \tag{4.4}$$

Dealing with numeric features is often easier than dealing with categorical features (ordinal or nominal) since it does not make sense to apply arithmetic analysis to nominal and ordinal values. For that reason, it is imperative to transform categorical values into numeric labels and then apply some encoding scheme on these values. The encoding methods that could have been used to transform both the country and sector nominal features into numeric features are:

- **Label Encoder**, as shown in Figure 4.22, create labels with a value between 0 and *n_classes*-1 where n is the number of distinct classes. The Label Encoder is not a good option when there are more than two classes to transform since it will associate a natural unexistent order to each of the different classes.

- **One Hot Encoder**, as shown in Figure 4.22, performs "*binarization*" to each of the categorical features. Hence, for each categorical value, a new column is created, with a boolean value.



| COUNTRY | OPEN RATE | COUNTRY | OPEN RATE | PORTUGAL | SPAIN | OPEN RATE |
|---|---|---|---|---|---|---|
| portugal | 20 | 0 | 20 | 1 | 0 | 20 |
| spain | 15 | 1 | 15 | 0 | 1 | 15 |

*Label Encoding*
*One Hot Encoding*

Figure 4.22: Encoding Methods

Because the model could misunderstand the data to be in some kind of order, the One Hot Encoder was employed. Hence, all categorical variables were converted into a form that could be provided to a model without compromising the overall performance.

## 4.6 Modeling

Data Mining algorithms are biased to look for different types of patterns, and because there is no learning bias across all situations, neither algorithm can be considered the best [Joh]. Each algorithm requires different parameters to be set. Choosing the appropriate hyperparameters plays a crucial role in the success of the model and, therefore, the parameters need to be tuned in order to achieve the optimal model, the one which minimizes the error. Thus, the proposed methodology involved parameters tuning.

Furthermore, since depending on the selected features, more or less accurate models can be generated, different experiments were performed with different methods and features. Recapping the different applied methodologies:

- **Experiment 1 - Structural Analysis:** Only the subject structure, the sector, and the country were considered.
- **Experiment 2 - Structural and Content Analysis:** Additionally to the structure and company information (country and sector), also the content was considered.
  - **Experiment 2.1:** *Lemmas Past Performance* feature was calculated through the Weighted Average.
  - **Experiment 2.2:** *Lemmas Past Performance* feature was calculated through the Average.
  - **Experiment 2.3:** *Lemmas Past Performance* feature was calculated through the Maximum quality.
- **Experiment 3 - Structural and Content Analysis with Bag Of Words:** BOW with a maximum of 4% of the vocabulary, which includes the words contained in all past subjects, followed by dimensionality-reduction. In order to do that, Principal Component Analysis (PCA) was applied. PCA [LJ09] is a mathematical procedure for dimensionality reduction in data, which transforms a number of correlated variables into a smaller number of uncorrelated variables called principal components.

Due to the variety of experiments, each one with distinct features and, therefore, different transformers to apply, a Pipeline[11] was employed.

As shown in Figure 4.23, Pipelines make it easier for experiments by streamlining a lot of the routine processes, encapsulating little pieces of logic into one function call, thus simplifying the modeling phase. Instead of just writing a bunch of code, pipelines sequentially apply a list of transforms and a final estimator.

For illustration purposes, Listing 4.2 represents the Pipeline applied on Experience 2. Additionally to the data preprocessing, the final model can also be added to the pipeline. Therefore, training the model can be done simply and the preprocessing can be adjusted easily.

[11] https://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html

```

’’’ Returns a pipeline encapsulating all the preprocessing steps.’’’

def get_pipeline():

    categorical_transformer = Pipeline(steps=[(’hotencoder’, OneHotEncoder
        (handle_unknown=’ignore’, sparse=False))])

    numerical_transformer = Pipeline(steps=[(’minmax’, MinMaxScaler(
        feature_range=(0, 1), copy=False))])

    transformer = ColumnTransformer(transformers=[(’categorical’,
        categorical_transformer, [’country’, ’sector’]), (’numerical’,
        numerical_transformer, [’number_words’, ’number_chars’, ’
        case_percentage’, ’lemmas_past_performance’, ’number_lemmas’])],
    remainder=’passthrough’)

    pipeline = Pipeline(steps=[(’preprocessing’, transformer)])

return pipeline
```

Listing 4.2: Example of Pipeline: Experience 2

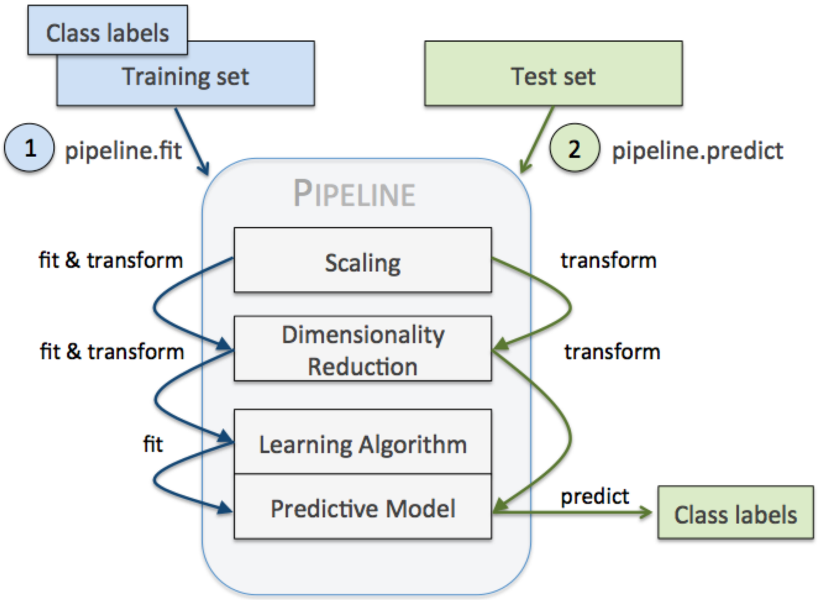


Figure 4.23: Pipelines in scikit learn [Ras17]

Table 4.5 summarizes all the generated features and the respective context of creation. A detailed description of the attributes used in this study is presented in Table A.1, in Appendix A.

For each mentioned methodology, a set of modeling techniques were applied, and their parameters were calibrated to optimal values. Given the classification problem at hand, six different modeling algorithms were studied:

Table 4.5: Available Features

| Feature | Type | No. of categories | Context | Exp. 1 | Exp. 2.1 | Exp. 2.2 | Exp. 2.3 | Exp. 3 |
|---|---|---|---|---|---|---|---|---|
| Country | Nominal | 13 | Company's Information | X | X | X | X | X |
| Sector | Nominal | 21 | Company's Information | X | X | X | X | X |
| Number of words | Continuous | - | Structure | X | X | X | X | X |
| Number of characters | Continuous | - | Structure | X | X | X | X | X |
| Case Percentage | Continuous | - | Structure | X | X | X | X | X |
| Punctuation | Binary | - | Structure | X | X | X | X | X |
| Prefix | Binary | - | Structure | X | X | X | X | X |
| Emojis | Binary | - | Structure | X | X | X | X | X |
| Personalization | Binary | - | Structure | X | X | X | X | X |
| Special_chars | Binary | - | Structure | X | X | X | X | X |
| Numbers | Binary | - | Structure | X | X | X | X | X |
| Currency | Binary | - | Structure | X | X | X | X | X |
| Lemmas Past Performance (AVG) | Continuous | - | Content: Past Performance | | X | | | |
| Lemmas Past Performance (W_AVG) | Continuous | - | Content: Past Performance | | | X | | |
| Lemmas Past Performance (MAX) | Continuous | - | Content: Past Performance | | | | X | |
| Number of lemmas | Numerical | - | Content: Past Performance | | X | X | X | |
| List of TF-IDF* | Numerical | - | Content: Bag Of Words | | | | | X |
| ** For each subject, the BOW approach produces a vector of tf-idf frequencies* | | | | | | | | |

- C-Support Vector Classifier
- Gaussian Naive Bayes
- Multi-layer Perceptron Classifier
- Decision Tree Classifier
- Random Forest Classifier
- Adaptive Boosting

Following the selection of the algorithms, two essential steps were employed: hyperparameters tuning and, subsequently, model selection. Since tunning the parameters and generalization error estimation in the same data could lead to overfitting, cross-validation revealed insufficient. To address this problem, nested cross-validation, each one using 10 Folds, was employed.

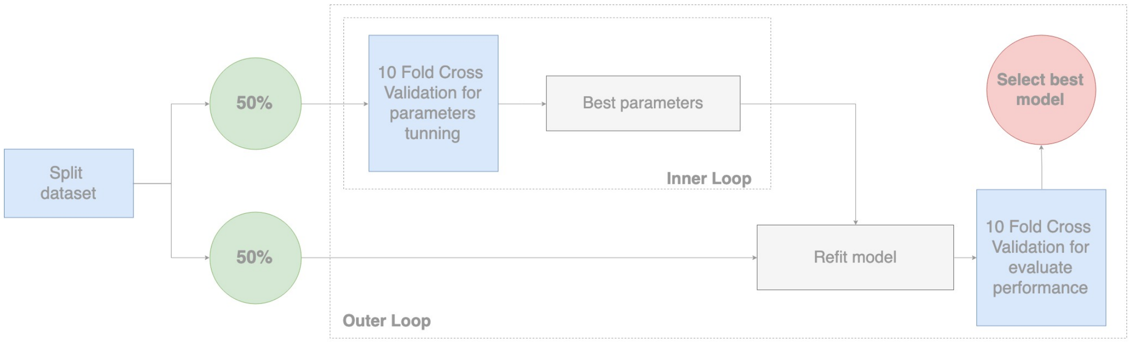

Split dataset
50%
50%
10 Fold Cross Validation for parameters tunning
Best parameters
Inner Loop
Select best model
Refit model
10 Fold Cross Validation for evaluate performance
Outer Loop

Figure 4.24: Nested cross Validation

Figure 4.24 illustrates the carried out process. The data was split into two equal parts using stratification, providing stratified randomized folds in order to guarantee the same percentage of samples for each class. With 50% of the data, the hyperparameters tunning. With the remaining 50% of the data, the error was estimated and the overall performance was analyzed.

```
def evaluate():

    labels, features = load_data()
    pipeline = get_pipeline()
    ''' Preprocess and split data'''
    transformed_features = pipeline.fit_transform(features)
    features_train, features_test, labels_train, labels_test =
        train_test_split(transformed_features, label, test_size=0.5,
        random_state=42, stratify=labels)

    ''' Grid search with the tunned models'''
    grid_searches = nested_cross_validation(features_train, labels_train)

    best_estimator_index = evaluate_performance(grid_searches,
        features_test, labels_test)

    '''Append best model to pipeline'''
    pipeline.steps.append(['model', grid_searches[best_estimator_index
        ][1]])
    pipeline.fit(features, label)
    save_model(pipeline)
```

Listing 4.3: Example of Pipeline: Experience 2

Listing 4.3 describes the steps required to evaluate the model.

### 4.6.1 Hyperparameters Tunning

Hyperparameters are parameters whose value is set before training a specific model. Depending on the chosen parameters, the model learns the data patterns differently. In order to achieve the parameters that better learn the data patterns, hyperparameter tuning was employed.

The hyperparameters tuning was achieved through Grid Search[12] module, available in Sckit-Learn. Grid Search is an exhaustive search over given parameters values for an estimator. It trains the algorithm for all combinations of parameters and measures the performance using the cross-validation technique. In this particular case, K-folds cross-validation was employed with 10 folds, helping to provide sufficient data for training the model and sufficient data for validation. The scoring metric employed to evaluate the model performance will be explained in the following section.

For each of the chosen algorithms, the hyperparameters that were tuned are stated and briefly described:

[12] https://scikit-learn.org/stable/modules/grid_search.html

- C-Support Vector Classifier: The parameters *C*, representing the penalty for misclassifying a data point, and the *gamma* ($\gamma$), the Kernel coefficient, were tuned. For the *kernel* parameter, which selects the type of hyperplane, the default value was employed, i.e. the Radial basis function kernel or *rbf*.
- Multi-layer Perceptron Classifier: The *hidden_layer_sizes* parameter, representing the number of neurons in the *i'th*[13] hidden layer and the *learning_rate*, which controls how quickly or slowly a neural network model learns a problem, were tunned.
- Decision Tree Classifier: The *criterion*, function to measure the quality of a split, the *max_depth*, maximum depth of the tree and the *min_samples_leaf* parameter, the minimum number of samples required to split an internal node, were tunned.
- Random Forest Classifier: The *max_features* parameter, which represents the number of features to consider when looking for the best split, was tunned. The *n_estimators* parameter, corresponding to the number of trees in the forest, was also tunned.
- Adaptive Boosting: The *n_estimators* parameter, denoting the maximum number of estimators at which boosting is concluded, was tunned.

Table 4.6: Range of values tested with grid search

| **Data Mining technique** | **Hyperparameters** |
|---|---|
| C-Support Vector Classifier | *C*: [$10^{-1}$, $10^2$]<br>*Gamma ($\gamma$)*: [$10^{-1}$, $10^2$] |
| Multi-layer Perceptron Classifier | *Hidden_layer_sizes*: [(50, 50, 50), (50, 100, 50), (100,)]<br>*Learning_rate*: [constant, invscaling, adaptive] |
| Decision Tree Classifier | *Criterion*: [gini, entropy]<br>*Max_depth*: [1, 30, step =1]<br>*Min_samples_leaf*:[$10^{-1}$, 0.5] |
| Random Forest Classifier | *Max_features*: [1, number of features, step =1]<br>*N_estimators:* [10, $10^2$,500] |
| Adaptive Boosting | *N_estimators*: [10, $10^2$,500] |

Table 4.6 gives an insight into the hyperparameters' range of values tested upon grid search.

## 4.7 Evaluation

During the evaluation phase, it is crucial to thoroughly evaluate and assess results with respect to business success criteria. This question can be framed as follows:

*What is the most accurate model we can get regarding the business challenge?*

[13] Position i in the sequence of hidden layers.

From the business perspective, it could be extremely risky to misclassify a poor subject as a brilliant subject since the company creates high expectations, leading to disappointment. In fact, saying that a subject has 5 stars quality and afterward, ending with a low OR will lead to dissatisfaction. Indeed, it would be preferable to surprise a company rather than disappointing.

At this point, it is imperative to identify the metrics that are relevant to the problem at hand. As explained in Section 3.1.3.1, the accuracy metric calculates the portion of predictions the model got right which, in this particular case, cannot express how good our model is.

As explained in Section 3.1.3.1, **Precision** metric expresses how precise the model is out of those predicted positives, how many of them are actual positives. Precision is a good measure when the cost of False Positive is high. **Recall** calculates how many of the actual positives our model capture through labeling it as Positive, commonly calculated when the False Negatives have a big impact on business goals. Explaining both metrics for the matter at hand, with a focus on class 5 stars :

- Recall expresses how many subjects were classified as 5 considering all the existent subjects with 5 stars quality class.
- Precision reveals from all subject classified 5 stars quality, how many subjects were well labeled.

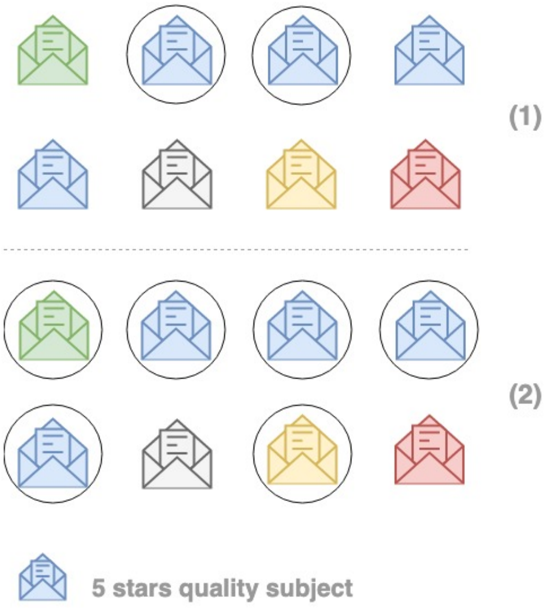


Figure 4.25: Recall and Precision applied to the problem

In Figure 4.25, particularly in sub-figure (1), there is a perfect precision since all the picked 5 stars quality subjects are in fact from class 5 stars, however poor recall since a substantial amount of subjects were not found to belong to class 5 stars. In sub-figure (2), a perfect recall since the classifier detected all 5 stars quality subjects but a poor precision due to a large number of false positives.

Regarding the business goals, and answering the framed question, the most accurate model balances the Recall and Precision results for each of the classes.

**F1 score**, as explained in Section 3.1.3.1, is defined as the harmonic mean between precision and recall and was the metric adopted to evaluate the model and, consequently, for the hyper-parameters tuning. The **F1 score** for a specific class *n* can be calculated through the following equation:

$$F1score_n = 2 \times \frac{precision \times recall}{precision + recall}$$

Given the multi-class problem at hand, the **F1 score** was measured as the **weighted** average of each of the five classes:

$$F1score = \frac{\sum_1^n F1score_n \times No.Samples_n}{\sum_1^n No.Samples_n}$$

Where *F1score_n* is the *F1 score* of class *n* and *No.Samples_n* is the number of samples of class *n*, with n being the number of classes.

