

Universidad ORT Uruguay  
Facultad de Ingeniería

# Age Prediction of Spanish-speaking Twitter Users

Entregado como requisito para la obtención  
del título de Master en Ingeniería

Verónica Tortorella - 153303

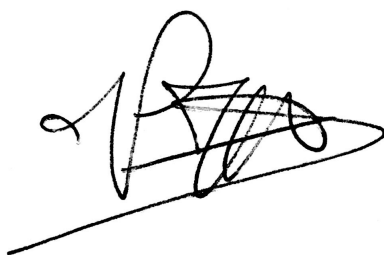
Tutor: Sergio Yovine

2018

# Declaración de Autoría

Yo, Verónica Tortorella, declaro que el trabajo que se presenta en esta obra es de mi propia mano. Puedo asegurar que:

- La obra fue producida en su totalidad mientras realizaba el Proyecto;
- Cuando he consultado el trabajo publicado por otros, lo he atribuido con claridad;
- Cuando he citado obras de otros, he indicado las fuentes. Con excepción de estas citas, la obra es enteramente nuestra;
- En la obra, he acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, he explicado claramente qué fue contribuido por otros, y qué fue contribuido por mí;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.

A handwritten signature in black ink, consisting of a series of loops and a long horizontal stroke at the bottom.

Verónica Tortorella  
14 de febrero de 2018

# Agradecimientos

Me parece importante agradecer a todas aquellas personas que me brindaron su apoyo y colaboración durante el desarrollo de la Maestría.

En primer lugar agradezco a Benjamin Machin, quien ofició de sponsor y referente técnico. Su experiencia fue vital para guiarme a lo largo de todo el proceso de desarrollo.

Asimismo, quiero agradecer a Pyxis, quien bajo el programa Pyxis Research me permitió dedicar horas de mi jornada de trabajo a la realización de esta tesis.

Agradezco también a Sergio Yovine, mi tutor, quien compartió sus ideas, su tiempo y me guió durante el camino.

Finalmente el agradecimiento más importante va dedicado a mi familia, mi novio y amigos, quienes me dieron su apoyo incondicional durante la tesis, así como durante toda la maestría.

# Resumen

La predicción de la edad en Twitter es un tema muy interesante pero a la vez constituye un gran desafío, que surge como necesidad para mejorar el marketing online así como para colaborar en la detección de la ciber pedofilia, identificando a los usuarios que fingen ser menores mediante el uso de perfiles falsos.

En el presente trabajo nos enfocamos en el análisis de los usuarios de Twitter cuyo lenguaje es el Español. Así como toda tarea de caracterización de autores, la predicción de edad depende en gran medida del lenguaje empleado por el grupo objetivo. En el caso particular del Español, una de las mayores complejidades radica en la falta de un corpus etiquetado. En consecuencia, en este trabajo se exploran estrategias de generación, y como resultado surge TweetLab, un software compuesto por un streamer encargado de la extracción y etiquetado automático de usuarios, así como de su customización para los usuarios de lengua Española ubicados en Uruguay y parte de Argentina.

Otra complejidad significativa es la limitante de largo de los tweets (280 caracteres). Para mitigar esta dificultad, resulta necesario recolectar la mayor cantidad de información posible a partir de los mismos, así sea mediante la inferencia de relaciones no explícitas o a través del cálculo de métricas lexicográficas.

En consecuencia, analizamos tres tipos de atributos: metadatos del usuario, atributos de estilometría sobre el texto de los tweets, y atributos resultantes de la aplicación de técnicas de Procesamiento de Lenguaje natural sobre tweets así como listas de suscripción, las cuales contienen información acerca de los intereses del usuario. Asimismo, incluimos en el conjunto una serie de atributos novedosos e innovadores que modelan la vinculación del perfil de Twitter con otras redes sociales.

Dichos atributos recolectados son posteriormente utilizados para entrenar los modelos de Aprendizaje Automático, con el fin de predecir la edad de los usuarios y así proceder a clasificarlos en los rangos etéreos definidos.

Finalmente realizamos una serie de experimentos con distintos set de datos y algoritmos. Los resultados experimentales muestran que los atributos extraídos constituyen un elemento muy útil a la hora de detectar la edad de los usuarios.

**Palabras clave:** Predicción de Edad; Redes Sociales; Clasificación multi-clase; Representaciones Latentes; Caracterización de Autor; Categorización de Texto; Estilometría; Detección de Ciberpedofilia; Procesamiento de Lenguaje Natural; Aprendizaje Automático.

# Abstract

Age prediction in Twitter is an interesting but challenging task, that arises as a way to improving online marketing and potentially helping with the detection of cyber-pedophiles who pretend to be younger users by using fake profiles.

In this work, we focus the analysis on Twitter users writing in Spanish. As any author profiling task, age prediction greatly depends on the language used by the target group. In the case of Spanish, one of the biggest difficulties is the lack of a labeled corpus. Hence, we explore strategies to generate it and, as a result, we develop TweetLab, a software pipeline to extract and label Twitter and customize it for users in Spanish from Uruguay and part of Argentina.

Another identified problem is the short nature of the tweets. Therefore, it is necessary to gather as many information as possible from them, even by inferring hidden relations or calculating lexical metrics.

In order to do that, we study three types of features: user metadata, stylistometric features from tweets text and Natural Language Processing features extracted from tweets as well as subscription lists, which contain information about the user's interests. We also present a novel set of features that model the presence of other social networks profiles linked to the Twitter account.

Those extracted features are used to build models which are used as input of Machine Learning algorithms, in order to predict the age of the users and classify them into the age groups defined. We run several experiments with different datasets and algorithms. The experimental results show that these features work well in detection of users age.

**Keywords:** Age prediction; Social Networks; Multi-class classification; Latent Representations; Author Profiling; Text categorization; stylometry; cyberpedophilia detection; Natural Language Processing; Machine Learning.

# Content

<b>1</b>	<b>Introduction</b>	<b>11</b>
1.1	Introduction . . . . .	11
1.2	Motivation . . . . .	13
1.3	Related Work . . . . .	15
1.3.1	Study of short texts . . . . .	16
1.3.2	The problem of extracting meaningful features . . . . .	16
1.3.3	The problem of Age Prediction and useful Machine Learning Techniques . . . . .	18
1.3.4	The labelling problem . . . . .	20
1.4	Objectives . . . . .	22
1.5	Contributions . . . . .	23
1.6	Outline of the thesis . . . . .	24
<b>2</b>	<b>Corpus Generation</b>	<b>25</b>
2.1	First Approach . . . . .	25
2.2	Second approach using web scraping . . . . .	27
2.3	Final approach: TweetLab . . . . .	28