The **F1 score** was the scoring metric applied to the parameters tuning, explained in Section 4.6.1. For the algorithm selection, both the F1 score and the accuracy were employed although always weighing the business goals and expectations.

## 4.8 Deployment

This phase includes a summary of the deployment strategy performed including the necessary steps of integration.

### 4.8.1 API Implementation

Once the algorithm was tuned and ready for use, a Representational State Transfer (REST) API was created so that the model could be accessible from the outside of the project and was developed using Flask[14] and FlaskRESTPlus[15] framework. The API Documentation was generated through Swagger[16]. Therefore, two different routes were created:

- **Classifying a subject:** This is the simplest resource available, providing a GET method which returns the predicted quality for a specific subject, with the respective company country and sector.
- **Training the model:** This resource should be used to train or re-train the model. A POST method triggers a training process.

For additional information regarding the developed API, the documentation generated through its specification can be checked at Appendix B.

[14] http://flask.pocoo.org/

[15] https://flask-restplus.readthedocs.io/en/stable/

[16] https://swagger.io/

### 4.8.2 Integration with E-goi

In order to integrate this dissertation project with the E-goi platform, the following steps were taken:

1. Docker[17] integration, automating the deployment of the application inside software containers.

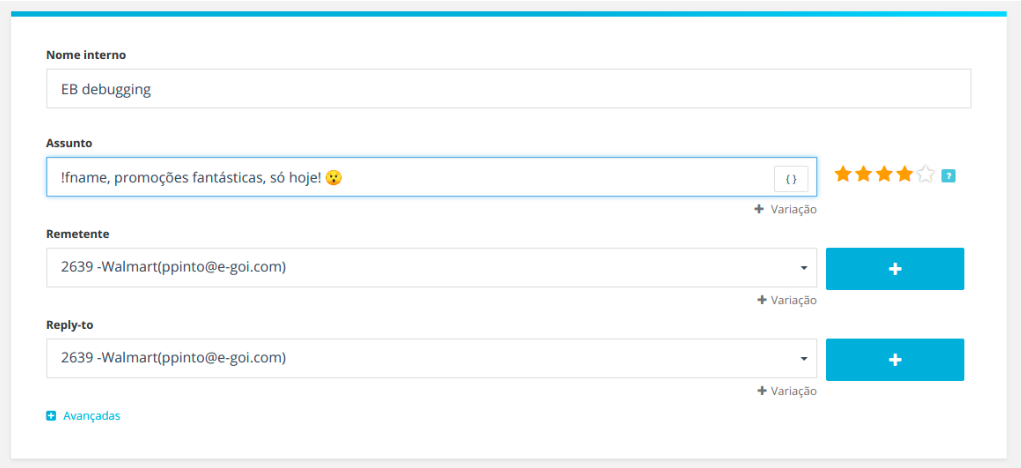
2. In order to keep the model up-to-date, a shell script was created and scheduled to be monthly updated. The script is responsible for updating the data file with the most recent sent campaigns, for also updating the dictionary and, finally, for retraining the model.

3. Integration of the subject analyzer API and subsequent update of the User Interface (UI) in two different environments:

   (a) Campaign creation form: when creating a new email campaign.

   (b) E-goi Content Checker: update the module explained in Section 2.2.2 so that it is also possible to support subject analysis. Therefore, the company existent API service responsible for generating a campaign report regarding the Spam and Source Code was updated in order to also return the subject quality, from one to five stars.

4. With the intention of avoiding misunderstandings and, to give more intelligence to the model, a threshold was stipulated to deal with situations in which the model only recognizes less than $\frac{1}{3}$ of the words contained in the subject-line. In that specific case, the model confesses having no sufficient knowledge to classify the subject.



Nome interno
EB debugging

Assunto
!fname, promoções fantásticas, só hoje! 😮 ★★★★☆

Remetente
2639 -Walmart(ppinto@e-goi.com)

Reply-to
2639 -Walmart(ppinto@e-goi.com)

Figure 4.26: Email campaign creation form

Figure 4.26 illustrates the email campaign creation form. As the user makes changes to the subject-line, the respective quality, in specific the number of stars, is automatically updated.

[17] https://www.docker.com/

## 4.9 Conclusion

Throughout this chapter, each phase of the CRISP-DM process is explored. Firstly, the project goals and the requirements, from a business perspective, are defined. The available data is deeply studied and the creation of the dependent and independent variables is described. Additionally, the different performed experiments are scrutinized, and the employed evaluation metrics, selected taking into account the business problem at hand, are explained. Finally, the process of integration of the tool into the E-goi platform is summarized.

# Chapter 5

# Experiments and Results

This chapter lists all the distinct performed experiments, each one corresponding to a different approach and set of features. The achieved results for each of the six algorithms, mentioned in the previous chapter, are shown and described. Appendix C contains additional information regarding the Experiments.

## 5.1 Experiment 1: Structure analysis

As explained in Section 4.6, the first experiment aims to analyze the subject concerning the country, business sector, and structure, completely disregarding the meaning and the quality of the words included in the subject. The performance of the proposed model for each of the six different algorithms is synthesized in Table 5.1.

Table 5.1: Experiment 1 performance results

| | **Random Forest** | **Neural Network** | **Naive Bayes** | **Gradient Boosting** | **SVM** | **Decision Tree** |
|---|---|---|---|---|---|---|
| F1 Score | 60.4% | 56.3% | 23.1% | 45.6% | 58.2% | 31.7% |
| Accuracy | 60.6% | 56.8% | 32.0% | 47.0% | 58.3% | 41.6% |
| Precision 1 | 69% | 63% | 31% | 53% | 65% | 43% |
| Precision 2 | 51% | 47% | 17% | 38% | 48% | 0% |
| Precision 3 | 50% | 47% | 45% | 41% | 48% | 38% |
| Precision 4 | 55% | 51% | 29% | 37% | 54% | 22% |
| Precision 5 | 76% | 75% | 49% | 57% | 76% | 45% |
| Recall 1 | 72% | 70% | 91% | 67% | 69% | 82% |
| Recall 2 | 51% | 48% | 0% | 26% | 51% | 0% |
| Recall 3 | 47% | 45% | 26% | 42% | 47% | 55% |
| Recall 4 | 53% | 46% | 41% | 30% | 47% | 2% |
| Recall 5 | 80% | 75% | 3% | 70% | 76% | 68% |

Analyzing the obtained results, the Random Forest appears to be the algorithm ensuring better predictive results, with 60.6% of Accuracy and 60.4% of F1 score. The worst result was obtained using Naive Bayes, with 23.1% of F1 score and 32.0% of Accuracy.

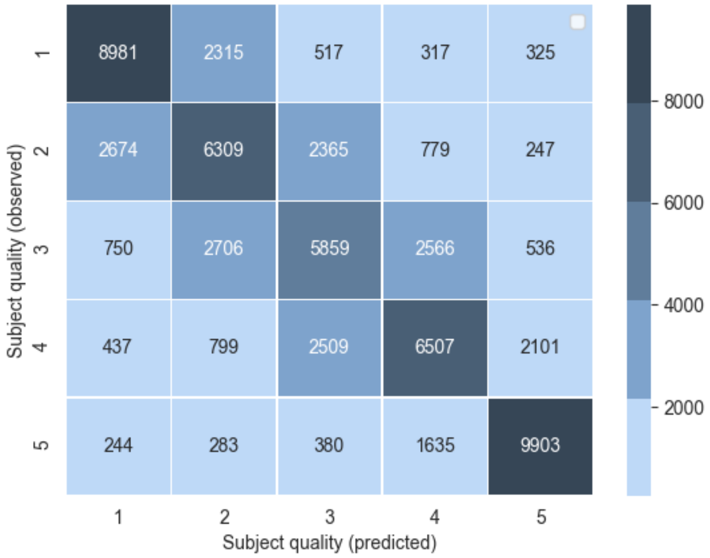


Figure 5.1: Confusion Matrix on Random Forest : Experiment 1

The confusion matrix represented in Figure 5.1 is the result of the classification performed with the Random Forest. From the analysis of this matrix and of the recall and precision values presented in Table 5.1, we can conclude that the model can better classify edge classes, specifically classes 1 and 5, than the other three intermediate classes, i.e. class 2, 3 and 4. From all the intermediate classes, class 3 is the hardest to predict. Indeed, from all the subjects classified with 1 star quality, 69% were well classified and only 28% of the subjects belonging to class 1 were not detected. Regarding class 5, from all the subject-lines classified as 5 stars, 76% were well classified. Moreover, only 20% were not found to belong to class 5.

Furthermore, from the analysis of Table C.1 generated through the Random Forest method for feature selection, the number of chars, case percentage and number of words are the three most important features on the prediction of the subject quality.

## 5.2 Experiment 2: Structure and Content analysis

The second experiment bears on the country, business sector, structure and also on the content features. Totally disregarding the order in which the words appear in the subject, this experiment attempts different approaches on calculating the quality of the lemmas, in order to identify the one which better estimate the subject words quality.

### 5.2.1 Experiment 2.1: Lemmas Past Performance calculated using Weighted Average

As explained in Section 4.6, Experiment 2.1 calculates the variable *lemmas_past_performance* through the weighted average. As a result, the calculation of the past performance of the lemmas depends on the past quality of each lemma and also on how often each lemma was used in emails history.

Table 5.2: Experiment 2.1 performance results

| | Random Forest | Neural Network | Naive Bayes | Gradient Boosting | SVM | Decision Tree |
|---|---|---|---|---|---|---|
| F1 Score | 61.7% | 57.5% | 24.0% | 50.2% | 60.1% | 35.7% |
| Accuracy | 62.1% | 57.7% | 32.7% | 51.1% | 60.1% | 40.2% |
| Precision 1 | 70% | 64% | 32% | 57% | 69% | 48% |
| Precision 2 | 53% | 48% | 24% | 42% | 51% | 32% |
| Precision 3 | 52% | 48% | 44% | 44% | 50% | 30% |
| Precision 4 | 57% | 54% | 29% | 43% | 54% | 0% |
| Precision 5 | 76% | 75% | 54% | 64% | 77% | 53% |
| Recall 1 | 74% | 68% | 89% | 69% | 69% | 50% |
| Recall 2 | 51% | 51% | 0% | 38% | 50% | 50% |
| Recall 3 | 51% | 46% | 28% | 38% | 49% | 32% |
| Recall 4 | 54% | 47% | 43% | 38% | 54% | 0% |
| Recall 5 | 81% | 78% | 3% | 73% | 77% | 69% |

Table 5.2 includes the performance results for the six elected models. In fact, Random Forest Classifier remains the model generating the most accurate results. This time, achieving 61.7% of F1 Score and a 62.1% of Accuracy. Once again, Naive Bayes was the model obtaining the worst results, with only 24% of F1 score and 32.7% of Accuracy.

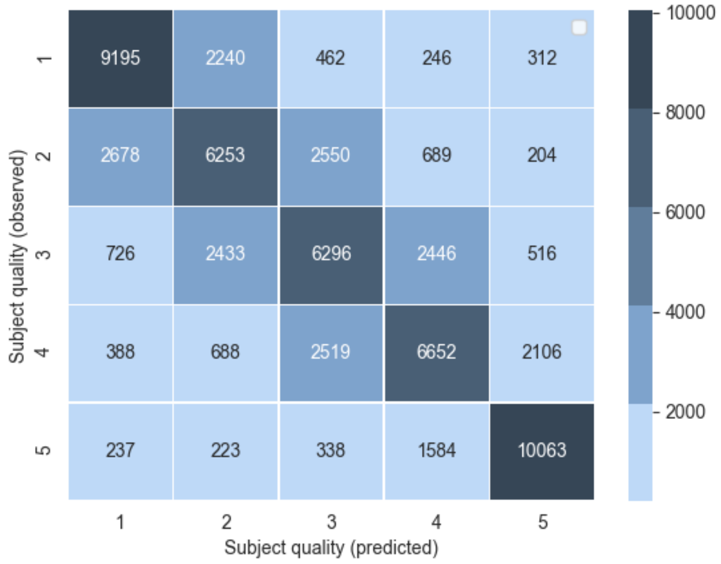


Figure 5.2: Confusion Matrix on Random Forest : Experiment 2.1

Figure 5.2 is the confusion matrix generated using the Random Forest classifier. Similarly to what was concluded in Experiment 1, the model predicts remarkably well on edge classes, i.e. 1 and 5. Indeed, class 5 remains the easiest to predict, with 76% of Precision and 81% of Recall, followed by class 1, with 70% of Precision and 74% of Recall.

However, as can be anticipated, the cells adjacent to the diagonal are very populated, meaning that there is a significant number of subjects classified with quality exactly one class above or one class below from what was expected. In a nutshell, taking as example subjects of quality 2, there exists a considerable number of those subjects being predicted to belong to class 1 or to class 3.

Table C.2 in Appendix C confirms the substantial contribution of the *lemmas past performance* in the final decision. The following two most important features are the number of characters and the case percentage.

When compared to the latter experiment, this one achieved better and more accurate results, with a positive difference of 1,3% regarding the F1 Score. Hence, it is plausible to conclude that the past performance of the words contained in the subject is relevant in the prediction of the overall subject quality.

### 5.2.2 Experiment 2.2: Lemmas Past Performance calculated using Average

As explained in Section 4.6, Experiment 2.2 calculates the variable *lemmas_past_performance* by the average of the quality of each lemma contained in the subject.

Table 5.3: Experiment 2.2 performance results

| | **Random Forest** | **Neural Network** | **Naive Bayes** | **Gradient Boosting** | **SVM** | **Decision Tree** |
|---|---|---|---|---|---|---|
| F1 Score | 62.2% | 58.8% | 24.1% | 51.8% | 60.5% | 36.5% |
| Accuracy | 62.4% | 58.9% | 32.7% | 52.6% | 60.6% | 43.4% |
| Precision 1 | 71% | 67% | 32% | 59% | 69% | 55% |
| Precision 2 | 53% | 49% | 28% | 44% | 50% | 31% |
| Precision 3 | 52% | 48% | 44% | 44% | 50% | 31% |
| Precision 4 | 57% | 54% | 29% | 43% | 55% | 0% |
| Precision 5 | 77% | 76% | 54% | 66% | 78% | 58% |
| Recall 1 | 74% | 70% | 89% | 68% | 70% | 56% |
| Recall 2 | 50% | 48% | 0% | 37% | 50% | 20% |
| Recall 3 | 51% | 50% | 28% | 42% | 50% | 59% |
| Recall 4 | 55% | 48% | 43% | 40% | 55% | 0% |
| Recall 5 | 82% | 79% | 3% | 75% | 78% | 82% |

Table 5.3 shows that Random Forest Classifier is, once again, the model performing better, ending up with an F1 score of 62.2% and an Accuracy of 62.4%. Naive Bayes was the model with the worst performance results, with a F1 Score of 24.1% and 32.7% of Accuracy.

Figure 5.3 is the confusion matrix obtained through the Random Forest classification. The matrix presents results extremely similar to the ones produced in the previous experiment, with class 1 and especially class 5, being the easiest to predict. The small number of samples identified through the row and column 5, except for the position [5,5] report how well the model can predict a 5 stars quality subject, in conformity with the excellent value of 77% of Precision and 82% of Recall.

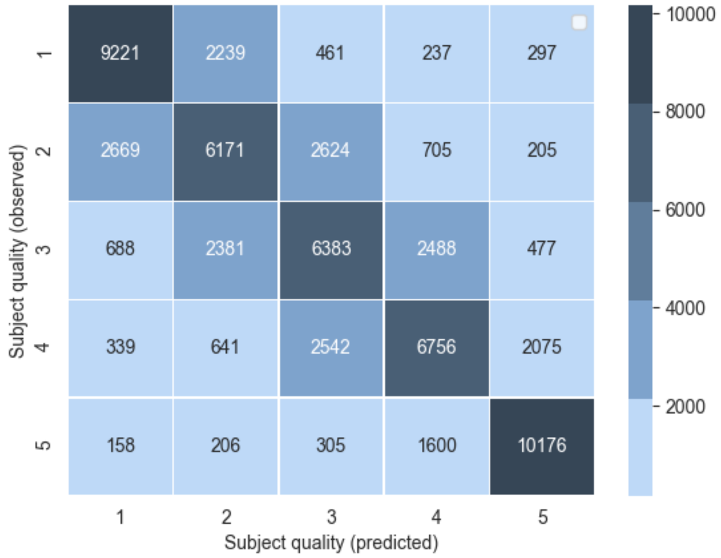


Figure 5.3: Confusion Matrix on Random Forest : Experiment 2.2

Table C.3 lists the most relevant features and the respective importance value.

In comparison to Experiment 2.1, even though both have achieved pretty similar performance results, Experiment 2.2 produced a higher value for the F1 Score, with an improvement of 0,5%. Therefore, the lemmas past performance feature measured through the naive average proved to predict better concerning the quality of the subject-line.

### 5.2.3 Experiment 2.3: Lemmas Past Performance calculated using Maximum

As mentioned in Section 4.6, Experiment 2.3 estimates the *lemmas_past_performance* variable by the maximum quality of the lemmas contained in the subject.