2.3.1	Overview . . . . .	28
2.3.2	Design of the solution . . . . .	29
2.4	Underlying Software Tools . . . . .	33
2.4.1	Document Database . . . . .	33
2.4.2	Twitter API . . . . .	34
<b>3</b>	<b>Feature Extraction</b>	<b>36</b>
3.1	User metadata features . . . . .	37
3.2	Stylometric features in tweets . . . . .	39
3.3	Natural Language Processing features . . . . .	41
3.3.1	Tweets . . . . .	41
3.3.2	Subscription Lists . . . . .	46
<b>4</b>	<b>Experimental evaluation</b>	<b>50</b>
4.1	Classifiers . . . . .	50
4.2	Experiments . . . . .	52
4.2.1	Experiment 1: Prediction between the five Age Groups	53
4.2.2	Experiment 2: Prediction for Cyber Pedophilia Detection	61
4.2.3	Experiment 3: Prediction with a Balanced Dataset . .	71
4.2.4	Experiment 4: Our Predictions vs. Microsoft Face API	73
<b>5</b>	<b>Conclusions</b>	<b>75</b>
<b>6</b>	<b>Bibliographical References</b>	<b>78</b>



# 1 Introduction

## 1.1 Introduction

According to Cambridge Dictionary, a social network “is a website or computer program that allows people to communicate and share information on the internet using a computer or mobile phone”.

On Twitter and on most online social networks, the most basic element for information sharing is the user’s profile. A profile is a user-controlled page that includes descriptive information about the person it represents. Also, it can be connected with other profiles through explicitly declared friend relationships and numerous messaging mechanisms [1].

Twitter allows users to choose between making their profiles public (default option) or private. If a user’s profile is designated as private, only the user’s friends are allowed to view the profile’s detailed personal information (tweets, friends, subscription lists). However, a private profile still reveals the user’s name, picture, biography, and location.

After the creation of this social network more than ten years ago, Twitter still remains one of the most popular sites of social-media networking in use. Since its creation, it has become more and more viral. In fact, the first hashtags appeared in Twitter in 2007, when they were proposed as a way to keep together related tweets.

Today, it has about 330 million monthly active users and around 500 million tweets are posted into the platform every 24 hours from all over the globe, yielding an impressive rate of 6,000 tweets per second<sup>1</sup>.

---

<sup>1</sup><https://www.brandwatch.com/blog/44-twitter-stats>

Since the beginning, the platform had a strict tweet size limit, which allowed a maximum of 140 characters per tweet. This constraint makes really hard to profile the author. On November 17th 2017, after a big controversy, Twitter decided to expand its character count to 280 to all users in supported languages<sup>2</sup>. However, it is still considered a short text when it comes to trying to predict the age of the user.

In fact, communication in social networks happens via short messages, often using non-standard language variations [2]. It is unstructured and noisy, and people do not always spell words correctly, sometimes even on purpose, to show excitement (i.e., Happyyyy!) or to maximize their typing speed by omitting letters or using acronyms (i.e., TGIF, brb, idk). Some people even have their own set of made-up words. In addition, punctuation marks are rarely used and uppercase works as a way to emphasize the content. It is also very frequent to find emoticons and smileys in tweets. These characteristics make it really hard to apply Natural Language Processing (NLP) techniques on this type of texts.

Many studies have been conducted regarding this matter. They reached to the conclusion that younger people use more alphabetical lengthening, more capitalization of words, shorter words and sentences, more self-references, more slang words, and more internet acronyms [3],[4],[5],[6],[7].

Another difficulty lies in the fact that Twitter has limited metadata available about its users. Important attributes of the user such as age and gender that are fundamental to provide personalized services are not available in profiles or metadata [5].

Even though Twitter requires the birth date when trying to access to restricted content, it checks that the date entered is at or above the applicable legal age limit for the country. To remember this information, Twitter may associate with the account an acknowledgement that the user met or did not meet the age requirement, but it does not keep the birth date entered<sup>3</sup>. As a consequence, there is no way to extract this information by calling the Twitter API or doing scraping on the profile.

In fact, little information about the user is publicly shown in the profile, and many social networks do not even provide open access to the user's data. So, it is very difficult to come up with a labeled training set to build

---

<sup>2</sup><https://techcrunch.com/2017/11/07/twitter-officially-expands>

<sup>3</sup><https://help.twitter.com/en/safety-and-security/age-verification>

machine-learning models. In the case of Twitter, it provides an API to request information, but there is a limit of requests in a period of time, and some information is not returned such as all the tweets of a user, or the gender and age.

Furthermore, if we take a look at the studies about this particular topic, the large majority analyze age prediction in social networks for English users only. However, almost 470 million people speak Spanish and other 21 million study it as a foreign language. Spanish is the third language more used in internet, and the second one in massive social networks such as Facebook or Twitter. In fact, 7.9% of the users of the network express themselves in Spanish, and it has expanded 1,100% between 2,000 and 2013<sup>4</sup>.

As we can see, user's age is a difficult attribute to learn, since it changes constantly, its perception varies due to a series of socioeconomic variables and there is no explicit indicator in Twitter.

## 1.2 Motivation

Indeed, age prediction is a special case of author profiling. There are many reasons why author profiling is important. Two of the most important ones are online marketing and the detection of pedophiles who pretend to be younger users by using fake profiles.

Firstly, 65.8% companies with 100+ employees use Twitter for marketing, while the average Twitter user follows five businesses.

The 92% of companies tweet more than once a day, 42% tweet 1-5 times a day, and 19% tweet 6-10 times a day. Moreover, 54% of users surveyed by Twitter reported that they had taken action after seeing a brand mentioned in tweets (including visiting their website, searching for the brand, or re-tweeting content)<sup>5</sup>.

In conclusion, not only Twitter is a widespread social network, but also has become a powerful platform for businesses, with important applications in advertising, personalization and recommendation (i.e., to viralize market-

---

<sup>4</sup>[https://elpais.com/cultura/2016/01/19/actualidad/1453209550\\_723124.html](https://elpais.com/cultura/2016/01/19/actualidad/1453209550_723124.html)

<sup>5</sup><https://www.brandwatch.com/blog/44-twitter-stats>

ing campaigns and to get in touch with potential customers).

From a marketing viewpoint, companies may be interested in knowing, on the basis of the analysis of blogs and online product reviews, the demographics of people that like or dislike their products. The focus is on author profiling in social media since we are mainly interested in everyday language and how it reflects basic social and personality processes [8].

Likewise, the brands need to determine whether a follower meets a minimum age requirement. On one hand, they must verify whether a person's age is relevant to the industry target, and on the other, they must be sure it meets legal guidelines. This is fundamental for advertisers with content not suitable for minors (e.g., alcohol advertisers)<sup>6</sup>.

Due to this, Twitter provides a mechanism for age screening, a solution for brands that requires new Twitter followers to enter their birth date before being permitted to follow their account. The user will only be required to enter the age information once, and this information will be accessible to all brands who want to participate in age screening. This way, Twitter advertisers who use the solution will not have access to the birthdate or age, but will know when you've entered an age that's above their indicated threshold. Moreover, in some networks the age is a required field. This is the case of Facebook, where birthdate is mandatory, and the minimum age requirement is 13.

Back in 2016, the privacy policy of Twitter contained a section "Our Policy Towards Children" which stated the following: "Our Services are not directed to persons under 13. If we become aware that a child under 13 has provided us with personal information, we take steps to remove such information and terminate the child's account"<sup>7</sup>. However, on the latest version this section was removed, and the birthdate is not required to create a new twitter account.

It is very usual for young audiences to provide a fake age in social networks, in order to access unrestricted content, and sometimes even to be able to create an account. So even if this attribute was public on the user profile, it is not trustworthy.

The other motivation for this study is related to sexual predators. As we

---

<sup>6</sup><https://help.twitter.com/en/safety-and-security/age-verification>

<sup>7</sup>[https://twitter.com/en/privacy/previous/version\\_11](https://twitter.com/en/privacy/previous/version_11)

mentioned before, social networks sometimes work as hunting grounds for pedophiles by creating fake profiles with a false name, profile picture, age, gender and location, posing as adolescents while hiding their true identity. Similarly, the massive amount of profiles and interactions between them make manual analysis impossible, not only for social network moderators, but also for law enforcement teams.

In fact, to catch online predators, law enforcement officers or volunteers pose as youths in social networks. However, the number of law enforcement officers and volunteers will never be enough to detect and deter people with criminal intent [9].

As a consequence, there is the need to have automated methods to identify this type of behavior, or at least to narrow the results to a list that can be manually verified.

On the last decade, a lot of studies have been driven in order to decipher information about the author from his texts. In fact, there has been significant progress in natural language processing to perform analysis of syntactic and semantic properties of texts.

From a forensic linguistics perspective one would like being able to know the linguistic profile of the author of a harassing text message (language used by a certain type of people) and identify certain characteristics (language as evidence) [8].

Most text analysis have focused on the topic of a text, focusing on what it is about, instead of considering how it was written, which provides much useful information in its style. Also, the majority focused on the study of lengthy texts or several short texts per author (at least 1,000 words) [10].

## 1.3 Related Work

In recent years, social networks have become a massive phenomenon, used by millions of people all around the world. They contain a lot of information that can be helpful to discover interesting facts about the users, and even help fight crime. Hence, a lot of work has been made in the field of “Author Profiling and Identification”.

### 1.3.1 Study of short texts

As we previously mentioned, most of the traditional analysis has been made over long texts. Hirst and Feiguina [11] explain “The smaller the text in the corpus, the less certain the results are”. Thus, methods that could predict user characteristics with reliability on smaller texts would be welcome both in literary studies and in forensic analysis.

In 1996, Glover and Hirst [12] already started working with short texts. Their aim was to discriminate the authorship of collaborative documents, by working with short texts as several paragraphs or just a single paragraph. They used conventional authorship discrimination methods to see how well they could work on text of just a few paragraphs, but got poor results.

Later, Burrows [13] performed authorship profiling on poems of less than 500 words, obtaining an accuracy of 27%, and concluded that his procedure works only on texts greater than 1,500 words. Graham et al [14] also tried simple letter bigrams, but these failed for texts of less than about 500 words.

Ultimately, as Hirst [11] states, the problem with small texts is that they are small. They contain less information, and hence fewer clues to author profiling. It therefore becomes more important to use as much information as possible from what is given. An interesting strategy to try is to make better use of the stylistic properties of the text, as well as the user metadata.

### 1.3.2 The problem of extracting meaningful features

After traversing across the bibliography on this topic, most of the authors convey on some features that are worth considering while predicting age in texts, while others only increment dimensions degrading performance.

Word n-grams (unigrams, bigrams and trigrams) are one of the featured attributes when predicting age [10]. Some authors like Pendar [15] suggests an enhancement by removing the stopwords. In his case, he removed the 79 most frequent word types in his corpus.

Later Tam et al [9] followed this line of work, and included as well character trigrams and word meta-data features, such as average number of capital letters per post, average number of tokens per post, average number of emoticons per post, average post length, and average word types per document.



They concluded that the feature that did best was trigrams, and realized that character n-grams were not important, so the context appears to be a necessary feature.

Peersman et al [2] arrived to the same conclusion: token features outperformed character features. In fact, word unigram features (i.e., words, punctuation marks and emoticons) were the most robust feature type for age prediction in his study.

Also, Tam [9] proposed to use entropy as a measure of information gain, to learn how much a given feature contributes to identify an age class. Thus, they came up with a new list of stopwords composed by the 75 n-grams with the highest entropy. The accuracy obtained was the same, so this works as a tool to reduce dimensionality without degrading performance. Nevertheless, the entropy outperformed the mutual high-frequency n-grams.

Goswami et al [6] studies age prediction in English blogs, for age groups of 10s, 20s, 30s and higher. It proposes a new type of stylometric analysis by considering the use of slang words (non-dictionary words with frequency greater than 50) combined with content words, as well as the average length of sentences across various age groups and gender. They observed interesting facts, for instance, that teenagers generally use more slang words than adults. Also, they observed that the results are not sufficient to conclude that the average sentence length increases with age. In fact, when adding average sentence length as a feature, the accuracy went from 80.32% to 80.38% which is a minimal enhancement.

Marquadt et al [16] also considers stylistic features such as readability, HTML tags, grammatical errors, and emoticons, but also extracts content-based ones, using the Linguistic Inquiry and Word Count (LIWC) dictionary<sup>8</sup>, the Medical Research Council (MRC) psycholinguistic database [17], which capture information about frequency of words that connote psycholinguistic concepts such as familiarity, concreteness, and imagery.

Rosenthal et al [3] studies the performance of three types of features: 1) “Online Behavior” which includes quantity of friends, quantity of posts, average number of comments, 2) “Lexical-Stylistic” including emoticons, acronyms, slang, punctuation, capitalization, sentence length and quantity of links and images, and 3) “Lexical-content” with the following features: top collocations in the age group, top syntax collocations in the age group,

---

<sup>8</sup><http://liwc.wpengine.com/>

top part-of-speech collocations in the age group, and top words in the age group. Like many other authors, they concluded that the stylistic features outperform content based ones. As a novel discovery, they identified an increase of the accuracy of 10% when adding online behavior features as well. The best accuracy was achieved by combining bag of words (BOW), online behavior and lexical stylistic features.

Likewise, Rao and Yarowsky [5] compare the efficacy of Sociolinguistic features against n-gram features. On one hand, the lexical choices can distinguish from young and old users: empiric results show that terms like “dude” or “bro” indicate a younger user as well as the use of ellipses (...), while older users are more articulate. On the other hand, unigrams and bigrams were obtained from the tweet text. In the end he arrived to the same conclusion as many other authors: n-gram model performs better. However, they found an interesting fact: alphabetic character repetition was 30% more likely in younger users.