Table 5.4: Experiment 2.3 performance results

| | **Random Forest** | **Neural Network** | **Naive Bayes** | **Gradient Boosting** | **SVM** | **Decision Tree** |
|---|---|---|---|---|---|---|
| F1 Score | 61.5% | 57.8% | 23.6% | 48.8% | 59.7% | 39.1% |
| Accuracy | 61.8% | 58.1% | 32.5% | 50.0% | 59.8% | 44.2% |
| Precision 1 | 69% | 64% | 32% | 56% | 68% | 47% |
| Precision 2 | 52% | 47% | 25% | 41% | 51% | 0% |
| Precision 3 | 52% | 48% | 44% | 44% | 47% | 42% |
| Precision 4 | 57% | 54% | 29% | 39% | 55% | 29% |
| Precision 5 | 76% | 76% | 50% | 62% | 78% | 60% |
| Recall 1 | 73% | 32% | 89% | 66% | 69% | 74% |
| Recall 2 | 51% | 25% | 0% | 31% | 48% | 0% |
| Recall 3 | 49% | 44% | 27% | 42% | 52% | 37% |
| Recall 4 | 54% | 29% | 43% | 34% | 52% | 40% |
| Recall 5 | 81% | 50% | 3% | 76% | 78% | 70% |

Table 5.4 demonstrates that the Random Forest is the model generating better results, with a F1 Score of 61.5% and an Accuracy of 61.8%. Naive Bayes continues the model performing worst, with a F1 Score of 23.6% and an Accuracy of 32.5%.

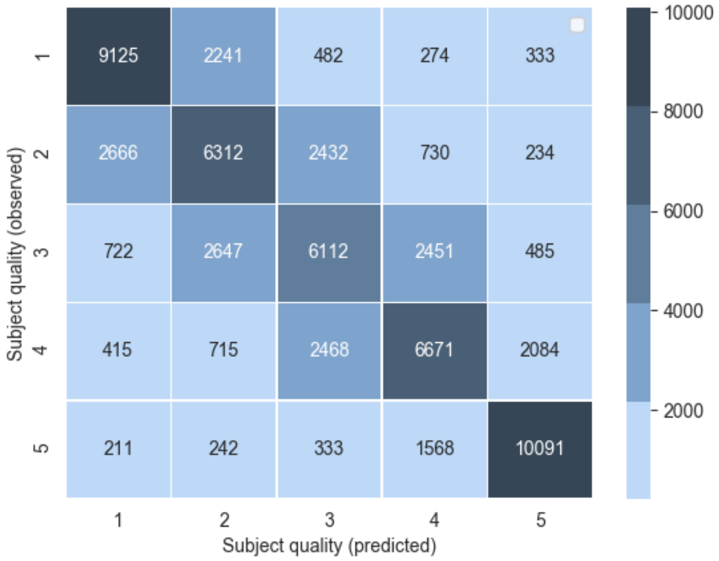


Figure 5.4: Confusion Matrix on Random Forest : Experiment 2.3

Figure 5.4 evinces the model difficulty in predicting subjects of quality 3. As already stated in the earlier experiments, the model works extremely well on edge classes, i.e. 1 and 5. The 5 stars quality subjects are the easiest to predict, producing low values of False Positives and False Negatives and, hence, ending with very satisfactory values of Precision, in specific 76% and Recall, with 81%.

Nevertheless, what differs this experiment from the first two, is the importance of the features. Table C.4 shows the list of the features and the respective importance value. While in Experiment 2.1 and Experiment 2.2, the *lemmas_past_performance* feature was the one impacting most in the final result, in the current experiment, this feature loses relevance.

As illustrated in Section 4.3 and, once again represented in Figure 5.5, the *lemmas_past_perfor_ mance* feature, when calculated through the Maximum approach, cannot be straight associated with the quality class. Indeed, for example, subjects with *lemmas_past_performance* of quality 4 were spread through all five classes.

This experiment reached worse results comparing to Experiment 2.2, with an F1 Score reduced in 0.7%.

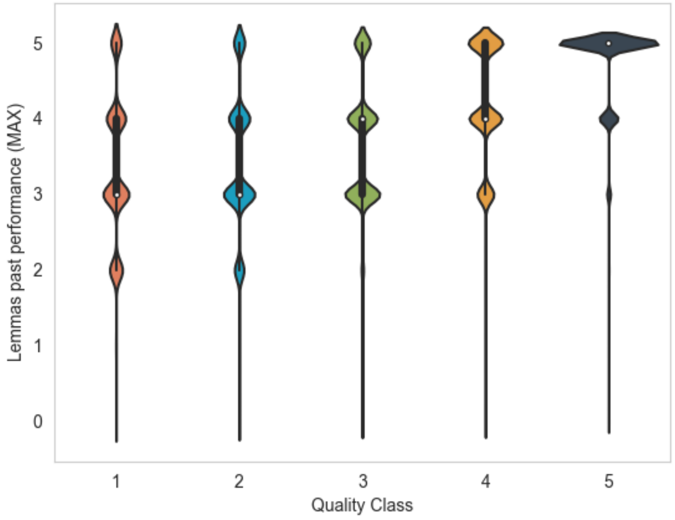


Figure 5.5: Relation between the quality class and the lemmas past performance (MAX)

## 5.3 Experiment 3: Structure Analysis and Bag Of Words With Feature Selection

As explained in Section 4.6, Experiment 3 operates with the BOW approach in order to study the quality of the words contained in the subject. Therefore, a BOW approach with a maximum of 4% of the vocabulary was employed with a subsequent dimensionality reduction in data. The vocabulary was chosen considering the term frequency across the corpus.

When employing the BOW approach, each word of the vocabulary becomes a new feature of the model and so, space and time complexity increase exponentially. Hence, and due to memory issues, it was impossible to manipulate more than 4% of the words of the corpus.

With the purpose of attenuating the dimensionality issues, PCA was used. It reduces the number of features while preserving as much information as possible. When invoking PCA, the parameters were set with the intention of giving the model responsibility to guess the most suitable feature dimension.

Figure 5.5 presents the performance results for each employed algorithm. Random Forest is the classification technique achieving the most accurate results, with a F1 Score of 60.0% and an Accuracy of 60.2%, and the Decision Tree and Naive Bayes are the ones producing worst results.

Even though this technique is one of the most common and easy methods used in text classification, BOW representation suffers from its intrinsic extreme sparsity and high dimensionality [ZM18]. Indeed, when compared to the other experiments, Random Forest produced worse results and Naive Bayes slightly better outcomes, what could be related to the large number of features. Actually, Trees usually fail in situations where the data is sparse [TGvL18].

These unsatisfactory results can be related to the need of dealing with three different languages, i.e. pt, en, and es. In fact, the model evaluates not only one but three languages and

Table 5.5: Experiment 3 performance results

| | **Random Forest** | **Neural Network** | **Naive Bayes** | **Gradient Boosting** | **SVM** | **Decision Tree** |
|---|---|---|---|---|---|---|
| F1 Score | 60.0% | 59.6% | 35.0% | 50.0% | 44.0% | 32.0% |
| Accuracy | 60.2% | 59.5% | 36.6% | 50.9% | 46.9% | 41.6% |
| Precision 1 | 67% | 68 % | 42% | 56% | 45% | 43% |
| Precision 2 | 52% | 51% | 32% | 43% | 41% | 0% |
| Precision 3 | 57% | 49% | 37% | 43% | 40% | 38% |
| Precision 4 | 56% | 54% | 27% | 45% | 40% | 22% |
| Precision 5 | 75% | 74% | 43% | 65% | 69% | 45% |
| Recall 1 | 72% | 71% | 38% | 66% | 77% | 82% |
| Recall 2 | 46% | 48% | 31% | 39% | 20% | 0% |
| Recall 3 | 54% | 50% | 14% | 36% | 56% | 55% |
| Recall 4 | 51% | 52% | 36% | 43% | 19% | 2% |
| Recall 5 | 79% | 78% | 65% | 70% | 62% | 68% |

thus, the number of lemmas available in each one could be insufficient. Furthermore, whereas in Experiment 2 we assign a quality to each known word, if it has already been sent at least once in the past email campaigns, in this case, only specific words have relevance in the prediction and the others are totally insignificant.

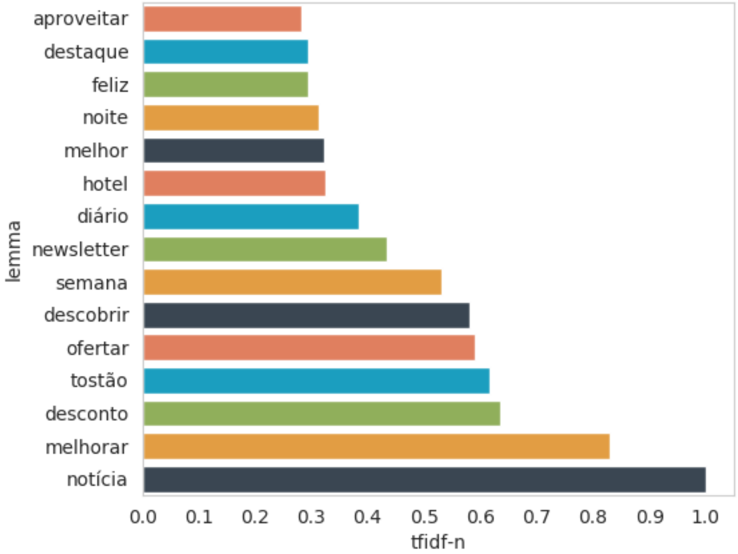


Figure 5.6: Top 15 most relevant lemmas

Figure 5.6 presents the 15 most relevant lemmas, obtained by the BOW approach.

Figure 5.7 illustrates the confusion matrix obtained through the Random Forest Classification. Contrary to Experiments 1 and 2.3, there exists a balanced percentage of Precision and Recall values among the classes. Although class 5 continues to be the easiest to predict, this time, all the classes have a similar probability of predicting the subject quality successfully.

Working with only 4% of the vocabulary was not enough to surpass the performance of Experiment 2.2, although we believe that with more vocabulary the results could have improved.

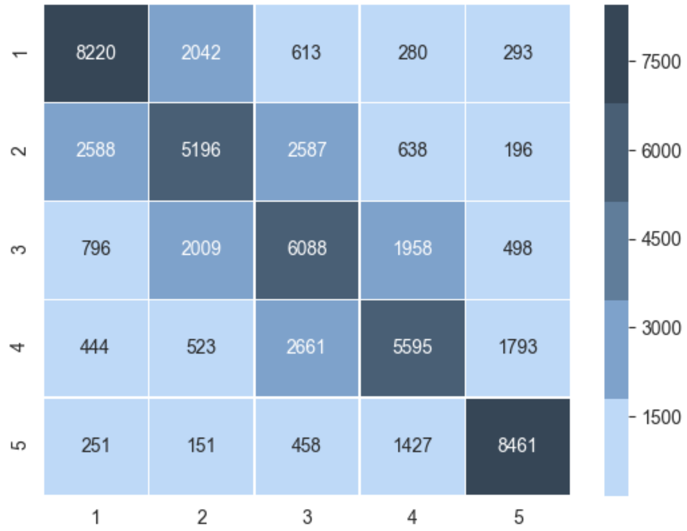


Figure 5.7: Confusion Matrix on Random Forest : Experiment 3

## 5.4 Final Discussion and Conclusion

Table 5.6 summarizes the best performance results achieved for each conducted experiment, in particular, the F1 Score and Accuracy.

Table 5.6: Summary of the performance results for each experiment

| | **Description** | **Performance (F1 Score / Accuracy )** |
|---|---|---|
| **Experiment 1** | Structural Analysis | 60.4% / 60.6% |
| **Experiment 2.1** | Structural + Content Analysis *(WAVG)* | 61.7% / 62.1% |
| **Experiment 2.2** | Structural + Content Analysis *(AVG)* | 62.2% / 62.4% |
| **Experiment 2.3** | Structural + Content Analysis *(MAX)* | 61.5% / 61.8% |
| **Experiment 3** | Structural + Bag Of Words + PCA | 60.3% / 60.6% |

Reframing the question made in Section 4.7, *What is the most accurate model we can get regarding the business challenge?* As explained earlier, the most accurate model avoids classifying a low quality subject as an excellent subject and hence, seeks to maximize the F1 Score metric.

Therefore, the results bring out Experiment 2.2 as being the one yielding the best results. To classify a subject quality it uses the country, sector, structural and content features, employing the average to calculate the *lemmas_past_performance*, producing a F1 score of 62.2% and an Accuracy of 62.4%.

It is important to highlight the great achievements, especially when taking into account the multi-class problem at hand and that, a random prediction would produce a $\frac{1}{5}$% of Accuracy, only 20% compared to the notable achieved 62%.

# Chapter 6

# Conclusions and Future Work

## 6.1 Conclusions

This study was set out to overcome and improve the notably low open rates that are hindering companies of engaging efficiently their audience. Thus, this project sought to help editors on creating relevant subject-lines which capture recipients' attention, creating sufficient curiosity on them to open and read the content of the message.

This thesis proposes a model, which, employing data mining classification techniques, is capable of predicting a subject-line quality, considering the country from where is sent, the business sector, the structural and the content features, i.e. the set of words chosen.

In this particular case study and, regardless of the specificities of the different model's tests, Random Forest Classification algorithm is the one providing the best results. The experiment which calculates the *lemmas_past_performance* feature using the naive average of the quality of the lemmas contained in the subject achieved extremely favorable performance results when employed with Random Forest Classification technique.

Therefore, we conclude that the data available for this study was sufficient for an effective prediction of the subject quality, with the model reaching performance levels of about 62.4%, in terms of Accuracy and 62.2%, in terms of F1 Score, although believing that some external and hidden characteristics of the data may have contributed to lower outcomes as, for example, information regarding the quality of the list of subscribers or the day of the week the campaign was delivered.

From E-goi standpoint, this promising tool will support their customers on creating engaging and relevant subject-lines, contributing to better performance results, i.e. more emails opened and, thus, more successful marketing campaigns. Indeed, this thesis contributed to the development of the first tool embedded into a marketing automation platform, able to predict the quality of a subject through Data Mining techniques. Moreover, despite all the research carried out in the area, this project is the first one offering a model prepared for adjusting itself to the natural evolution of trends and patterns over time.

Conversely, predicting the probability of an email to be open through the subject-line quality remains a complex task due to diverse factors. Firstly, personal interests and curiosities can strongly influence an individual to open or to neglect a specific email. Secondly, the sender recognition and reputation can influence the decision to open or overlook an email and finally, owning an outdated customer database, with low-quality user data can completely skew email campaign performance results.

In conclusion, this thesis provides a prosperous and extremely valuable tool, not only for E-goi platform but also for any other marketing automation platform seeking to offer customers solutions to overcome their daily obstacles during the creation of email marketing campaigns.

A research paper, attached in Appendix D, and based on the present thesis research, was submitted to the International Journal of Production Economics[1].

## 6.2 Future Work

Evaluating the overall performance of each of the models through nested cross validation was time-consuming and hence, many different experiments and ideas have been left for the future due to the lack of time.

Considering that, these days, virtually every company in the world uses email marketing to operationalize their customer relationship management strategy, this field has a tendency to continue lacking for advances and improvements for better reaching their customer's needs. Therefore, there is a list of some promising future works that could be developed:

1. To measure customer satisfaction when using and interacting with the tool. Considering that the deployment phase was carried out in the final stage of this thesis, it was not possible to make a review of the subject analyzer tool, when integrated into the E-goi platform. It could be valuable to try assessing what needs to be improved.

2. To study the possibility of integration of the product into other marketing automation platforms. Given that this tool is completely agnostic and interoperable, the embodiment among other systems should be evaluated.

3. To build a different model for each distinct language. Having multiple languages in the same model increases complexity which can lead to less accurate performance results.

4. To implement n-grams approach in the content analysis. N-gram [WMW07] is simply a sequence of N words and, instead of just interpreting word by word, the model could interpret a set of *n* words as a whole. Hence, it would take into consideration expressions like *out of stock*, which together mean much more than individually.

[1] https://www.journals.elsevier.com/international-journal-of-production-economics

5. To consider both the subject and the sender reputation as the OR drivers, recognizing that notorious companies such as Zara, Apple or Ryanair can influence the willingness and curiosity to open an email.

6. To include sentiment analysis, quantifying the psychological effects induced when reading an email subject-line, for example, happiness, curiosity or sadness.

7. To analyze the subject semantically, i.e the general meaning of the subject, instead of analyzing just the set of words, each one treated independently and not considering any order.

8. To get to know the type of subscriber and what he/she enjoys most to read. This may involve trying to define a customer profile, for example using age as a proxy for his/her behavior. For instance, the use of emojis can influence differently depending on whether an email is sent to a teenager or to an adult.

9. To try employing a more sophisticated model for feature selection, for example, forward and backward selection procedure.

10. To explore the potential of techniques dedicated to ordinal dependent variables, namely neural network and support vector approaches to ordinal regression.

# References

[Abu17] Ibrahim Abu El-Khair. TF*IDF. In *Encyclopedia of Database Systems*. Springer, New York, NY, 2017.

[Apo15] Durga Apoorv. Email Marketing and Marketing Automation Tools. *Econtent*, 11(1):30–32, 2015.

[Apo17a] Dimitris Apostolopoulos. Improving Subject Lines: An endless quest using Machine Learning (pt. 1/2), 2017.

[Apo17b] Dimitris Apostolopoulos. Improving Subject Lines: An endless quest using Machine Learning (pt. 2/2), 2017.

[BP15] Raju Balakrishnan and Rajesh Parekh. Learning to predict subject-line opens for large-scale email marketing. In *Proceedings - 2014 IEEE International Conference on Big Data, IEEE Big Data 2014*, pages 579–586, 2015.

[Bre01] Leo Breiman. *Random Forests*. Kluwer Academic Publishers, 2001.

[Bre04] Leo Breiman. *Bagging predictors*, volume Machine Le. Kluwer Academic Publishers, 2004.

[Dan17] Raluca Dania TODOR. Promotion and communication through e-mail marketing campaigns. *Bulletin of the Transilvania University of Braşov Series V: Economic Sciences*, 10(59), 2017.

[DP06] Geoffrey B. Duggan and Stephen J. Payne. How much do we understand when skim reading? page (p. 730). Association for Computing Machinery (ACM), 2006.

[FFK$^+$13] Sebastian Feld, Heiko Frenzen, Manfred Krafft, Kay Peters, and Peter C. Verhoef. The effects of mailing design characteristics on direct mail campaign performance. *International Journal of Research in Marketing*, 30(2):143–159, 2013.

[Fri02] Jerome H. Friedman. Stochastic gradient boosting. *Computational Statistics and Data Analysis*, 2002.

[FSGS14] Christina Ramires Ferreira, Sérgio Adriano Saraiva, Jerusa Simone Garcia, and Gustavo Braga Sanvido. Feature Engineering for Text Classification. *Bioinformatics*, 2014.

[Joh] Brendan Tierney John D. Kelleher. Data Science. The MIT Press (April 13, 2018).

[Lew98] David D. Lewis. Naive (Bayes) at forty: The independence assumption in information retrieval, 1998.

[LJ09] Stan Z Li and Anil Jain, editors. *PCA (Principal Component Analysis)*, page 1056. Springer US, Boston, MA, 2009.

[MC17] R. Miller and E. Y.A. Charles. A psychological based analysis of marketing email subject lines. In *16th International Conference on Advances in ICT for Emerging Regions, ICTer 2016 - Conference Proceedings*, pages 58–65, 2017.

[OG10] Oleg Okun and Giorgio Valentini. Supervised and Unsupervised Ensemble Methods and their Applications. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2010.

[Qui86] J. R. Quinlan. Induction of Decision Trees. *Machine Learning*, pages 81–106, 1986.

[Ras17] Sebastian Raschka. *Python Machine Learning*. Packt Publishing ©2015, 2017.

[Sen15] Shahana Sen. An Overview of Data Mining and Marketing. 4(5):254–259, 2015.

[SL16] Natalie Sappleton and Fernando Lourenço. Email subject lines and response rates to invitations to participate in a web survey and a face-to-face interview: the sound of silence. *International Journal of Social Research Methodology*, 2016.

[SM13] C. Saranya and G. Manikandan. A study on normalization techniques for privacy preserving data mining. *International Journal of Engineering and Technology*, 2013.

[Ste18] Andreas ChristmannIngo Steinwart. Support Vector Machines, 2018.

[SWC16] Navdeep S. Sahni, S. Christian Wheeler, and Pradeep K. Chintagunta. Personalization in Email Marketing: The Role of Non-Informative Advertising Content. *Ssrn*, pages 1–41, 2016.

[TGvL18] Cheng Tang, Damien Garreau, and Ulrike von Luxburg. When do random forests fail? 2018.

[Tin10] Kai Ming Ting. *Confusion Matrix*, page 209. Springer US, Boston, MA, 2010.

[TSK$^+$06] Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Tan Pang-Ning, Michael Steinbach, and Vipin Kumar. Introduction to data mining: Instructur's. *Library of Congress*, page 769, 2006.

[VB 16] VB STAFF. Email marketing: Learn the strategies that are achieving 300% ROI and more (VB Live), 2016.

[Wat18] Watson Marketing. Watson Marketing Email and Mobile Metrics for Smarter Marketing 2018 Marketing Benchmark Report. page 48, 2018.

[WDK11] Jaclyn Wainer, Laura A Dabbish, and Robert Kraut. Should I open this email?: inbox-level cues, curiosity and attention to email. 2011.

[Wir00] Rüdiger Wirth. 13 CRISP-DM : Towards a Standard Process Model for Data Mining. *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining*, (24959):29–39, 2000.

[WMW07] Xuerui Wang, Andrew McCallum, and Xing Wei. Topical N-grams: Phrase and topic discovery, with an application to information retrieval. In *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2007.

[Zer18] Aegeus Zerium. Artificial Neural Networks Explained, 2018.

[ZM18] Rui Zhao and Kezhi Mao. Fuzzy Bag-of-Words Model for Document Representation. *IEEE Transactions on Fuzzy Systems*, 26(2):794–804, 2018.

# REFERENCES

# Appendix A

# Dataset Variables

Table A.1: Dataset variables

| Attribute | Type | Description | Values (Frequency)/ Mean (std. deviation) |
|---|---|---|---|
| 1 | Categorical | Subject quality in stars | 1: 20%<br>2: 20%<br>3: 20%<br>4: 20%<br>5: 20% |
| 2 | Categorical | Country | Angola: 0.078%<br>Argentina: 0.007%<br>Australia: 0.075%<br>Brazil: 26.310%<br>Chile: 0.009%<br>Colombia: 0.246%<br>Curacao: 0.001%<br>Spain: 0.996%<br>Honduras: 0.003%<br>Malta: 0.002%<br>Mexico: 0.017%<br>Morocco: 0.013%<br>Mozambique: 0.013%<br>Peru 0.143%<br>Portugal: 71.350%<br>United Kingdom: 0.023%<br>United States of America: 0.730% |
| 3 | Categorical | Sector | Accounting/Financial Services: 0.362%<br>Architecture/Construction: 0.210%<br>Arts and Entertainment: 3.810%<br>Beauty/Health: 1.908%<br>Church: 0.238%<br>E-commerce: 9.889%<br>Education: 2.143%<br>Food: 0.192%<br>Government: 0.464% |

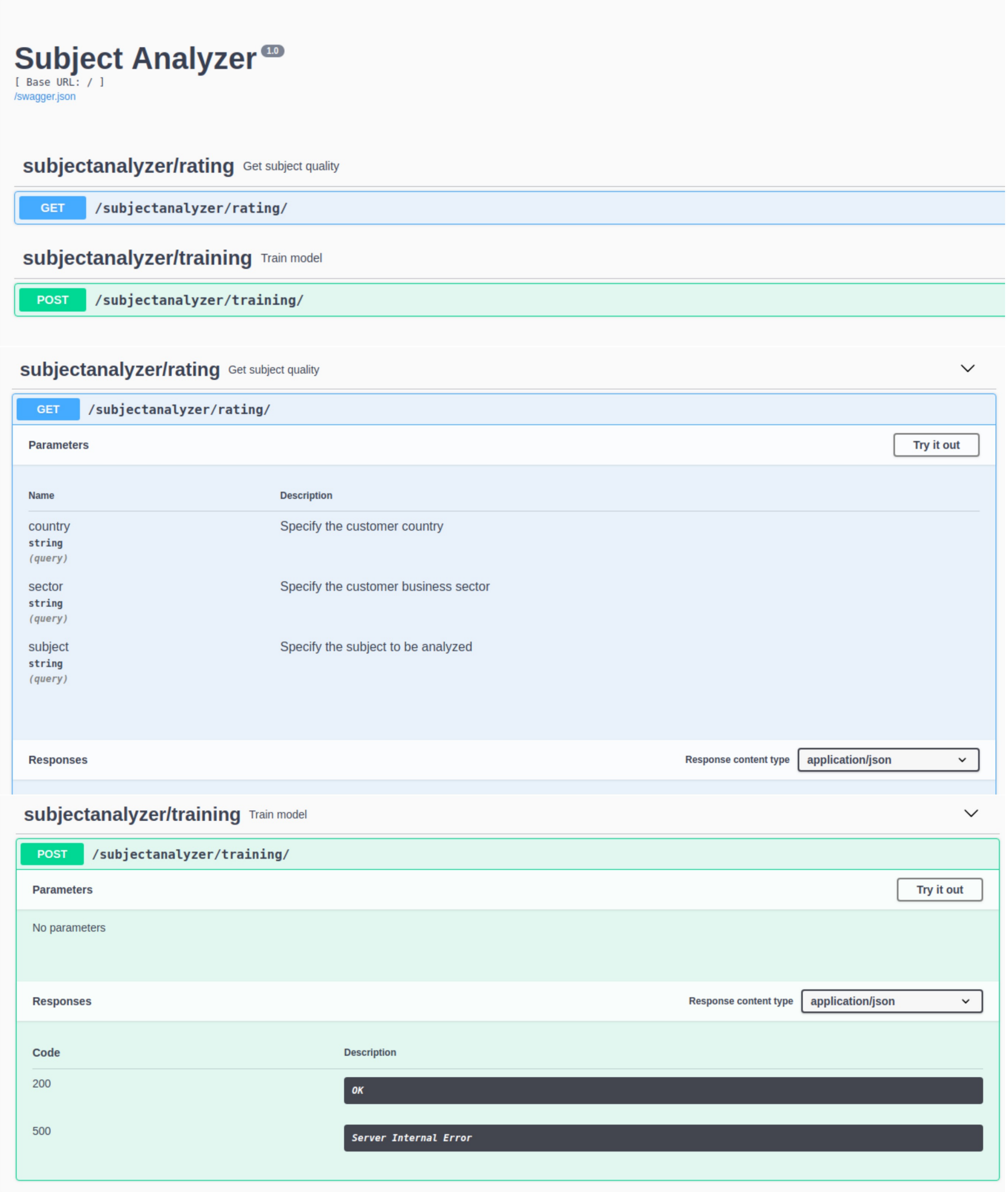
*Continued on next page*

Table A.1 – *Continued from previous page*

| **Attribute** | **Type** | **Description** | **Values(Frequency)/Mean(std.deviation)** |
|---|---|---|---|
| | | | Hotel, Restaurant and Travel: 10.725% |
| | | | Information Technologies: 4.969% |
| | | | Insurance: 0.023% |
| | | | Manufacturing: 1.047% |
| | | | Marketing: 20.415% |
| | | | Non-profit: 1.542% |
| | | | Other: 2.560% |
| | | | Publishing/Media: 12.045% |
| | | | Real Estate: 0.359% |
| | | | Retail: 2.904% |
| | | | Services: 7.595% |
| | | | Undefined: 15.604% |
| 4 | Categorical | Punctuation | False: 77.4% |
| | | | True: 22.6% |
| 5 | Categorical | Prefix | False: 99.9% |
| | | | True: 0.1% |
| 6 | Categorical | Emojis | False: 98.1% |
| | | | True: 1.9% |
| 7 | Categorical | Personalization | False: 95.1% |
| | | | True: 4.9% |
| 7 | Categorical | Special chars | False: 34.6% |
| | | | True: 65.4% |
| 8 | Categorical | Numbers | False: 60.1% |
| | | | True: 39.9% |
| 9 | Categorical | Currency | False: 81.9% |
| | | | True: 18.1% |
| 10 | Numerical | Number of characters | 61.2 (37.18) |
| 11 | Numerical | Number of words | 10.5 (6.84) |
| 12 | Numerical | Case percentage | 12.0 (15.37) |
| 13 | Numerical | Lemmas past performance (W_AVG) | 3.2 (0.99) |
| | Numerical | Lemmas past performance (AVG) | 3.2 (0.93) |
| | Numerical | Lemmas past performance (MAX) | 3.9 (0.97) |
| 13 | Numerical | Number of lemmas | 4.9 (3.1) |

# Appendix B

# API’s Swagger UI



Subject Analyzer 1.0

[ Base URL: / ]

/swagger.json

subjectanalyzer/rating Get subject quality

GET /subjectanalyzer/rating/

subjectanalyzer/training Train model

POST /subjectanalyzer/training/

subjectanalyzer/rating Get subject quality

GET /subjectanalyzer/rating/

Parameters

| Name | Description |
|---|---|
| country<br>string<br>*(query)* | Specify the customer country |
| sector<br>string<br>*(query)* | Specify the customer business sector |
| subject<br>string<br>*(query)* | Specify the subject to be analyzed |

Responses

Response content type application/json

subjectanalyzer/training Train model

POST /subjectanalyzer/training/

Parameters

No parameters

Responses

Response content type application/json

| Code | Description |
|---|---|
| 200 | *OK* |
| 500 | *Server Internal Error* |

API’s Swagger UI

# Appendix C

# Experiences Results

Table C.1: Normalized features importance: Experience 1

| Feature | Importance | Feature | Importance |
|---|---|---|---|
| no_chars | 1,0000 | Non-profit | 0,0158 |
| case_percentage | 0,6192 | Manufacturing | 0,0134 |
| no_ words | 0,5650 | spain | 0,0117 |
| Publishing/Media | 0,2150 | Government | 0,0077 |
| Marketing | 0,1963 | Food | 0,0067 |
| brazil | 0,1079 | united_states_of_america | 0,0059 |
| special_chars | 0,0945 | colombia | 0,0050 |
| punctuation | 0,0934 | Accounting/Financial Services | 0,0048 |
| numbers | 0,0909 | Real Estate | 0,0045 |
| portugal | 0,0844 | Architecture/Construction | 0,0035 |
| Hotel-Restaurant and Travel | 0,0787 | Church | 0,0034 |
| currency | 0,0640 | peru | 0,0028 |
| Services | 0,0451 | angola | 0,0012 |
| Undefined | 0,0402 | australia | 0,0005 |
| Information Technologies | 0,0353 | Insurance | 0,0004 |
| E-commerce | 0,0333 | united_kingdom | 0,0004 |
| Retail | 0,0279 | mexico | 0,0003 |
| Other | 0,0270 | argentina | 0,0003 |
| Arts and Entertainment | 0,0257 | prefix | 0,0002 |
| personalization | 0,0252 | mozambique | 0,0001 |
| Education | 0,0248 | honduras | 0,0001 |
| emojis | 0,0200 | chile | 0,0000 |
| Beauty/Health | 0,0172 | malta | 0,0000 |
|  |  | curacao | 0,0000 |

Table C.2: Normalized features importance: Experience 2.1

| **Feature** | **Importance** | Feature | Importance |
|---|---|---|---|
| lemmas_past_performance | 1,0000 | Non-profit | 0,0180 |
| no_chars | 0,7326 | Manufacturing | 0,0164 |
| case_percentage | 0,5389 | spain | 0,0129 |
| no_ words | 0,5155 | Government | 0,0078 |
| no_lemmas | 0,3570 | Food | 0,0070 |
| Marketing | 0,1956 | Real Estate | 0,0054 |
| Publishing/Media | 0,1752 | united_states_of_america | 0,0052 |
| brazil | 0,1160 | Accounting/Financial Services | 0,0052 |
| Hotel-Restaurant and Travel | 0,1098 | Architecture/Construction | 0,0042 |
| portugal | 0,1048 | colombia | 0,0040 |
| punctuation | 0,0938 | Church | 0,0027 |
| numbers | 0,0910 | peru | 0,0022 |
| special_chars | 0,0883 | angola | 0,0013 |
| currency | 0,0759 | australia | 0,0007 |
| Undefined | 0,0569 | united_kingdom | 0,0004 |
| Services | 0,0563 | Insurance | 0,0004 |
| E-commerce | 0,0441 | mexico | 0,0004 |
| Information Technologies | 0,0394 | argentina | 0,0003 |
| Arts and Entertainment | 0,0333 | prefix | 0,0002 |
| Retail | 0,0302 | mozambique | 0,0001 |
| personalization | 0,0285 | honduras | 0,0001 |
| Other | 0,0283 | malta | 0,0001 |
| Education | 0,0247 | chile | 0,0000 |
| emojis | 0,0223 | curacao | 0,0000 |
| Beauty/Health | 0,0197 | | |