### **1.3.3 The problem of Age Prediction and useful Machine Learning Techniques**

Some authors have treated age prediction as a classification problem, while others considered it a regression. Basically, classification is about predicting a label and regression is about predicting a quantity. In our case, classifying consists of assigning an age range, while regression predicts the exact age of a user.

PAN Clef [8] is an organization that conducts a series of scientific events and shared tasks on digital text forensics. One of those tasks consists of doing Author Profiling over Twitter users and their tweets. As a result, many authors explored this problem and obtained interesting conclusions.

This is the case of the work of Arroju et al [10], in which the authors predict age in a multilingual setting considering tweets in 2 different languages: English and Spanish. In fact, it is one of the few studies that focus on Spanish tweets, but it included a very small Spanish dataset (61 users). In order to do that, they perform multi-class classification with class labels for every age range. They applied a linear model with Stochastic Gradient Descent (SGD) learning. As a result, after using 10-fold cross validation they obtained an accuracy of 0.69 for English tweets, and 0.48 for Spanish tweets

(only 0.2 better than the majority baseline). As we can see, the results for English tweets outperformed the outcome for Spanish.

Marquadt et al [16] also analyzes texts in Spanish, but in this case it includes the following online media genres: 88 blogs, 178 Twitter feeds and 1,272 unspecified social media posts. In order to predict age and gender, two models are trained. The first model is based on label powerset transformation (which turns a multi-label classification problem into a single label one by unifying labels for age and gender, i.e., female-18-24). This model is later trained with Support Vector Machines (SVM). The second model is based on classifier chains. It consists in two single label classifiers, in which the prediction made by the first is used as a feature in the second. In this case the gender inferred with the first classifier is used as a feature for age prediction. With both models they achieved the same accuracy of 48.31 for Twitter users, very close to the baseline. In this case, the accuracy for Twitter users in Spanish overcame the accuracy of English ones, where the models scored an average of 47.31. This lies on the fact that features that work well across many genres may not necessarily perform well on others. This is why we will focus our study in Twitter corpora only.

Prior work by Peersman et al [2] has examined the corpus of more than 1.5 million Flemish dutch posts from the Belgian social network Netlog, and predicted the age of the users. This one of the few papers that link the age prediction problem with fake profiles and pedophile identification. For that reason, they considered the relationship between two age ranges in particular: min16 (from 11 to 15 years old) and plus25 (25 and older), this way there is no doubt about the illegal character of a sexual interactions between both classes. For the main dataset of 10,000 posts per class, the SVM classifier yielded an accuracy of 71.3% for min16 vs. plus16, which increased when the distance between the age groups became larger. Also, they run experiments to test the minimum dataset size with 10,000, 5,000 and 1,000 instances per class, arriving to the conclusion that with only 10% of the dataset the classifier was still able to improve considerably upon baseline performance for binary age classification.

Likewise, Tam and Martell [9] also worked on an automatic recognition system of adults conversing with teens, leading towards building a system to detect online predators. They focused on online chat rooms conversations to identify adults soliciting youths. They experimented with Naive Bayesian (NB) and SVM classifiers. Tam et al [9] and Pendar [15] also experimented with these two models, and [9] observed that SVM performed slightly better

while classifying in the different age ranges, and did significantly better when classifying teens from adults. However, they mentioned the possibility that NBC did not do as well because the lack of class balance in the dataset.

Some authors like Schler et al [18] considers this aspect, and in order to prevent bias, they create a subcorpus consisting of an equal number of male and female in each age group. Most of the bibliography tried to balance by gender, but not by the size of each age class. Having said that, to avoid a bias problem Rao [5] constrained the number of users in each class to be similar.

In the context of this thesis, we will face the problem of working with unbalanced datasets as well. This is due to the fact that some age classes use social networks more than others. For instance, older users tend to be a small group.

On the other hand, age prediction can be seen as a regression problem, in order to predict the exact age of the users. Perozzi and Skiena [19] propose an approach, called DeepWalk, to learn social representations which capture community information as covariates. According to their preliminary experiments on Pokec, the most popular social network in Slovakia, the accurate prediction is possible, even when as little as 5% of users have shared their age. After conducting tests with four different methods for network regression, ordinary linear regression (OLS) on DeepWalk features provided the best mean absolute error (MAE) until 95% of training data is used, obtaining a predicted age within 4.15 years on average.

This is also the case of the work of Nguyen et al [20], where a linear regression model is used to predict the age in years of a person. They perform L2 regularization to prevent over-fitting and cross-validation on the training set to tune the parameters. For the feature extraction task, they only use unigram features. The model achieved a MAE of 6.15 years.

### **1.3.4 The labelling problem**

Caverlee et al [1] presents an extensive analysis of over 1.9 million profiles in MySpace, a social network considered the most visited one worldwide between 2005 and 2008. MySpace profiles, similar to Netlog, contain the user's age, which makes it simpler to gather a dataset of labeled users.

The same happens with Tam et al [9], where data is formed by 160,740 posts belonging to 2,161 different authors, all of them labeled by the self-reported age in their profile.

This makes a big difference compared to Twitter, where the age does not appear in the profile. So, more complex methods must be used to label users to gather a training dataset. In fact, MySpace would still reveal the user’s age even if it had a private profile.

Rao et al [5] already mentioned the difficulty of age prediction in Twitter, and the necessity to build an annotated dataset from scratch. In order to do that, they used crawlers and manual annotation. The crawls looked at Twitter lists for topics that should identify an age class (i.e., “baby boomers”, “young moms”), and also looked for specific terms in description (i.e., “junior”, “freshman”). They also crawled the user profile in other social networks like Facebook and LinkedIn that were linked with the Twitter account. By doing so they obtained 1,000 users in each class (2,000 in total). Even though they mentioned that the manually annotated Twitter dataset will be available as a shared resource to the research community, it was not possible to find it on the web.

In the works of Arroju et al [10] and Marquardt et al [16], the labeled dataset was provided by PAN Clef. This dataset is very small and there is no documentation on the labeling process they used.

Goswami et al [6] performs a stylometric analysis of bloggers’ age and gender over a corpus of more than 70 thousand blogs from almost 20 thousand different bloggers, but again, each one had author-provided indication of both gender and age.

Nguyen et al [20] proposes an alternate method to label Twitter users through an online game, in which thousand of players guessed the age and gender of Twitter users based on only the tweets, and compared the human guess against an automatic prediction. To do this, they annotated biological sex and chronological age using information from tweets, the profile, and external social media, such as Facebook and LinkedIn. After comparing users guesses against actual age, they concluded to be 5.7 years off on average.

In particular, the players had difficulty predicting the ages of older Twitter users. This can be explained by sociolinguistic studies that have found that people between 30 and 55 years use standards forms, because of the pressure in the workplace. It concludes that creating datasets to predict age and

gender is a difficult task, in fact, when using a dataset biased towards people who show a strong gender identity (i.e., sororities), the results obtained may not be representative of a more random set.

Nguyen et al [4] and Al Zamal et al [21] propose a complementary approach to generating age-labeled data when labeled data are unavailable in Twitter by identifying Twitter accounts that had tweets about birthdays that also mentioned the age of the person: either individuals who tweeted about their own birthdays (e.g. “Happy XX birthday to me!”) or individuals who sent birthday wishes to others (e.g. “Wishing @xxxxxxx a happy XX birthday”).

Nguyen et al also used age from cross referencing with LinkedIn profiles and estimated age for youth who tweeted about a particular grade level in school. However, approaches that combine the use of age-annotated data are still in their infancy, and these methods have not been widely applied to predict age of Twitter users[22].

## 1.4 Objectives

The primary goal of this work is to predict the age of Twitter users in Spanish, focusing on the region of Uruguay and part of Argentina.

To achieve it, according to what it was mentioned in the Related work section 1.3), we need to focus on the tasks of data collection and the extraction of meaningful features.

Therefore, the first important objective of this thesis is to generate a corpus in Spanish, labeled with the user age and gender, and publish it online to be available for other investigations. As the related work shows, there are not many datasets in Spanish, and the few that exist, contain about 200 users, and this does not allow NLP and machine learning algorithms to make meaningful and interesting conclusions.

The second major objective is to extract significant features for the task of age prediction. By studying the characteristics of Twitter users and their tweets, we aim at providing insight into the types of users, how the network is organized, and the important features that distinguish these users. To that end, the analysis will study some features proposed by other authors to

verify if they also apply to Spanish users.

We want to test the performance of different types of features: user meta-data (i.e., quantity of tweets, quantity of friends, etc.), stylometric features of tweets (for instance: quantity of emoji per tweet, quantity of uppercase letters used per tweet, among others) and NLP features extracted from tweets and subscription lists (unigrams, bigrams, trigrams, etc.).

We will not consider semantic and context features, since prior work [16], [3] demonstrate that they were outperformed by stylistic and syntactic features.

We also include novel stylistic features that were not explored in previous research, that capture cross-referring data from Twitter and other social networks like Facebook, LinkedIn and Instagram.

Finally, we want to train different models to conclude which one works better in our context and for our custom features.

## 1.5 Contributions

The contribution of this thesis is threefold.

First, we generated a corpus labeled with age and age range, that will be published online so it can be used by fellow investigators to use it as valuable input for research. To the best of our knowledge, there are no bigger datasets in Spanish annotated with age with more than 200 entries. In fact, one of PAN Clef organizers mentions that this is one of the reasons to organize a task for Age Profiling. He strongly encouraged us to build a new one and share it with the community, since “it is an invaluable asset”.

Secondly, we present a process pipeline to keep generating new corpus automatically. This way, we can keep increasing the size of our dataset without the need of manual intervention, the longer it runs, the more labeled users are collected. The process is composed by many tasks: a twitter streamer that automatically gets new twitter users from the network, a task to gather the age from the information in the profile (if available), tasks to populate interesting features such as quantity of URLs per tweet, quantity of emojis per tweet, flags to model if the user has other networks linked to his Twitter profile (i.e., Facebook, LinkedIn, Instagram, Snapchat). We execute other

tasks to retrieve the tweets in Spanish of the user, as well as the subscription lists the user belongs to. Also, a task connects to Microsoft Face API to extract the predicted age and gender by analyzing the profile picture.

Thirdly, we obtained a set of features and a classifier through different machine learning and NLP techniques, that are able to predict the age of the Spanish-speaking Twitter users with better accuracy than previous works in a dataset almost 5 times larger in number of users.

## 1.6 Outline of the thesis

The first step towards Age Prediction in Twitter is to gather a corpus with users annotated with the age. To that end, in Chapter 2, we will present three approaches to generate a labelled corpus of Twitter users. This chapter ends with the presentation of TweetLab, a customized pipeline to extract and label Twitter users in Spanish, as well as a mention to the underlying software tools used to create it.

Later, in Chapter 3 we provide a qualitative analysis of three types of features: user metadata, stylometric features in tweets and Natural Processing features (for Tweets as well as Subscription Lists)

These attributes will be the input of the Experiments presented in Chapter 4. The chapter begins with an overview of each Classifiers used to predict age, and next the four experiments driven are explained thoroughly.

To conclude with the investigation, Chapter 5 presents the conclusions and Future Work.



## 2 Corpus Generation

### 2.1 First Approach

As a first approach, we built a prototype solution stepping on an already existing labeled corpus in Spanish. For this, we used the PAN Clef dataset for Author Profiling task corresponding to year 2016 [8].

The focus of 2016 task was about age and gender identification. The dataset contained Twitter tweets in English, Spanish and Dutch. Due to Twitter’s privacy policy, PAN Clef did not provide tweets directly, but only the tweet text and the URL referring to it. So it was responsibility of the investigator to download them if necessary. Then, the performance of the author profiling solution was ranked by accuracy.

The Spanish dataset consisted in 250 users, classified into 5 age ranges: 18-24, 25-34, 35-49, 50-64 and 65-xx.

AGE	USERS
18-24	16
25-34	64
35-49	126
50-64	38
65-xx	6

Table 2.1: Age distribution in PAN Clef dataset

Since the dataset contained the tweet text only, we decided to do an enhancement consisting in retrieving the whole tweet entity using the Twitter API (see section 2.4.2), as a way to obtain more information to be used as

new features. Also, the PAN Clef dataset contained a maximum of 1,000 tweets, so this improvement would allow us to collect a larger number of tweets per user.

The first approach resulted in the following processing pipeline:



The dataset was formed by 250 XML files, each one representing a Twitter user. The XML file looked like this:

```

▼<author age_group="xx" gender="xx" lang="ES" type="twitter" url="https://twitter.com/EtcoMusic">
  ▼<documents count="224">
    ▼<document id="208854874207694848" url="https://twitter.com/EtcoMusic/status/208854874207694848">
      ▼<![CDATA[
        <a href="/jcsuau" class="twitter-atreply pretty-link js-nav" dir="ltr" data-mentioned-user-id="118103365" ><s>@</s><b>jcsuau</b></a> de película!! muxxxxx
      ]]>
    </document>
    ▼<document id="208855165711822848" url="https://twitter.com/EtcoMusic/status/208855165711822848">
      ▼<![CDATA[
        El día tiene k estar lleno de inspiraciones mágicas y vibrantes si no no me levantoooo!!
      ]]>
    </document>
  
```

In this example, the user “EtcoMusic” contained 224 tweets. For each tweet, the URL to download it is provided.

Hence, the first task **Extract screen names** was in charge of getting Twitter **usernames** from these XML files.

The next step **Extract tweets** would receive the list of screen names extracted, and retrieve the last 3,200 tweets using Twitter API as mentioned before (see Section 2.4.2).

After executing this step, we obtained information of 226 users from the original 250, this might be because it is information from 2016 and some users might have changed the account privacy, or even deleted their account.