Table C.3: Normalized features importance: Experience 2.2

| **Feature** | **Importance** | **Feature** | **Importance** |
|---|---|---|---|
| lemmas_past_performance | 1,0000 | Non-profit | 0,0171 |
| no_chars | 0,8143 | Manufacturing | 0,0164 |
| case_percentage | 0,5891 | spain | 0,0123 |
| no_ words | 0,5596 | Government | 0,0078 |
| no_lemmas | 0,3768 | Food | 0,0062 |
| Marketing | 0,1816 | Accounting/Financial Services | 0,0054 |
| Publishing/Media | 0,1476 | Real Estate | 0,0054 |
| Hotel-Restaurant and Travel | 0,1143 | united_states_of_america | 0,0053 |
| brazil | 0,1142 | Architecture/Construction | 0,0043 |
| portugal | 0,1070 | colombia | 0,0041 |
| punctuation | 0,0969 | Church | 0,0031 |
| numbers | 0,0929 | peru | 0,0025 |
| special_chars | 0,0871 | angola | 0,0011 |
| currency | 0,0849 | australia | 0,0008 |
| Services | 0,0577 | Insurance | 0,0005 |
| Undefined | 0,0572 | united_kingdom | 0,0004 |
| E-commerce | 0,0437 | mexico | 0,0004 |
| Information Technologies | 0,0361 | argentina | 0,0003 |
| Arts and Entertainment | 0,0309 | prefix | 0,0003 |
| personalization | 0,0295 | mozambique | 0,0001 |
| Retail | 0,0294 | honduras | 0,0001 |
| Other | 0,0276 | chile | 0,0000 |
| Education | 0,0247 | malta | 0,0000 |
| emojis | 0,0214 | curacao | 0,0000 |
| Beauty/Health | 0,0206 | | |

Table C.4: Normalized features importance: Experience 2.3

| Feature | Importance | Feature | Importance |
|---|---|---|---|
| no_chars | 1,0000 | Non-profit | 0,0202 |
| case_percentage | 0,7120 | Manufacturing | 0,0179 |
| no_ words | 0,6726 | spain | 0,0148 |
| no_lemmas | 0,4690 | Government | 0,0095 |
| lemmas_past_performance | 0,4689 | Food | 0,0071 |
| Marketing | 0,2092 | Accounting/Financial Services | 0,0065 |
| Publishing/Media | 0,1800 | Real Estate | 0,0060 |
| brazil | 0,1296 | united_states_of_america | 0,0059 |
| Hotel-Restaurant and Travel | 0,1278 | Architecture/Construction | 0,0049 |
| portugal | 0,1232 | colombia | 0,0047 |
| punctuation | 0,1118 | Church | 0,0035 |
| numbers | 0,1067 | peru | 0,0025 |
| special_chars | 0,1034 | angola | 0,0014 |
| currency | 0,0954 | australia | 0,0008 |
| Undefined | 0,0672 | Insurance | 0,0005 |
| Services | 0,0649 | united_kingdom | 0,0005 |
| E-commerce | 0,0511 | mexico | 0,0004 |
| Information Technologies | 0,0430 | argentina | 0,0003 |
| Arts and Entertainment | 0,0377 | prefix | 0,0003 |
| Retail | 0,0359 | mozambique | 0,0002 |
| personalization | 0,0325 | honduras | 0,0001 |
| Other | 0,0325 | malta | 0,0001 |
| Education | 0,0284 | chile | 0,0001 |
| emojis | 0,0257 | curacao | 0,0000 |
| Beauty/Health | 0,0218 | | |

# Experiences Results

## Appendix D

# Article

# Learning to classify a subject-line quality for email marketing using Data Mining techniques

M. Paulo[1,3], V. L. Miguéis[1], and I. Pereira[2,3]

[1]Faculty of Engineering of the University of Porto

[2]Porto School of Engineering

[3]E-goi

## ABSTRACT

In today's fast-paced digital world, companies started to replace the traditional marketing strategy with digital marketing procedures. Among the technology-intensive techniques, email marketing arose as one of the most preferred methods for companies to reach customers. Despite being one of the most cost-effective methods, it remains a challenging field due to the low rate of opened emails and the high rate of unsubscribed campaigns. The sender and the subject-line are the unique information which the recipient will see at first when receiving an email and, hence, since the customer's decision to open an email critically depend on those two unique factors, it is essential that both stand out and catch the customer attention. Therefore, this study aims to support an email campaign's editors on subject choice, based on the subject potential quality. Through data mining classification techniques and, considering both the structural and content features of a subject-line, this research intends to create a model capable of classifying a subject regarding its respective quality, from 1 to 5 stars. A data set of 140.000 subjects is used to validate the proposed model. Different techniques such as Random Forest, Decision Trees, Neural Networks, Naive Bayes, Support Vector Machines, and Gradient Boosting were applied. The Random Forest technique revealed to be the one generating significantly better classification results.

## Keywords

Digital Marketing, Email marketing, Predictive Modeling, Data Mining, Artificial Intelligence, Machine Learning

## 1. INTRODUCTION

Nowadays, with so many people connected to the Internet, it is understandable why digital marketing is overpowering traditional marketing. Due to the capability to reach customers in a very short time, the easy recovery of investments and the effectiveness of this channel for customer retention, consumer awareness, and customer conversion, email marketing remains one of the most preferred and powerful methods of contact by firms [6].

However, this marketing tool effectiveness is being challenged, since every modernized company is adopting it, resulting in a low rate of opened emails [8].

In fact, consumers receive, each day, a massive amount of emails, which increases the competition for their limited attention [8].

Every email campaign is built around very specific objectives that a business intends to accomplish. A necessary precondition for any of those distinct emails to be considered successful emails is to be opened and read by the user. Therefore, having high open rates becomes critical to business success [4].

Focusing on catching the customer's attention when opening the mailbox, and considering that both the sender and the subject are the unique drivers of an effective campaign result, this study seeks to help email campaigns' editors increasing subject quality and therefore, improving open rates regarding the campaigns they are designing.

Even though endless website pages continue to teach users on how to improve subject-lines quality, they are merely a descriptive analysis. The need for a tool which evaluates the subject, instantly, is the motivation behind this study.

In collaboration with a Portuguese company which offers a Multichannel Marketing Automation Platform solution and, making use of the data collected and available for use, it is intended to:

1. Predict the subject quality for new email campaigns to be sent through the platform, taking into account not only structural but also subject content features which could impact the overall subject quality. Hence, by improving the subject quality, we seek to achieve higher open rates.

2. Compare performance results regarding several data mining techniques, specifically Naive Bayes, Random Forest, Decision Trees, Support Vector Machines, Gradient Boosting, and Neural Networks.

This work is described in several sections. Section 2 details related researches and studies on this topic. Section 3 describes the available data and the steps taken to produce new valuable variables. In section 4, the methodology employed and the performance criteria applied are described, along with a brief explanation of the most relevant Data Mining classification techniques. Section 5 presents all the performance results for the developed experiments. Finally, the conclusions of the work are presented and ideas for future work are suggested.

## 2. RELATED WORK

Since an email campaign is only considered a successful email if it is open and read by the receiver, big efforts have been made in order to explain and increase the number of recipients who opened and read the email. Moreover, in order to improve email engagement metrics, specifically in Email Marketing, a few studies have been developed over the years.

The following studies were divided into Primary Analysis and Secondary Analysis. Whereas a Primary Analysis involves primary data collected for the purpose of addressing the specific research problem, a Secondary Analysis handles secondary data which is data collected with a purpose not necessarily related to the current research problem.

## 2.1 Primary Analysis

According to prior research, when scanning the inbox, people prioritize some emails over others [22]. With the overwhelming volume of emails that people receive each day, it becomes impractical to pay attention to all that content. Wainer et al. [22] attempt to reveal the reasons why people choose to open certain emails over others, suggesting that driving attention to an email is a function of the inferred utility of message content and curiosity.

Sahni et al. [16] empirically studied the content of an email, specifically, quantifies the role of personalized content as customer-specific information. As firms usually have information about consumers, such as the customer name, it can potentially be incorporated into the marketing message to personalize it. Non-informative content was proven to be valuable in garnering customer attention and interest, increasing the probability of the recipient to opening and reading it.

Feld et al. [8] developed empirical studies on the effects of direct mail design characteristics such as visual design, the color, illustrations, sender identity, sender's name, logo and personalization on the open rate and keeping rate. The study concludes that the design characteristics that potentially generate curiosity can widely influence the open rate, offering specific guidelines for email campaigns editors, such as using colors with caution, using sender identity with care and using personalization as a differentiation factor. Both Feld et al. [8] and Sahni et al. [16] concluded that personalization can contribute to the increase of campaigns success.

## 2.2 Secondary Analysis

The secondary analysis was divided into Non-linguistic and Linguistic analysis. Whereas in the first one, only the morphology of the words was studied, in the second one, the meaning of words was also taken into consideration.

### *Non-linguistic analysis of subject-lines*

The Apostolopoulos [2]'s approach towards predicting the open rate was based on patterns found on the subject-lines of the most opened emails and disregarding the meaning of words. Therefore, the author attempts to foretell the number of openings through a Random Forest Regressor algorithm, considering the subject-line morphology and also the importance and value of the email sender based on the performance history.

### *Linguistic analysis of subject-lines*

Balakrishnan and Parekh [4] also intended to predict the open rate of an email based on the subject-line characteristics. This time focusing not only on syntactical and historical features but also on the influence of different keywords in the subject-line. In fact, it is not right to affirm that all the keywords in a subject-line have the same influence on the open rate. Using a Random Forest Regressor algorithm, the authors could see how different keywords on the same subject were performing, assigning a score to each of them.

Apostolopoulos, on his second study [3], proposes a different approach to the prediction of the open rate. This time, measuring the impact of each word and also the impact of the pattern, basically the position of the word inside the subject-line through the application of a Gradient Boosting Regressor. An entity recognition on the subject-line was run in order to remove stop words, numbers, punctuation, currency, percentages, and emojis, and then, the remaining words were stemmed. Hence, having only the root words[1], without the prefixes and suffixes, and a short wordlist, each one of the root words was scored.

According to Miller and Charles [13], the subject-line and email address of the sender are the main deciding factors for one to open the email or leave it. The authors analyzed the email subject-lines in a psychological point of view, studying their "effect in a person when he/she reads it and the decision he/she makes to open that email or neglect it" [13]. The authors studied the emotional effect of the subject-line on the open rate, through a lexicon-based approach, where the adjectives in the subject-line were classified in one of nine emotion categories between trust, joy, surprise, anticipation, peace, sad, anger, fear and disgust. A Support Vector Machine classifier was used to classify the subjectiveness of the email subject-lines as a fact or opinion and the opinion induced when reading an email as positive, negative or neutral. The presence of adjectives, verbs, a localization or an organization name, the occurrence or non-occurrence of certain terms as well as the number of words and characters were analyzed regarding the act of opening an email or rejecting it.

## 2.3 Related Frameworks

Additionally to the above-described research, newly technologies in this field are introduced and described. Believing that a successful email campaign starts with a subject-line that grabs the attention of the subscribers, the well known American company called MailChimp[2] developed a subject-line researcher tool that is able to show the effectiveness of different keywords. When searching for a keyword, the platform will compare that term to all subject-lines ever sent, listing related terms and phrases with the respective effectiveness of the word, in a 5-star rating system. This tool does not analyze a subject. Instead, it receives a list of words, for example, *discount and computer*, and returns, not the quality of each of the given words but a list of related terms and the respective effectiveness. So in this case, would return for example, the word *promotion*, and the effectiveness of that word in the past sent email campaigns.

Another existent tool is the CoSchedule Email subject-line Tester[3] which is able to classify a subject quality, from 1 to 100, also providing clear feedback in order to optimize the subject-line even further. Taking into consideration words that increase openings, words that decrease openings, the effects of the case, the presence of numbers, character count, word count, and emoji count, this tool helps editors optimizing subjects line before sending it to the subscriber list.

[1] For example, the root word of waiting and waited, is wait.

[2] https://mailchimp.com/

[3] https://coschedule.com/email-subject-line-tester

**Table 1: Data set variables collected by the service provider**

| Variable | Description |
|---|---|
| country | Company's country |
| sector | Company' business sector |
| subject | Campaign's subject field |
| unique_open_rate | No. of emails opened for the first time by individual users |

Although being an innovative tool, it does not interpret the words contained in the subject. Indeed, checks if the subject contains any word of a pre-defined list of words. For example, the words "% off" or "voucher" are contained in the list of words that increase openings. The words "cash" and "www" are examples of words that decrease openings.

## 2.4 Conclusions

To sum up, even though the efforts made in the investigation field to influence customer behavior when receiving an email campaign, the literature on this topic is very incipient.

Regarding the literature that uses secondary data, the available studies do not evaluate the impact of using personalized messages and lack on investigating the influence of the industry to which the campaign is sent. Also, these studies fail on undervaluing the impact of the country to where the email is sent. Indeed, the median of the OR is absolutely different depending on the country [24]. Finally, the utilization of machine learning techniques to support these studies is still very limited to a small set of techniques.

Solutions in the market continue to be scarce. Most of the websites that are seeking to help customers on choosing the best subject-line offer merely descriptive guidelines and, hence, most of them, end up easily outdated. Moreover, none of the actual technologies incorporate Machine Learning and there is where this project distinguishes itself. To operate in this the fast-changing industry, this technology should be automatically updated, with new data and, hence, with the latest trends.

# 3. CASE STUDY AND DATA

The company used as case study offers a Multichannel Marketing Automation Platform solution for distinct business sectors, such as Education, Marketing, Retail, Hotels, among others. This company (from now on referenced as *service provider*) provides a service for their clients that are, in this specific case, companies (henceforward referred to as *company*) that benefit from the email marketing automation tools in order to communicate to their respective customers (from this time forth referenced as *recipients* or *customers*). Despite working with several well-renowned companies from around the world, the service provider has much more incidence in Portugal, Brazil, Colombia, Spain and most of Latin America.

The data made available for this research is comprised of 140.000 email campaigns. The email campaigns were required to have, at least, 100 subscribers and, because campaigns take time to work and reach the recipients, only campaigns sent over one week before data collection were considered. This way, we guarantee it includes exclusively campaigns with viable performance reports.

Table 1 lists the information provided for each email campaign, for the development of this study.

Considering the continuous value for the *unique_open_rate*, and the classification problem at hand, the first step was to convert the open rate into a discrete value. Therefore, a binning algorithm (equal-width binning) was employed to establish 5 levels of quality. The following five elected classes represent a 1 to 5 stars scale rate.

$$quality\_class = \begin{cases} if & OR \geq 22.4 & ,5 \\ if & 22.4 > OR \geq 13.7 & ,4 \\ if & 13.7 > OR \geq 9.37 & ,3 \\ if & 9.37 > OR \geq 5.41 & ,2 \\ else, & 1 \end{cases} \tag{1}$$

The above equation 1 lists the five classes which were created through the binning technique.

## 3.1 Data Cleaning

The primary focus of the data cleaning phase is to handle missing data, noisy data and remove outliers, minimizing duplication and computed biases within the data.

Beginning with missing values, many samples were found to have the sector column without any value. Those cases were related to users who don't specify any business sector in their profile account. In order to fix that issue, samples with empty sector column were filled with the "undefined" value.

Concerning the diverse languages across the data set and, since it could be challenging to address efficiently so many different vocabularies and so many distinct countries, the most relevant marketplaces were filtered out, by developing a simple application aiming to identify the most prevalent languages. Employing the langdetect[4] library to detect the subject language and, studying all the available data, we could sustain the original idea that the service provider's most significant marketplaces are Portugal, Brazil, Colombia, Spain and most of Latin America.

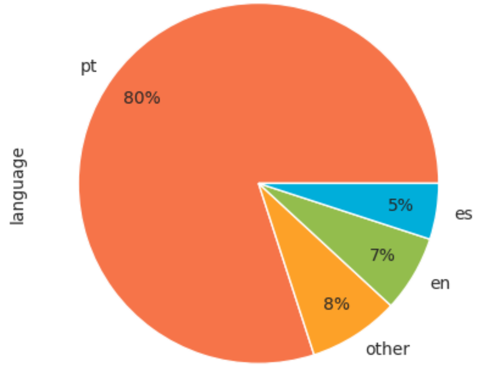


**Figure 1: Most common languages in the case study data set**

As shown in Figure 1, the most widespread languages among the subjects are Portuguese (pt), English (en) and Spanish (es). In fact, 80% of the subjects were written in

[4] https://pypi.org/project/langdetect/

Portuguese, 7% in English and 5% in Spanish. The other 8% correspond to 28 different languages, including, for example, Dutch, French, Italian, and Catalan. Therefore, and as recommended by the service provider, only the three most relevant languages will be considered to tackle this problem.

Lastly, outliers were found regarding the unique open rate value. Some data points differed significantly from other observations, containing fraudulent open rates, with values greater than 100% and, hence, were excluded.

## 3.2 Feature Engineering

Considering the subject, both structural and content analysis were made. As a result, new features were created, acting as inputs of the final model.

### *Structural Features*

The structural assembler was developed to care about the structure and size of the subject-line, considering the number of words, number of characters, the use of upper case, punctuation, special characters, numbers or currency, the presence of emojis and the personalization of the message.