The age and gender labelling of the users was made by mapping these XML files against another file **truth.xml**, that contained the name of the XML file, and its corresponding age and gender:

cede0dea9347713bc0b3b0aaf4e42020:::FEMALE:::35-49

7739f4c59c29b7406090fda3670fe2be::FEMALE::18-24  
ffe80a44666c1d12f61d0ad47499eb1e::MALE::50-64

This was the responsibility of module **Extract Age From File** that ran afterwards, to get the corresponding age from `truth.xml` file.

Finally, we stored every user in a collection in a MongoDB database (see Subsection `refMongoDBdatabase`).

After generating the extended dataset, we used it as input of different machine learning models. Overall, we obtained an accuracy similar to the 48% accuracy published by Arroju et al [10] in PAN Clef 2015 competition. Therefore, we concluded that the extraction of more tweets did not improve the prediction.

There could be several reasons for not obtaining any improvement. First, it might be because of the size of the dataset, which is still small (only 226 users to classify). Second, it might also be due to the unbalanced nature of the dataset. In fact, almost half of the dataset belongs to one class, despite the fact that there are four more classes.

As a consequence, we understood that in order to improve the accuracy of the prediction, we needed to expand the dataset by collecting more users.

## 2.2 Second approach using web scraping

Twitter allows the user to link a Facebook account to his Twitter profile, providing the option of automatically posting the tweets in Facebook as well. So, even if the age is not an attribute in the Twitter profile, it does appear on the Facebook profile. This is why it is interesting to try to access the linked Facebook profile of an account.

The age field might be public or private, if the user selected to keep it private, only the users' friends will be able to see this information. Nevertheless, some users have that information available for everyone, which seemed a good way to build our corpus.

However, this is not an easy task, because of the limitations of the Facebook Graph API. We tested several approaches (see Appendix 7) but the

results were not satisfactory.

The final approach developed performed web scraping in Facebook. It was very tied to the way the source code is created. Hence, a minimal change in the front-end might provoke this process to stop working, so it does not work as a long term solution to label users.

Moreover, it is very unusual for Facebook users to leave the birthdate public, since by default it is only visible for the user's friends.

Likewise, we encountered difficulties related to the need for the page to load before performing the request, the lack of IDs inside the HTML that makes it really hard to track and get the elements of interest, and the different representations of the same information in the profile.

Finally, there is still the need to work with a streamer to collect more users to perform this logic and get the age from Facebook, which leads to a very difficult, complex and not performant process, as we mentioned.

## **2.3 Final approach: TweetLab**

### **2.3.1 Overview**

After obtaining this poor results, further investigation was driven to find another way to increase the dataset size.

To that end, we developed a new software solution for data collection, named TweetLab. Our tool is a customized pipeline that aims at extracting and labeling Twitter users in Spanish. Its starting point is a Twitter streamer that serves as input to automatically get users that have specified their age in the profile description.

Also, it includes tasks to gather meaningful information for feature extraction, such as information about the user, his subscription lists, and text metrics (i.e., quantity of uppercase letters per tweet, quantity of emojis per tweet, etc.).

In addition, it interacts with an external API to get the gender and age predicted from the profile picture, as a way to be able to evaluate the accuracy

against different market options that make similar predictions.

### 2.3.2 Design of the solution

Here we present the architecture of TweetLab by giving an overview of each of its tasks.



#### Twitter Streamer

It all begins with the **Twitter Streamer**. This module steps on the Twitter streamer provided to us by IDATHA<sup>1</sup>. It is itself based on Twython<sup>2</sup>, the streamer coded in Python provided by Twitter (see Section 2.4.2).

In our particular case, we want to keep only the Spanish tweets. Twitter API does not have any call to filter by language, but it does have the possibility to define a bounding box (area defined by two longitudes and two latitudes) to specify the tweet locations. Hence, we created a bounding box to include Uruguay and part of Argentina, since both countries use the same variant of Spanish, often called “Rioplatense”.

---

<sup>1</sup><http://www.idatha.com/acerca.html>

<sup>2</sup>[https://twython.readthedocs.io/en/latest/usage/streaming\\_api.html](https://twython.readthedocs.io/en/latest/usage/streaming_api.html)



We customized this streaming task, since we are not interested in storing streamed tweets, but in extracting the author to store it's information in an auxiliary collection in the database.

## Analyze Bio

After exploring different profiles, we found that, even if the age is not a field present, many users actually wrote it in their description. For instance: *"17 añosss // vivir- reir - soñar // concordia"*. Of course, some labelling might be incorrect, such as those cases where the user specifies the years of experience in a company.

Later, we examine this collection, and try to identify the users that have stated their age in the profile description, so we store the age as well.

Using the Twitter streamer we collected 40,072 users, and we were able to extract the age of 2.3% of them from their biography. This shows the complexity of the labelling task, and justifies the need of a streamer that can run for long periods of time.

## Extract users and tweets

Then, for each user with age, we run the task called **Extract Users and tweets**. This module basically queries the Twitter API (see section 2.4.2) and retrieves the last 3,200 tweets as well as user information such as friends count, name, profile picture, language, among others. It is worth mentioning that not only the tweet text is obtained, but also metadata information (i.e., if it is a re-tweet, if it contains URLs, user mentions and hashtags, the language, the location, etc.).

In few cases, we needed to reduce the size of the tweet collection for a user, since by default documents cannot exceed 16Mb in MongoDB. This is an interesting fact, and we can take it into account to enhance the design of the solution in future versions by storing the tweets in an independent collection, and store the user id to be able to merge both collections when necessary.

After getting the user tweets, we only kept the messages that were either broadcasts or replies and ignored the ‘re-tweet’ (RT) messages since the authorship of this text cannot be attributed to the user, as suggested by Rao and Yarowsky [5]. Also, we made sure that every tweet stored is in Spanish, since many users might write some posts in another language, or they might live in the frontier and speak Portuguese. This way, we avoid the noise in the dataset introduced by tweets in another language.

Since we already had 226 users from PAN Clef dataset, we did the same pre-processing and added information about the user and the tweets as well. This way, we enriched those users and now we have more information to extract other features.

Also, the age obtained from the description gets stored, and it is also mapped to an age range. In the beginning, we started using PAN Clef age ranges (18-24, 25-34, 35-49, 50-64, and 65-xx). But when analyzing the information, we found out that many accounts belonged to users younger than 18, this is why we decided to add an extra class for users between 10 and 17.

To avoid rare values, we did not consider users with age less than 10 or greater than 99, since those values will probably not refer to the users age (i.e years of experience in something, age of a son, etc)

Likewise, when we looked at the distribution between age ranges, we discovered that only the 0.69% of users belonged to class 65-xx, so we opted for merging classes 50-64 and 65-xx into one new class 50-xx. This is because this significant unbalance in the dataset might provoke biased predictions. Of course, the module could be customized at will to generate other age ranges.

In conclusion, the final version of the dataset included the following age ranges: 10-17, 18-24, 25-34, 35-49 and 50-xx.

It is worth mentioning that in our experiments, every user is labeled with the age range since we run classification algorithms. Nevertheless, we also store in the database the exact age extracted from the profile, so this dataset can be used as an input of regression algorithms as well.

## **Extract features**

The set of labeled users and tweets are fed into the module **Extract features**. Here, metadata and stylometric features are gathered and stored as user attributes as well. This phase is explained in detail in Chapter 3.

## **Analyze profile picture**

As another enhancement compared to the previous version, we added a new task to predict the age and gender from the profile picture using Microsoft Face API<sup>3</sup>. This service detects one or more human faces in an image and analyzes face attributes to later make machine learning-based predictions of facial features. The face attribute features available are: Age, Emotion, Gender, Pose, Smile, and Facial Hair.

The platform was able to predict the age and gender of the 39% of the users in the dataset based on their profile picture. This might be due to many reasons, such as the image not being clear enough, not being of a human, too many faces in the picture, among others.

In our case, we stored the gender and age, as a way to compare the results obtained by different models. We will not use the predicted age as a feature for two main reasons. First, we want to focus on the text analysis. Second,

---

<sup>3</sup><https://azure.microsoft.com/en-us/services/cognitive-services/face>



since it is based on the profile picture, and if a sexual predator uses a picture belonging to somebody else, it will affect the prediction.

### **Extract Subscription Lists**

We included this task to obtain the subscription lists of the user, and use it as another feature for NLP analysis.

### **Store in database**

Finally, we store the user enriched data and his tweets in a new collection, and mark the user as exported in the auxiliary one, so it does not get imported twice.

## **2.4 Underlying Software Tools**

### **2.4.1 Document Database**

MongoDB is a free and open-source document database that stores data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time. This is fundamental because Twitter API returns JSON, and not all requests to Twitter API would return the same fields. In fact, if the API changes (i.e., new fields added in the returned JSON) then it is not necessary to change the structure of the database store.

Also, every document type (i.e., tweet, user, subscription list) may contain several levels of depth. In this way, we avoid the need to create all the columns like traditional RDBMS systems (i.e., Oracle, MySQL, MS MSQL Server etc.)

Finally, PyMongo is a Python distribution containing tools for working with MongoDB, and makes it very easy to query the database from a Python script.

## 2.4.2 Twitter API

The standard (free) Twitter APIs consist of a REST API and a Streaming API. The Streaming API provides low-latency access to tweets<sup>4</sup>.

With the exception of the Streaming API, the Twitter API endpoints attempt to conform to the design principles of Representational State Transfer (REST). Twitter APIs use the JSON data format for responses (and in some cases, for requests).

In particular, Rest API usage is rate-limited. Clients may access a theoretical maximum of 3,200 statuses via the page and count parameters for the “user timeline” REST API methods. Other timeline methods have a theoretical maximum of 800 statuses.

Requests for more than the limit will result in a reply with a status code of 200 and an empty result in the format requested. Twitter still maintains a database of all the tweets sent by a user. However, to ensure performance, this limit is in place on the API calls.

Another key thing to mention is that there are Twitter API libraries for almost any language: Java, PHP, Python, Javascript, Ruby, C, .NET, among many others.

In our case, we both use the Streaming and the Rest API. Since our development platform is Python, in order to connect to the Twitter APIs we use **Twython**, a pure Python wrapper for the Twitter API, that supports both normal and streaming modes. It allows the user to query data for user information, Twitter lists, timelines, direct Messages, etc.

### TwythonStreamer

We use **statuses/filter** request to filter realtime tweets. It returns public statuses that match one or more filter predicates. Multiple parameters may be specified which allows most clients to use a single connection to the Streaming API.

It allows you to filter the tweets you want to stream by different criteria:

---

<sup>4</sup><https://developer.twitter.com>

**follow** : A comma separated list of user IDs, indicating the users to return statuses for in the stream.

**track** : Keywords to track. Phrases of keywords are specified by a comma-separated list.

**locations** : Specifies a set of bounding boxes to track. We used this parameter to filter the tweets coming from Uruguay and Argentina.

## Twython Rest API

We first use it to gather user information through the request **lookup\_user** which returns fully-hydrated user objects for up to 100 users per request, as specified by comma-separated values passed to `user_id` and/or `screen_name` parameters.

It is rate-limited, allowing to perform up to 900 requests in every 15 minutes time window. Native re-tweets of other statuses by the user are included in this total, regardless of whether `include_rts` is set to false when requesting this resource. This is why before storing the tweet collection of the user, we discard all re-tweets since they have a different author.

Later, we perform the request **statuses/user\_timeline** to return up to 3,200 of a user's most recent tweets posted by a user indicated by the `screen_name` or `user_id` parameters. It has the same rate limit of 900 requests every 15 minutes.

The returned timeline corresponds to the timeline seen as a user's profile on `twitter.com`.

User timelines belonging to protected users may only be requested when the authenticated user either "owns" the timeline or she is an approved follower of the owner. In our case, if the user had private tweets, it would not be stored in the dataset, since our investigation focuses on text analysis.

Last but not least, we perform a call to **lists/subscriptions** to obtain a collection of the lists the specified user is subscribed to. The default maximum is 20 lists per page. It also has a rate limit, allowing to perform 15 requests in a 15 minutes time window. Because of this, it was one of the tasks that took longer to execute.

## 3 Feature Extraction

This chapter will provide a deep analysis on the features extracted to predict the age of the users.

In all cases, the information was extracted from the dataset obtained with the custom streamer pipeline.

In order to run the experiments, we took a snapshot of the database right before starting the feature extraction tasks. As a result, we collected 1,156 users labeled, with the following distribution:

AGE	USERS
10-17	237
18-24	587
25-34	132
35-49	141
50-xx	59

Table 3.1: Age distribution in dataset

As we can see, this new streamer pipeline allowed us to obtain a dataset 5 times bigger than the original one from PAN Clef. The most important benefit of this approach is that the longer it runs, the bigger the dataset becomes, without the need of human interaction. In fact, if we leave it running on a computer for a week, we might duplicate the current size. Also, the bounding box can be changed to include other countries that speak Spanish as well.

From this dataset we extracted three different types of features to work with: user metadata, stylometric and NLP features.

### 3.1 User metadata features

This type of features, also referred as “online behavior” by other authors [3], consists of characteristics of the Twitter profile that are independent of the contents of the tweets. It includes some Twitter specific attributes, such as **favourites count** and **tweets count**, but also contains fields that are typically available in any other social network like **followers count** or **friends count**.