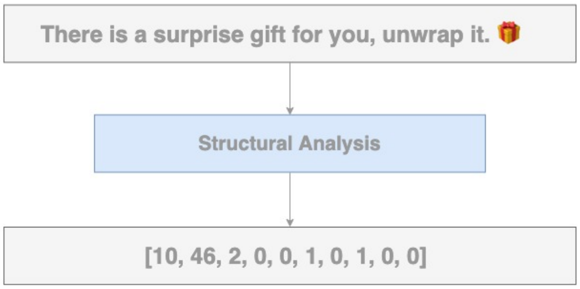


**Figure 2: Structural Analysis**

As shown in Figure 2, **the structural assembler** receives a subject and outputs a list with:

*[no. of words* ***(integer)****, no. of characters* ***(integer)****, case percentage* ***(integer)****, presence of punctuation* ***(boolean)****, presence of prefixes* ***(boolean)****, presence of emojis* ***(boolean)****, presence of personalization* ***(boolean)****, presence of special chars* ***(boolean)****, presence of numbers* ***(boolean)****, presence of currency* ***(boolean)****]*

Please note that emojis are treated as words, as they represent a set of characters. Summarizing, prefixes are abbreviations which are added by the author to the subject-line in order to improve the ability to prioritize, review, and process new inbox messages. Some examples of prefix codes could be, for example, *"FWD"* or *"ASAP"*. Personalizing the subject-line means adding subscriber unique information such as the name or email address, providing messages perfectly tailored to subscribers' needs. The platform provided by the case study service provider allows the company to address the subscribers by their personal information, under personalization codes such as *!fname* (First Name), *!lname* (Last Name) or *!email* (Email). Regarding the currency analysis, the structural assembler is capable of identifying currency by the symbol (€, $, £, R$) or by the abbreviation, for euro, real, dollar or pound sterling. With respect to the case percentage feature, it gives a percentage of the subject in upper case.

### *Content Features*

In order to study the quality of the set of words chosen, two distinct approaches were employed: Past Performance and Bag Of Words.

Before explaining the abovementioned approaches, and considering that the preprocessing phase was transversal across both strategies, the following steps summarize the sequence of operations taken to perform the cleaning and subsequently subject analysis:

- *Clean Subject:* Converts a subject to lower case, removes personalization codes, punctuation marks, special characters and emojis, keeping the accent marks.

- *Identify language:* The subject language was identified through the referred Python Library called langdetect[5]. Accordingly to the aforementioned research results on finding the most common languages across the past subject-lines, only the Portuguese, English and Spanish packages were made available and, hence, all the other languages were discarded. In case of being impossible to identify the language, the respective country language was used. Taking into account the available packages (English, Portuguese and Spanish), the following countries were accepted: Portugal, Honduras, Angola, Curaçao, Spain, Brazil, Colombia, Peru, Mozambique, United States Of America, Australia, Mexico, Malta, United Kingdom, Argentina, and Chile.

- *Discard Stop Words:* Depending on the language, different stop words were discarded. To identify them, the Spacy [6] library was used. Filtering out stop words before or after the processing of natural language data is a very common strategy to save time and space in these type of projects. As stop words correspond to common and irrelevant words of the vocabulary [7], they can be ignored.

- *Lemmatizing:* Depending on the detected language, different lemmatizers were applied. Concerning lemmatization, the Spacy [8] library was used. A lemma is the canonical form or dictionary form of a set of words [9].

Figure 3 summarizes the stated steps.
Getting deeper into the two distinct strategies:

1. **Past Performance:** Two new variables were created containing information regarding the subject's lemmas past performance. The *lemmas_past_performance* variable represents the quality of the words contained in the subject, predicated on the past sent subjects, and contains values between 1 and 5. The second feature, *nr_lemmas*, as the name suggests, holds information about the number of lemmas found.

   In order to do so, a dictionary was created, in which, for each lemma, the following information was available: *[lemma, count, average of quality from all past*

[5] https://pypi.org/project/langdetect/
[6] https://spacy.io/
[7] For example, the word "the".
[8] https://spacy.io/
[9] For example, the words running, runs and ran all have run as the lemma.

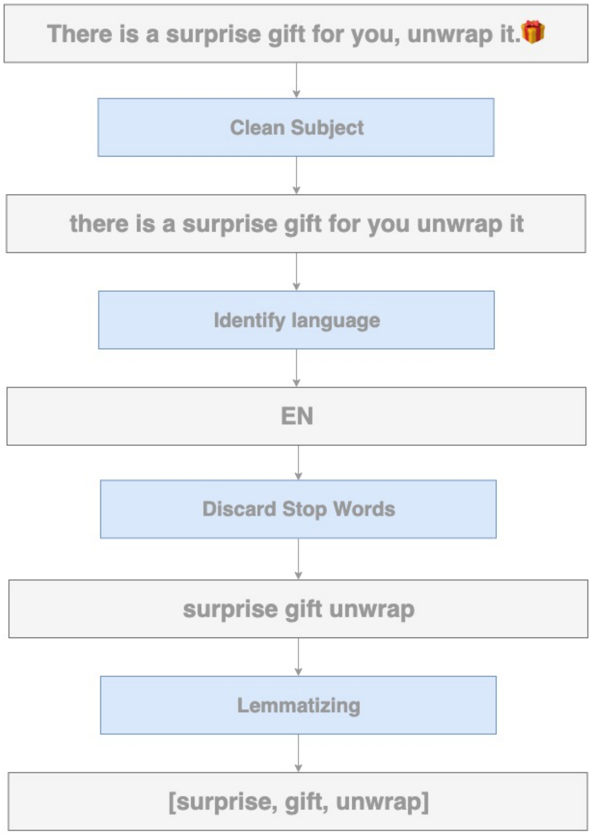


**Figure 3: Subject pre-processing steps**

*sent subjects]*. Subsequently to the dictionary creation, the *lemmas_past _performance* variable was calculated. Different strategies were attempted:

- **Weighted average (W_AVG):** calculated with the past quality of each of the lemmas contained in the subject and also with the number of appearances of each of these lemmas in emails history. The information regarding the number of appearances of each lemma is contained in the dictionary and here acts as the weight. This approach guarantees that, for example, if a word was used only once, it will impact differently compared to frequently used words.

$$weighted\_average = \frac{\sum_{i=1}^{n} (x_i * w_i)}{\sum_{i=1}^{n} w_i} \qquad (2)$$

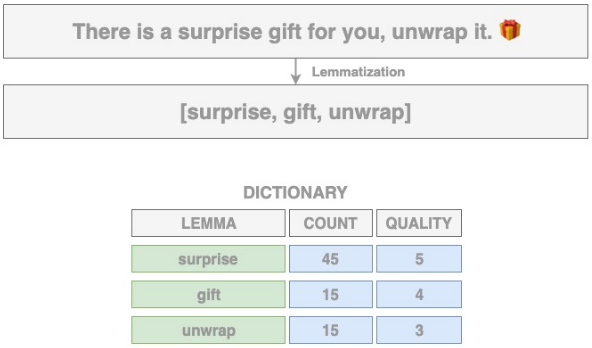
  For $n$ lemmas filtered from the subject, the weighted average is the quality of the lemma, $x_i$, multiplied by the weight of the lemma $w_i$, here known as the number of appearances of the lemma, divided by the sum of all the $n$ lemmas weights.

- **Average (AVG):** naive average, discarding the number of appearances of each of the lemmas.

$$average = \frac{1}{n} * \sum_{i=1}^{n} x_i \qquad (3)$$

  For $n$ lemmas filtered from the subject, the average is the sum of the quality of each lemma $w_i$, divided by the number of lemmas.

- **Maximum (MAX):** subject-lines need to quickly grab the recipient's attention. When opening the mailbox, recipients spend just a small fraction of a second evaluating email subject fields. According to Duggan et al. [7], "skimming" a text means reading a text quickly to get a general idea of meaning and is increasingly common in our information-rich time-limited society. The author concluded that readers focus on important information when skimming. Hence, given the small fraction of time spent in analyzing the subject-line, considering that a customer perceives only one word of the subject instead of all words, a different approach was tested, but this time considering just the **word that stands out**. Thus, the value of the variable *lemmas_past_performance* was calculated by the maximum quality value of the lemmas. Accordingly, if a subject contains a lemma of 5 stars quality, it is believed that the subject semantic quality is also 5 because the reader, subconsciously, ignores all the other lemmas.



DICTIONARY

| LEMMA | COUNT | QUALITY |
|---|---|---|
| surprise | 45 | 5 |
| gift | 15 | 4 |
| unwrap | 15 | 3 |

**Figure 4: Example of Past Performance strategy analysis**

**Table 2: Example of semantic analysis for each different approach**

| Approach | lemmas_past_performance |
|---|---|
| $W\_AVG$ | $\frac{(45*5)+(15*4)+(15*3)}{45+15+15} = 4.4$ |
| $AVG$ | $\frac{(5+3+4)}{3} = 4$ |
| $MAX$ | $Max(5,3,4) = 5$ |

Considering the example in Figure 4, Table 2 presents, for each approach, the *lemmas_past_performance* variable value.

2. **Bag of Words:** A Bag of Words (BOW) approach was employed and tested due to being commonly used in natural language processing and document classification [9]. The BOW technique extracts features from text documents, that can be used in machine learning algorithms. Essentially, is responsible for creating

a vocabulary of unique words contained in all documents, completely disregarding the order in which those words appear.

Taking, as example, the subject in Figure 4, the generated vocabulary would be *[surprise, gift, unwrap]*. Following the vocabulary creation, and as a new input appears, the count vectors are generated. Considering the following new subject *"I have a gift for you"* and the vocabulary *[surprise, gift, unwrap]*. The generated vector would be *[0, 1, 0]*, since only the word "gift" is contained in the vocabulary.

However, in this particular case, instead of the simple count of the words within the subject, and because it is essential to take in consideration all the other sent subjects, the Term Frequency-Inverse Document Frequency (TF-IDF) calculation was employed. TF-IDF measures relevance, not frequency. For that reason, word counts are replaced with TF-IDF scores across the whole data set. TF-IDF not only measures the number of times a word appears in a subject but then, it gives more importance to less common words and a lot of value to the rare ones [1].

## 4. METHODOLOGY

As mentioned before, this study aims at predicting the potential quality of an email marketing campaign subject. Five quality levels are considered, making this a multi-classification problem. To predict the performance class, several data mining techniques are explored.

Data Mining algorithms are biased to look for different types of patterns, and because there is no learning bias across all situations, there is no one best algorithm [11]. Each algorithm requires different constraints and distinct weights. Choosing the appropriate hyperparameters also plays a crucial role in the success of the model and, therefore, the parameters need to be tuned in order to achieve the optimal model, the one which minimizes the prediction error. Thus, the proposed methodology involved parameters tuning.

The modeling phase concerns the selection of the methodology and the subsequent algorithm. Since depending on the selected features, more or less accurate models can be generated, different experiments were performed with different methods and features. Recapping the different applied methodologies:

- **Experiment 1 - Structural Analysis:** Only the subject structure, the sector, and the country were considered.
- **Experiment 2 - Structural and Content Analysis:** Additionally to the structure and company's information (country and sector), also the content was considered.
  - **Experiment 2.1:** *Lemmas Past Performance* feature was calculated through the Weighted Average.
  - **Experiment 2.2:** *Lemmas Past Performance* feature was calculated through the Average.
  - **Experiment 2.3:** *Lemmas Past Performance* feature was calculated through the Maximum quality.
- **Experiment 3 - Structural and Content Analysis with Bag Of Words:** BOW with a maximum of 4% of the vocabulary, which includes the words contained in all past subjects, followed by dimensionality-reduction. In order to do that, Principal Component Analysis (PCA) was applied. PCA [21] is a mathematical procedure for dimensionality reduction in data, which transforms a number of correlated variables into a smaller number of uncorrelated variables called principal components.

Table 3 summarizes all the generated features and the respective context of creation.

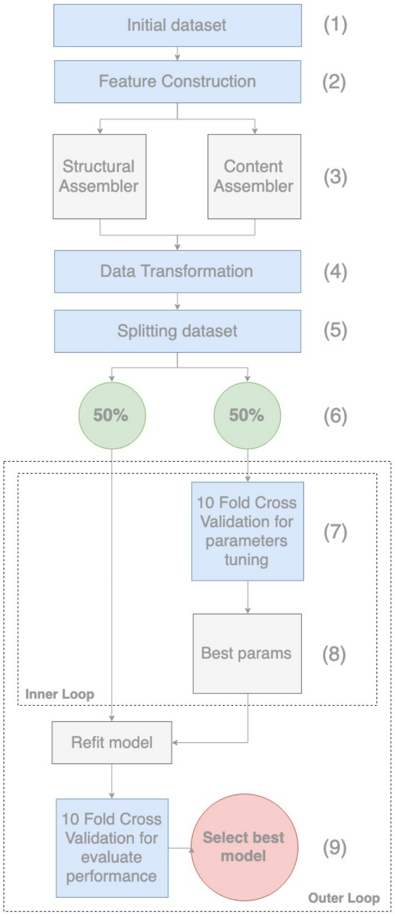


**Figure 5: Proposed Model**

Figure 5 illustrates the carried out methodology. After the feature construction, pictured on step 2, in which both structural and content-based features were created, follows the data transformation step. Since the data type of an attribute *(nominal or categorical - numerical or ordinal)* affects the way algorithms identify patterns and relationships between attributes, it was mandatory to transform the data. In order to bring all the numerical features to the same level of magnitudes, **Min-Max** normalization was employed [17]. The categorical features, specifically the sector and country, were **binarized**. In step 5, the data was split into two equal parts using stratification, providing stratified randomized folds to guarantee the same percentage of samples for

**Table 3: Available Features**

| Feature | Type | Nr$^{o}$ of categories | Context | Exp. 1 | Exp. 2.1 | Exp. 2.2 | Exp. 2.3 | Exp. 3 |
|---|---|---|---|---|---|---|---|---|
| Country | Nominal | 13 | Company's Information | X | X | X | X | X |
| Sector | Nominal | 21 | Company's Information | X | X | X | X | X |
| Nr$^{o}$ words | Continuous | - | Structure | X | X | X | X | X |
| Nr$^{o}$ chars | Continuous | - | Structure | X | X | X | X | X |
| Case Percentage | Continuous | - | Structure | X | X | X | X | X |
| Punctuation | Binary | - | Structure | X | X | X | X | X |
| Prefix | Binary | - | Structure | X | X | X | X | X |
| Emojis | Binary | - | Structure | X | X | X | X | X |
| Personalization | Binary | - | Structure | X | X | X | X | X |
| Special_chars | Binary | - | Structure | X | X | X | X | X |
| Numbers | Binary | - | Structure | X | X | X | X | X |
| Currency | Binary | - | Structure | X | X | X | X | X |
| Lemmas Past Performance (AVG) | Continuous | - | Content: Past Performance |  | X |  |  |  |
| Lemmas Past Performance (W_AVG) | Continuous | - | Content: Past Performance |  |  | X |  |  |
| Lemmas Past Performance (MAX) | Continuous | - | Content: Past Performance |  |  |  | X |  |
| Nr$^{o}$ lemmas | Numerical | - | Content: Past Performance |  | X | X | X |  |
| List of TF-IDF* | Numerical | - | Content: Bag Of Words |  |  |  |  | X |

* *For each subject, the BOW approach produces a vector of tf-idf frequencies*

each class. With half the data, a set of modeling techniques were applied and their parameters were calibrated to optimal values, the hyperparameters tuning, represented on step 8. With the remaining 50% of the data, the error was estimated and the overall performance was analyzed towards the selection of the final model, on step 9.

## 4.1 Data Mining algorithms

Six different algorithms were studied. This section presents a brief description of each select algorithm.

### 4.1.1 Support Vector Machines (SVM)

Support Vector Machines [18] is an algorithm based on the concept of decision planes that define decision boundaries called hyperplanes. One unique aspect of this technique is that the decision boundary is defined using only a subset of the training examples, known as the support vectors [19]. Consider the two classes, i.e. *square* and *circle*. A hyperplane is a frontier such that all the squares reside on one side of the hyperplane and all the circles reside on the other side, taking into account that the selected hyperplane is the one that maximizes the margin between the two classes with the help of support vectors.

### 4.1.2 Naive Bayes (NV)

The Naive Bayes [12] is a probabilistic machine learning algorithm based on the Bayes' Theorem conditions on the independent and equal contribution of each feature to the outcome. The Bayes' Theorem determines the probability of an event occurring given the probability of another event has already occurred. Mathematically, the Bayes theorem is stated by the following equation:

$$P(Y|X) = \frac{P(X|Y) * P(Y)}{P(X)} \tag{4}$$

In equation 4, Y acts as the outcome to predict, and X as the evidence or features. The Bayes Rule is a way of predicting P(Y|X), in other words, the probability of Y given X, from P(X|Y), known from the training data set. In the case of dealing with a multi-classification problem, for each class Y, P(Y|X) is calculated and the class with the highest probability is **assigned to the observation**.

### 4.1.3 Decision Tree Classifier (DT)

Based on the concept of divide and conquer and, through a set of if-then-else operations, a problem is solved through a tree representation where, in specific for classification problems, the leaves represent the output class and the branches correspond to the features. The basic idea of a decision tree [15] is to go from the root, including all observations, to conclusions about the item's target value, represented in the leaves, through the satisfaction of the conditions expressed in the branches.

A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values, usually obtained through the entropy and information gain calculation. The entropy measures the heterogeneity of a specific sample and the information gain estimates the reduction on entropy resulting from the division of the sample based on one attribute. Constructing a decision tree is all about continuously finding the attributes which provide the highest information gain until all the samples belong to the same class.

### 4.1.4 Ensemble Methods

Ensemble methods [14] are known for improving prediction accuracy by aggregating the predictions of multiple predictors and can be classified into bagging or boosting algorithms. Although a single decision tree may not be a very good predictor, very powerful models can be obtained by combining the results of several decision trees classifiers.

### Random Forest (RF)

This algorithm [5] is an example of ensemble methods based on bagging, working for both classification and regression. Operates with a bag of decision trees, each one considering a random subset of features and having access to a random set of training data points. The output value becomes the mode of the classes, regarding classification, and the mean prediction in the case of regression, of all the individual trees. This algorithm achieves robust overall predictions because of the diversity of results and since the output is not swayed by a single atypical data source, also fixing the decision tree algorithm problem of overfitting to their training set.

### Gradient Boosting (GB)

Gradient Boosting [10] is a boosting ensemble method. Whereas in bagging algorithms, independent predictors are combined using some model averaging techniques, in boosting algorithms, predictors are not grouped independently, but sequentially. This technique employs the logic in which the subsequent predictors learn from the mistakes of the previous predictors. An example of a boosting model is the AdaBoost technique, which focuses on classification problems and aims to convert a set of weak classifiers into a strong one.

### *4.1.5 Artificial Neural Networks (ANN)*

An Artificial Neuron Network is a computational model that imitates the biological Neural Networks of the human body. An ANN is composed of a set of connected neurons organized in layers: The Input Layer is responsible for bringing the initial data into the system, to be processed by the subsequent layers of the artificial network. The Hidden layer is a layer where artificial neurons take in a set of weighted inputs and produce an output through an activation function and the Output layer is responsible for producing given outputs for the program.

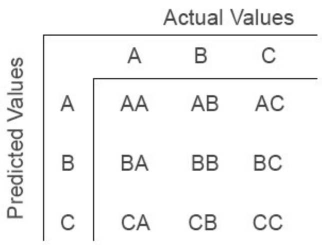
In this study, we adopt a Multilayer ANN because of being often applied to supervised learning due to its capability for solving complex classification and regression problems. It contains one or more hidden layers and can learn nonlinear functions, using a backpropagation method to train the network. Training involves adjusting the parameters, or the weights, of the model in order to minimize the overall error. It is also called feed-forward because, in this specific network, the nodes in the first layer are connected only to the nodes in the next layer. Initially, with the backpropagation method, all the edge weights are randomly assigned. For every input in the training data set, the Multilayer ANN is activated and its output is observed. The given output is compared with the desired one, and the error is propagated back to the previous layer. This error is recorded and the weights are adjusted accordingly. This process is repeated until the output error is below a predetermined threshold.

The activation function of a node defines the output of that node given a specific input or a set of inputs. There exist several activation functions that can be applied depending on the problem at hand, for example, the sigmoid function, Tan-h, Softmax, ReLU or Leaky ReLU. An activation function allows the model to produces a result (target variable, class label, or score) that varies non-linearly with its explanatory variables [25].

## 4.2 Evaluation criteria

As introduced in Figure 5, 50% of the data was used for tuning the hyperparameters. With the parameters tuned for each selected model, the remaining half of the data was used to measure the performance of the proposed prediction models. To achieve this, we employed Nested Cross Validation, each one with 10 folds. As shown in Figure 5, at first, an inner cross-validation loop to tune the hyperparameters and select the best model. Second, an outer cross-validation to evaluate the model selected by the inner cross-validation.

Regarding the appropriate way to evaluate and compare the models, from the business perspective, it could be extremely risky to misclassify a poor subject as a brilliant subject since the company creates high expectations, leading to disappointment.



| | | Actual Values | | |
|---|---|---|---|---|
| | | A | B | C |
| Predicted Values | A | AA | AB | AC |
| | B | BA | BB | BC |
| | C | CA | CB | CC |

**Figure 6: Confusion Matrix**

Considering the multi-class confusion matrix, on Figure 6, each row of the matrix represents the instances in a predicted class and each column the instances in an actual class. {A, B, C} refer to the three classes. For example, concerning class A, AA refers to the number of observations classified correctly, where A was classified as A; AB refers to the number of observations belonging to B class and classified as A called the False Positives, whereas BA refers to the number of observations belonging to A class and classified as B, also known as False Negative. Almost all the evaluation metrics depend on these values and terminology.

The **average accuracy** is the ratio of correct predictions to the total number of predictions, how often is the classifier correct. **Precision** metric expresses how precise the model is out of those predicted positives, thus how many of them are actual positives. **Recall** calculates how many of the actual positives our model capture through labeling it as Positive [20]. Explaining both metrics for the matter at hand, with a focus on class 5 stars:

- Recall expresses how many subjects were classified as 5 considering all the existent subjects with 5 stars quality class.
- Precision reveals from all subject classified 5 stars quality, how many subjects were correctly labeled.

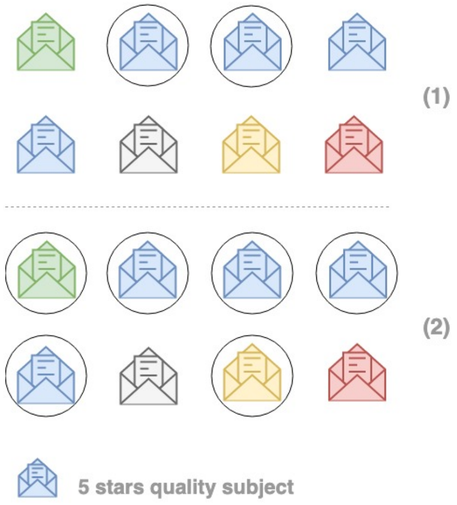


**Figure 7: Recall and Precision applied to the problem**

In Figure 7, particularly in sub-figure (1), there is a perfect precision since all the picked 5 stars quality subjects are in fact from class 5 stars, however poor recall since a substantial amount of subjects were not found to belong to class 5 stars. In sub-figure (2), a perfect recall since the classifier detected all 5 stars quality subjects but a poor precision due to a large number of false positives.

Regarding the business goals, the most appropriate model balances the Recall and Precision results for each of the classes. **F1 score** is defined as the harmonic mean between precision and recall and, for a specific class $n$, it can be calculated through the following equation:

$$F1score_n = 2 \times \frac{precision \times recall}{precision + recall}$$

Given the multi-class problem at hand, the **F1 score** was measured as the **weighted** average of each of the five classes:

$$F1score = \frac{\sum_1^n F1score_n \times No.Samples_n}{\sum_1^n No.Samples_n}$$

Where *F1score_n* is the *F1 score* of class *n* and *No.Samples_n* is the number of samples of class $n$, with n being the number of classes. The **F1 score** was the scoring metric applied to the parameters tuning. To the final algorithm selection, both the F1 score and the Accuracy were used, although always weighing the business goals and expectations.

# 5. RESULTS AND DISCUSSION

This chapter lists all the distinct performed experiments, each one corresponding to a different approach and set of features. The achieved results for each of the six algorithms, mentioned in the previous chapter, are shown and described.

## 5.1 Experiment 1

As explained in Section 4, the first experiment aims to analyze the subject concerning the country, business sector, and structure, completely disregarding the meaning and the quality of the words included in the subject.

**Table 4: Experiment 1 performance results**

| | RF | ANN | NB | GB | SVM | DT |
|---|---|---|---|---|---|---|
| F1 Score | 60.4% | 56.3% | 23.1% | 45.6% | 58.2% | 31.7% |
| Accuracy | 60.6% | 56.8% | 32.0% | 47.0% | 58.3% | 41.6% |
| Precision 1 | 69% | 63% | 31% | 53% | 65% | 43% |
| Precision 2 | 51% | 47% | 17% | 38% | 48% | 0% |
| Precision 3 | 50% | 47% | 45% | 41% | 48% | 38% |
| Precision 4 | 55% | 51% | 29% | 37% | 54% | 22% |
| Precision 5 | 76% | 75% | 49% | 57% | 76% | 45% |
| Recall 1 | 72% | 70% | 91% | 67% | 69% | 82% |
| Recall 2 | 51% | 48% | 0% | 26% | 51% | 0% |
| Recall 3 | 47% | 45% | 26% | 42% | 47% | 55% |
| Recall 4 | 53% | 46% | 41% | 30% | 47% | 2% |
| Recall 5 | 80% | 75% | 3% | 70% | 76% | 68% |

The performance of the proposed model for each of the six different algorithms is synthesized in Table 4. Analyzing the obtained results, the Random Forest appears to be the algorithm ensuring better predictive results, with 60.6% of Accuracy and 60.4% of F1 score. The worst result was obtained by using Naive Bayes, with 23.1% of F1 score and 32.0% of Accuracy.

The confusion matrix represented in Figure 9 is the result of the classification performed with the Random Forest. From the analysis of this matrix and of the recall and precision values presented in Table 4, we can conclude that the model can better classify edge classes, specifically classes 1 and 5, than the other three intermediate classes, i.e. class 2, 3 and 4. From all the intermediate classes, class 3 is the hardest to predict.

## 5.2 Experiment 2

The second experiment bears on the country, business sector, structure and also on the content features. Totally disregarding the order in which the words appear in the subject, this experiment studies different approaches to calculate the quality of the lemmas, in order to identify the one which better estimates the subject words quality.

### 5.2.1 Experiment 2.1

Experiment 2.1 calculates the variable *lemmas_past_per_formance* through the weighted average. As a result, the calculation of the past performance of the lemmas depends on the past quality of each lemma and also on how often each lemma was used in emails history.

**Table 5: Experiment 2.1 performance results**

| | RF | ANN | NB | GB | SVM | DT |
|---|---|---|---|---|---|---|
| F1 Score | 61.7% | 57.5% | 24.0% | 50.2% | 60.1% | 35.7% |
| Accuracy | 62.1% | 57.7% | 32.7% | 51.1% | 60.1% | 40.2% |
| Precision 1 | 70% | 64% | 32% | 57% | 69% | 48% |
| Precision 2 | 53% | 48% | 24% | 42% | 51% | 32% |
| Precision 3 | 52% | 48% | 44% | 44% | 50% | 30% |
| Precision 4 | 57% | 54% | 29% | 43% | 54% | 0% |
| Precision 5 | 76% | 75% | 54% | 64% | 77% | 53% |
| Recall 1 | 74% | 68% | 89% | 69% | 69% | 50% |
| Recall 2 | 51% | 51% | 0% | 38% | 50% | 50% |
| Recall 3 | 51% | 46% | 28% | 38% | 49% | 32% |
| Recall 4 | 54% | 47% | 43% | 38% | 54% | 0% |
| Recall 5 | 81% | 78% | 3% | 73% | 77% | 69% |

Table 5 illustrates that the Random Forest Classifier remains the model generating better results. This time, achieving 61.7% of F1 Score and a 62.1% of Accuracy. Once again, Naive Bayes was the model obtaining worst results, with only 24% of F1 score and 32.7% of Accuracy. Similarly to what was concluded in Experiment 1, class 5 remains the easiest to predict, with 76% of Precision and 81% of Recall, followed by class 1, with 70% of Precision and 74% of Recall.

When compared to the first experiment, this one achieves better and more accurate results, with a positive difference of 1,3% regarding the F1 Score. Accordingly, it is plausible to conclude that the past performance of the words contained in the subject is relevant to predict the overall subject quality.

### 5.2.2 Experiment 2.2

Experiment 2.2 calculates the variable *lemmas_past _performance* using the average of the quality of each lemma contained in the subject.

Table 6 shows that, once again, Random Forest Classifier is the model generating better results, this time with 62.2% of F1 score and 62.4% of Accuracy. Naive Bayes was the model with the worst performance results, with a F1 Score of 24.1% and 32.7% of Accuracy.

The achieved results showed extremely similar to the previous experiment, with class 1 and class 5 being the simplest to predict.

Figure 10 is the confusion matrix generated through the Random Forest classification on Experiment 2.2, identical

**Table 6: Experiment 2.2 performance results**

| | RF | ANN | NB | GB | SVM | DT |
|---|---|---|---|---|---|---|
| F1 Score | 62.2% | 58.8% | 24.1% | 51.8% | 60.5% | 36.5% |
| Accuracy | 62.4% | 58.9% | 32.7% | 52.6% | 60.6% | 43.4% |
| Precision 1 | 71% | 67% | 32% | 59% | 69% | 55% |
| Precision 2 | 53% | 49% | 28% | 44% | 50% | 31% |
| Precision 3 | 52% | 48% | 44% | 44% | 50% | 31% |
| Precision 4 | 57% | 54% | 29% | 43% | 55% | 0% |
| Precision 5 | 77% | 76% | 54% | 66% | 78% | 58% |
| Recall 1 | 74% | 70% | 89% | 68% | 70% | 56% |
| Recall 2 | 50% | 48% | 0% | 37% | 50% | 20% |
| Recall 3 | 51% | 50% | 28% | 42% | 50% | 59% |
| Recall 4 | 55% | 48% | 43% | 40% | 55% | 0% |
| Recall 5 | 82% | 79% | 3% | 75% | 78% | 82% |

to the one produced on the earlier experiment. The results revealed that class 1 and especially class 5 are the easiest to predict, as reflected by the small number of samples in the row and column 5, except for the position [5,5] which indicates the number of True Positives. The value of True Positives reports how well the model can predict a 5 stars quality subject. However, the cells adjacent to the diagonal are very populated, meaning that there is a significant number of subjects classified with quality exactly one class above or one class below from what was expected. In a nutshell, taking as example subjects of quality 2, there exists a considerable number of those subjects being predicted to belong to class 1 or to class 3.

Table 10 lists the most relevant features and the respective importance value.

In comparison to Experiment 2.1, even though both have achieved pretty similar performance results, Experiment 2.2 produced a higher value for the F1 Score, with an improvement of 0,5%. Therefore, the *lemmas past performance* feature measured through the naive average proved to predict better concerning the quality of the subject-line.

### 5.2.3 *Experiment 2.3*

Experiment 2.3 estimates the *lemmas_past_performance* variable by the maximum quality of the lemmas contained in the subject.

**Table 7: Experiment 2.3 performance results**

| | RF | ANN | NB | GB | SVM | DT |
|---|---|---|---|---|---|---|
| F1 Score | 61.5% | 57.8% | 23.6% | 48.8% | 59.7% | 39.1% |
| Accuracy | 61.8% | 58.1% | 32.5% | 50.0% | 59.8% | 44.2% |
| Precision 1 | 69% | 64% | 32% | 56% | 68% | 47% |
| Precision 2 | 52% | 47% | 25% | 41% | 51% | 0% |
| Precision 3 | 52% | 48% | 44% | 44% | 47% | 42% |
| Precision 4 | 57% | 54% | 29% | 39% | 55% | 29% |
| Precision 5 | 76% | 76% | 50% | 62% | 78% | 60% |
| Recall 1 | 73% | 32% | 89% | 66% | 69% | 74% |
| Recall 2 | 51% | 25% | 0% | 31% | 48% | 0% |
| Recall 3 | 49% | 44% | 27% | 42% | 52% | 37% |
| Recall 4 | 54% | 29% | 43% | 34% | 52% | 40% |
| Recall 5 | 81% | 50% | 3% | 76% | 78% | 70% |

Figure 7 demonstrates that the Random Forest is the model generating better results, with a F1 Score of 61.5% and an Accuracy of 61.8%. Naive Bayes continues the model performing worst, with a F1 Score of 23.6% and an Accuracy of 32.5%. Figure 11 evinces the model difficulty in predicting subjects of quality 3, and proves how well the model predicts edge classes, i.e. 1 and 5.

What differs this experiment from the last one, is the importance of the features. Table 11 lists the features and the respective importance value, for Experiment 2.3. While in Experiment 2.2, the *lemmas_past_performance* feature was the one impacting most in the final result with a normalized value of 1, in the current experiment, this feature loses relevance, ending up with a normalized value of 0,4689.

This experiment reached worse results comparing to Experiment 2.2, with an F1 Score reduced in 0.7%.

## 5.3 Experiment 3

Experiment 3 operates with the BOW approach in order to study the quality of the words contained in the subject. Therefore, a BOW approach with a maximum of 4% of the vocabulary was employed with a subsequent dimensionality reduction in data. The vocabulary was chosen considering the term frequency across the corpus.

When employing the BOW approach, each word of the vocabulary becomes a new feature of the model and so, space and time complexity increase exponentially. With the purpose of attenuating the dimensionality issues, PCA was used. It reduces the number of features while preserving as much information as possible. When invoking PCA, the parameters were set with the intention of giving the model responsibility to guess the most suitable feature dimension.

**Table 8: Experiment 3 performance results**

| | RF | ANN | NB | GB | SVM | DT |
|---|---|---|---|---|---|---|
| F1 Score | 60.0% | 59.6% | 35.0% | 50.0% | 44.0% | 32.0% |
| Accuracy | 60.2% | 59.5% | 36.6% | 50.9% | 46.9% | 41.6% |
| Precision 1 | 67% | 68 % | 42% | 56% | 45% | 43% |
| Precision 2 | 52% | 51% | 32% | 43% | 41% | 0% |
| Precision 3 | 57% | 49% | 37% | 43% | 40% | 38% |
| Precision 4 | 56% | 54% | 27% | 45% | 40% | 22% |
| Precision 5 | 75% | 74% | 43% | 65% | 69% | 45% |
| Recall 1 | 72% | 71% | 38% | 66% | 77% | 82% |
| Recall 2 | 46% | 48% | 31% | 39% | 20% | 0% |
| Recall 3 | 54% | 50% | 14% | 36% | 56% | 55% |
| Recall 4 | 51% | 52% | 36% | 43% | 19% | 2% |
| Recall 5 | 79% | 78% | 65% | 70% | 62% | 68% |

Figure 8 presents the performance results for each employed algorithm. Random Forest is the classification technique achieving the most accurate results, with a F1 Score of 60.0% and an Accuracy of 60.2%, and the Decision Tree and Naive Bayes are the ones producing worst results.

Even though this technique is one of the most common and easy methods used in text classification, BOW representation suffers from its intrinsic extreme sparsity and high dimensionality [26]. These unsatisfactory results can be related to the need of dealing with three different languages, i.e. pt, en, and es, since, in fact, the model evaluates not only one but three languages and thus, the number of lemmas available in each one could be insufficient. Furthermore, whereas in Experiment 2 we assign a quality to each known word, if it has already been sent at least once in the past email campaigns, in this case, only specific words have relevance in the prediction and the others are totally insignificant.

Figure 8 presents the 15 most relevant lemmas, obtained by the bag of words approach.

Figure 12 illustrates the confusion matrix obtained through the Random Forest Classification and the balanced percentage of Precision and Recall values among the classes, unlike Experiment 1. Although class 5 continues to be the easiest to predict, this time, all the classes have a similar probability of predicting the subject quality successfully.

Working with only 4% of the vocabulary was not enough to surpass the performance of Experiment 2.2, although we believe that with more vocabulary the results could have improved.

### 5.4 Discussion

Table 9 summarizes the best performance results achieved for each conducted experiment, in particular, the F1 Score and Accuracy.

**Table 9: Summary of the performance results for each experiment**

| | **F1 Score/Accuracy** |
|---|---|
| **Experiment 1** | 60.4% / 60.6% |
| **Experiment 2.1** | 61.7% / 62.1% |
| **Experiment 2.2** | 62.2% / 62.4% |
| **Experiment 2.3** | 61.5% / 61.8% |
| **Experiment 3** | 60.3% / 60.6% |

As explained in Section 4.2, the most accurate model avoids classifying a low quality subject as an excellent subject and hence, seeks to **maximize the F1 Score** metric.

Therefore, the results bring out Experiment 2.2 as being the one yielding the best results. To classify a subject quality it uses the country, sector, structural and content features, employing the average to calculate the *lemmas_past_performance*, producing a F1 score of 62.2% and an Accuracy of 62.4%. It is important to highlight the great achievements, especially when taking into account the multiclass problem at hand and that, a random prediction would produce a $\frac{1}{5}$% of Accuracy, only 20% compared to the notable achieved 62%.

## 6. CONCLUSIONS

This study was set out to overcome and improve the notably low open rates that are hindering companies of engaging efficiently their audience.

Thus, this project sought to help editors on creating relevant subject-lines which capture recipients' attention, creating sufficient curiosity on them to open and read the content of the message.

This thesis proposes a model, which, employing data mining classification techniques, is capable of predicting a subject-line quality, considering the country from where is sent, the business sector, the structural and the content features, i.e. the set of words chosen.

In this particular case study and, regardless of the specificities of the different model's tests, Random Forest Classification algorithm is the one providing the best results. The experiment which calculates the *lemmas_past_performance* feature using the naive average of the quality of the lemmas contained in the subject achieved extremely favorable performance results when employed with Random Forest Classification technique.

Therefore, we conclude that the data available for this study was sufficient for an effective prediction of the subject quality, with the model reaching performance levels of about 62.4%, in terms of Accuracy and 62.2%, in terms of F1 Score, although believing that some external and hidden characteristics of the data may have contributed to lower outcomes as, for example, information regarding the quality of the list of subscribers or the day of the week the campaign was delivered.

From the case study company standpoint, this promising tool will support their customers on creating engaging and relevant subject-lines, contributing to better performance results, i.e. more emails opened and, thus, more successful marketing campaigns. Indeed, this thesis contributed to the development of the first tool embedded into a marketing automation platform, able to predict the quality of a subject through Data Mining techniques. Moreover, despite all the research carried out in the area, this project is the first one offering a model prepared for adjusting itself to the natural evolution of trends and patterns over time.

Conversely, predicting the probability of an email to be open through the subject-line quality remains a complex task due to diverse factors. Firstly, personal interests and curiosities can strongly influence an individual to open or to neglect a specific email. Secondly, the sender recognition and reputation can influence the decision to open or overlook an email and finally, owning an outdated customer database, with low-quality user data can completely skew email campaign performance results.

In conclusion, this thesis provides a prosperous and extremely valuable tool for any marketing automation platform seeking to offer customers solutions to overcome their daily obstacles during the creation of email marketing campaigns.

## 7. FUTURE WORK

These days, virtually every company in the world uses email marketing to operationalize their customer relationship management strategy and, therefore, this field has a tendency to continue lacking for advances and improvements for better reaching their customer's needs.

Some promising future works could be developed, such as build a different model for each distinct language since having multiple languages in the same model increases complexity, or implement n-grams [23] approach in the content analysis, so the model could interpret a set of $n$ words as a whole instead of just interpreting word by word.

Also, the authors believe that the results can be improved if the model considers both the subject and the sender reputation as the OR drivers, recognizing that notorious companies such as Zara, Apple or Ryanair can influence the willingness and curiosity to open an email. Additionally, sentiment analysis should be explored, trying to quantify the psychological effects induced when reading an email subject-line, for example, happiness, curiosity or sadness.

Instead of analyzing just the set of words, each one treated independently and not considering any order, the subject semantic, i.e the general meaning of the subject could be examined. Moreover, it could be valuable to get to know the type of subscriber and what he/she enjoys most to read. This may involve trying to define a customer profile, for example using age as a proxy for his/her behavior. Finally, a more sophisticated model for feature selection should be tested, such as forward and backward selection procedure, and it could be explored the potential of techniques dedicated to ordinal dependent variables, namely neural network and support vector approaches to ordinal regression.

## 8. REFERENCES

[1] I. Abu El-Khair. TF*IDF. In *Encyclopedia of Database Systems*. 2017.

[2] D. Apostolopoulos. Improving Subject Lines: An endless quest using Machine Learning (pt. 1/2), 2017.

[3] D. Apostolopoulos. Improving Subject Lines: An endless quest using Machine Learning (pt. 2/2), 2017.

[4] R. Balakrishnan and R. Parekh. Learning to predict subject-line opens for large-scale email marketing. In *Proceedings - 2014 IEEE International Conference on Big Data, IEEE Big Data 2014*, 2015.

[5] L. Breiman. *Random Forests*. 2001.

[6] R. Dania TODOR. Promotion and communication through e-mail marketing campaigns. *Bulletin of the Transilvania University of Braşov Series V: Economic Sciences*, 10(59), 2017.

[7] G. B. Duggan and S. J. Payne. How much do we understand when skim reading? 2006.

[8] S. Feld, H. Frenzen, M. Krafft, K. Peters, and P. C. Verhoef. The effects of mailing design characteristics on direct mail campaign performance. *International Journal of Research in Marketing*, 30(2):143–159, 2013.

[9] C. R. Ferreira, S. A. Saraiva, J. S. Garcia, and G. B. Sanvido. Feature Engineering for Text Classification. *Bioinformatics*, 2014.

[10] J. H. Friedman. Stochastic gradient boosting. *Computational Statistics and Data Analysis*, 2002.

[11] B. T. John D. Kelleher. Data Science. The MIT Press (April 13, 2018).

[12] D. D. Lewis. Naive (Bayes) at forty: The independence assumption in information retrieval, 1998.

[13] R. Miller and E. Y. Charles. A psychological based analysis of marketing email subject lines. In *16th International Conference on Advances in ICT for Emerging Regions, ICTer 2016 - Conference Proceedings*, 2017.

[14] O. Okun and Giorgio Valentini. Supervised and Unsupervised Ensemble Methods and their Applications. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2010.

[15] J. R. Quinlan. Induction of Decision Trees. *Machine Learning*, 1986.

[16] N. S. Sahni, S. C. Wheeler, and P. K. Chintagunta. Personalization in Email Marketing: The Role of Non-Informative Advertising Content. *Ssrn*, pages 1–41, 2016.

[17] C. Saranya and G. Manikandan. A study on normalization techniques for privacy preserving data mining. *International Journal of Engineering and Technology*, 2013.

[18] A. C. Steinwart. Support Vector Machines, 2018.

[19] P.-N. Tan, M. Steinbach, V. Kumar, T. Pang-Ning, M. Steinbach, and V. Kumar. Introduction to data mining: Instructur's. *Library of Congress*, page 769, 2006.

[20] K. M. Ting. *Confusion Matrix*, page 209. Springer US, Boston, MA, 2010.

[21] A. L. C. T. van Rijswijk. *Cross-Validation*, page 306. Springer US, Boston, MA, 2017.

[22] J. Wainer, L. A. Dabbish, and R. Kraut. Should I open this email?: inbox-level cues, curiosity and attention to email. 2011.

[23] X. Wang, A. McCallum, and X. Wei. Topical N-grams: Phrase and topic discovery, with an application to information retrieval. In *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2007.

[24] Watson Marketing. Watson Marketing Email and Mobile Metrics for Smarter Marketing 2018 Marketing Benchmark Report. page 48, 2018.

[25] A. Zerium. Artificial Neural Networks Explained, 2018.

[26] R. Zhao and K. Mao. Fuzzy Bag-of-Words Model for Document Representation. *IEEE Transactions on Fuzzy Systems*, 26(2):794–804, 2018.

## APPENDIX

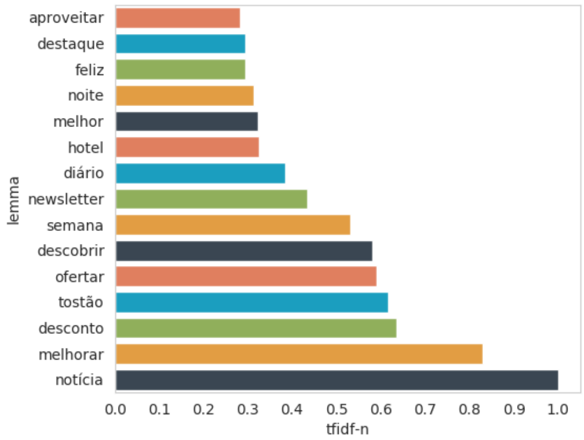


**Figure 8: Top 15 most relevant lemmas**

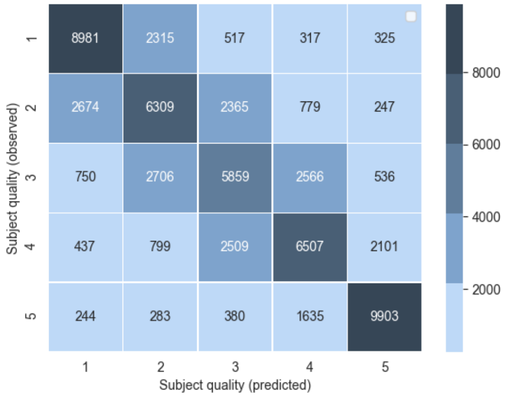


**Figure 9: Confusion Matrix on Random Forest : Experiment 1**

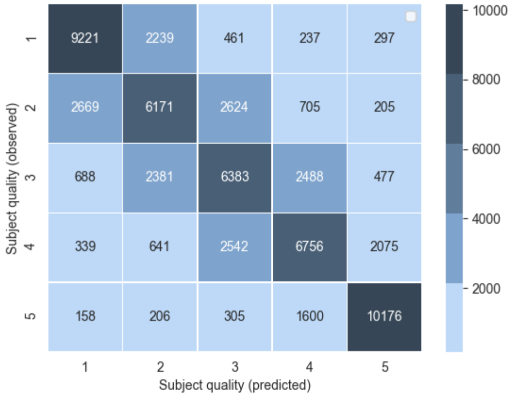


**Figure 10: Confusion Matrix on Random Forest : Experiment 2.2**

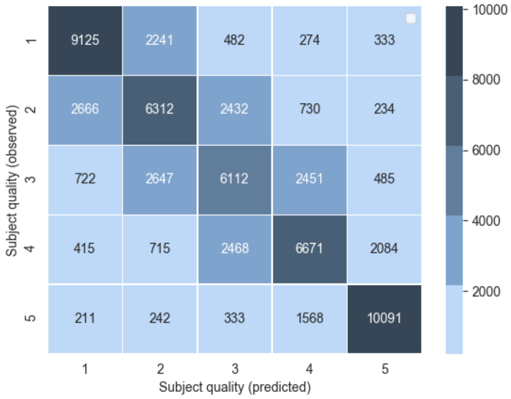


**Figure 11: Confusion Matrix on Random Forest : Experiment 2.3**

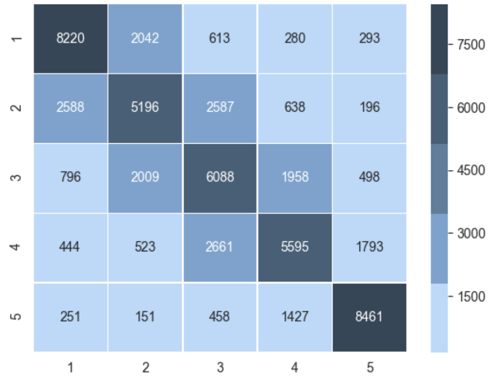


**Figure 12: Confusion Matrix on Random Forest : Experiment 3**

**Table 10: Normalized features importance: Experiment 2.2**

| Feature | Importance |
|---|---|
| lemmas_past_performance | 1,0000 |
| nr_chars | 0,8143 |
| case_percentage | 0,5891 |
| nr_ words | 0,5596 |
| nr_lemmas | 0,3768 |
| Marketing | 0,1816 |
| Publishing/Media | 0,1476 |
| Hotel-Restaurant and Travel | 0,1143 |
| brazil | 0,1142 |
| portugal | 0,1070 |
| punctuation | 0,0969 |
| numbers | 0,0929 |
| special_chars | 0,0871 |
| currency | 0,0849 |
| Services | 0,0577 |
| Undefined | 0,0572 |
| E-commerce | 0,0437 |
| Information Technologies | 0,0361 |
| Arts and Entertainment | 0,0309 |
| personalization | 0,0295 |
| Retail | 0,0294 |
| Other | 0,0276 |
| Education | 0,0247 |
| emojis | 0,0214 |
| Beauty/Health | 0,0206 |
| Non-profit | 0,0171 |
| Manufacturing | 0,0164 |
| spain | 0,0123 |
| Government | 0,0078 |
| Food | 0,0062 |
| Accounting/Financial Services | 0,0054 |
| Real Estate | 0,0054 |
| united_states_of_america | 0,0053 |
| Architecture/Construction | 0,0043 |
| colombia | 0,0041 |
| Church | 0,0031 |
| peru | 0,0025 |
| angola | 0,0011 |
| australia | 0,0008 |
| Insurance | 0,0005 |
| united_kingdom | 0,0004 |
| mexico | 0,0004 |
| argentina | 0,0003 |
| prefix | 0,0003 |
| mozambique | 0,0001 |
| honduras | 0,0001 |
| chile | 0,0000 |
| malta | 0,0000 |
| curacao | 0,0000 |

**Table 11: Normalized features importance: Experiment 2.3**

| Feature | Importance |
|---|---|
| nr_chars | 1,0000 |
| case_percentage | 0,7120 |
| nr_ words | 0,6726 |
| nr_lemmas | 0,4690 |
| lemmas_past_performance | 0,4689 |
| Marketing | 0,2092 |
| Publishing/Media | 0,1800 |
| brazil | 0,1296 |
| Hotel-Restaurant and Travel | 0,1278 |
| portugal | 0,1232 |
| punctuation | 0,1118 |
| numbers | 0,1067 |
| special_chars | 0,1034 |
| currency | 0,0954 |
| Undefined | 0,0672 |
| Services | 0,0649 |
| E-commerce | 0,0511 |
| Information Technologies | 0,0430 |
| Arts and Entertainment | 0,0377 |
| Retail | 0,0359 |
| personalization | 0,0325 |
| Other | 0,0325 |
| Education | 0,0284 |
| emojis | 0,0257 |
| Beauty/Health | 0,0218 |
| Non-profit | 0,0202 |
| Manufacturing | 0,0179 |
| spain | 0,0148 |
| Government | 0,0095 |
| Food | 0,0071 |
| Accounting/Financial Services | 0,0065 |
| Real Estate | 0,0060 |
| united_states_of_america | 0,0059 |
| Architecture/Construction | 0,0049 |
| colombia | 0,0047 |
| Church | 0,0035 |
| peru | 0,0025 |
| angola | 0,0014 |
| australia | 0,0008 |
| Insurance | 0,0005 |
| united_kingdom | 0,0005 |
| mexico | 0,0004 |
| argentina | 0,0003 |
| prefix | 0,0003 |
| mozambique | 0,0002 |
| honduras | 0,0001 |
| malta | 0,0001 |
| chile | 0,0001 |
| curacao | 0,0000 |