Feature	10-17	18-24	25-34	35-49	50-xx
Friends count	665	971	1725	1694	3048
Tweets count	13010	22010	18927	13472	22251
Followers count	875	2056	10698	29257	307828
Favourites count	5114	6553	3727	3194	2377

Table 3.2: Average online features per age range

Among the other online behavior features, average friends count seems to be higher for older users. The number of tweets fluctuates, but in general decreases as users get younger which is as one would expect unless younger people were orders of magnitude more prolific than older people, similar to what Rosenthal [3] concluded.

However, when analyzing these values, we need to be cautious, because if one user has a massive amount of tweets but the rest does not, this rare value will affect the average and it will not reflect the majority behavior. However, we only calculate the average to display some metrics, but averages are not considered as features because of this reason.

Similar to our findings, Rao et al [23] examined profile statistics, such as the number of followers, the number of profiles the person followed, and the ratio of followers-to-following, but they found “no exploitable differences” in the distributions of the demographic characteristics examined.

Furthermore, we also analyzed the user profiles to extract other social networks linked to the Twitter account. As a result, we created four extra features that indicate the presence of a linked **Facebook**, **Snapchat**, **Instagram**, and **LinkedIn** account.

Social Network	Users
Facebook	175
Snapchat	107
Instagram	257
LinkedIn	17
None	712

Table 3.3: Users with social networks linked to Twitter

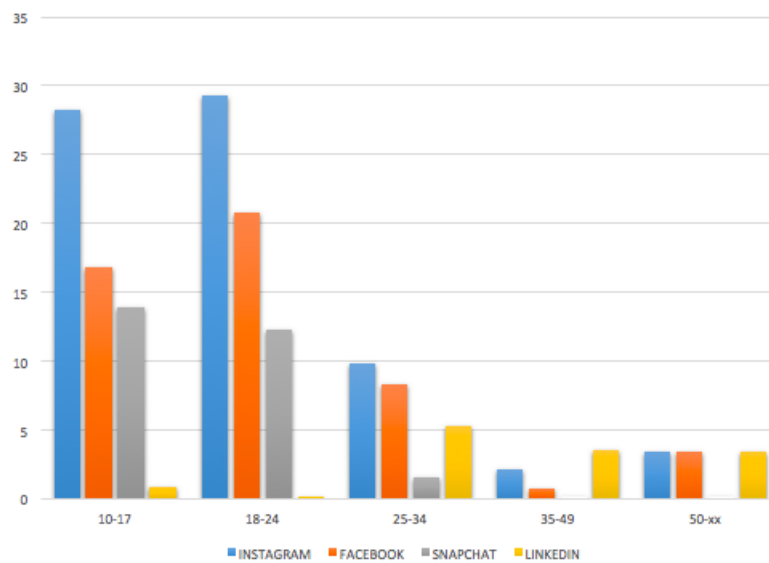


Figure 3.1: Percentage of users with linked social networks per Age group

Many interesting observations can be made from the chart in Figure 3.1. For instance, younger users tend to link more social networks to their Twitter profile than older users. However, in the case of LinkedIn, we identify an increase with the age. This makes sense since it is a business network, in which younger users might not be interested.

It is also worth mentioning that Facebook is losing appeal among teens and young adults. At the same time alternative social apps like Snapchat and Instagram are seeing a growth in the same youth demographic, suggesting younger users are favoring newer and more visual communications platforms. Both platforms have found success with this demographic since they are more aligned with how they communicate using visual content<sup>1</sup>.

<sup>1</sup><https://techcrunch.com/2017/08/22/teens-favoring-snapchat-and-instagram>

This is a big change on the board, since Facebook, who was created in 2004 is being replaced by newer networks created five and six years later. In fact, in the case of Snapchat, the decrease with the age is stronger. We do not find any user older than 35 having a linked Snapchat account. This makes sense because it is not a widespread network for older users

We only store a boolean instead of the URLs since we are not interested in cross referencing the profiles.

Finally, we add an extra feature with the help of the Microsoft Face API<sup>2</sup>. It is the predicted **profile pic gender** based on the profile picture. However, this is not always possible (i.e., image not clear enough, image of a non human element, many faces in the picture, etc), so we will store -1 to indicate that gender prediction was not possible for that user.

Predicted Gender	Users
Male	348
Female	362
Undefined	446

Table 3.4: Predicted gender by profile picture

As we can see, the API failed to tell the gender of about 40% of the users. Moreover, we can also note that the users successfully labeled are almost equally distributed between genders.

## 3.2 Stylometric features in tweets

In this study we consider the following features: **qtyMentions**, **qtyHash-tags**, **qtyUrls**, **qtyEmojis**, and **qtyUppercase**.

---

<sup>2</sup><https://azure.microsoft.com/en-us/services/cognitive-services/face>

Feature	Meaning
qtyMentions	Quantity of user mentions per tweet
qtyHashtags	Quantity of hashtags used per tweet
qtyUrls	Quantity of URLs referenced per tweet
qtyEmojis	Quantity of emojis, emoticons and smileys per tweet
qtyUppercase	Quantity of uppercase letters present (without the URLs)

Table 3.5: Sytlometric features extracted

These Lexical-Stylistic features were computed using the text from all the tweets in the dataset. They were all normalized by tweet to keep the numbers consistent between users regardless of whether the user wrote one or many tweets in his/her profile.

Feature	10-17	18-24	25-34	35-49	50-xx
qtyMentions	0.37	0.33	0.78	0.78	0.87
qtyHashtags	0.14	0.09	0.42	0.43	0.38
qtyUrls	0.12	0.12	0.36	0.48	0.50
qtyEmojis	0.69	0.60	0.36	0.26	0.13
qtyUppercase	2.97	2.55	4.46	4.12	4.41

Table 3.6: Average lexical-stylistic features per age group

We observe that the number of emoticons (Figure 3.6) decreased as users become older. This was an expected result, since according to millenials, “GIFs and Emojis Communicate Their Thoughts Better Than English”. Visuals like emojis can make up for the useful cues that are often missing from digital chit-chatting: the raise of an eyebrow, the shrug of the shoulders, the rolling of an eye. But a new survey reveals that many people believe those visuals are not just helpful for adding clarity in text and mobile messages. They actually feel that they can better express themselves through these digital tools than through old-fashioned English<sup>3</sup>.

In a survey conducted by Harris Poll, 36% of millennials ages 18 to 34 who use “visual expressions” such as emojis, GIFs and stickers say that those images better communicate their thoughts and feelings than words do. That is more than twice the amount of people over the age of 65 who say the

<sup>3</sup><http://time.com/4834112/millennials-gifs-emojis>



same. Roughly a quarter of people in the age groups between those two demographics feel that images can paint a clearer picture than words.

On the other hand, the quantity of uppercase letters, URLs and user mentions tend to increase with age.

## 3.3 Natural Language Processing features

### 3.3.1 Tweets

To determine how important language features are in classifying users into age categories, we created a set of variables that only require a user's tweet text. Firstly, we pooled together the tweets text of every user and later we converted into a bag-of-words (BOW) vector space model.

Since BOW models calculate term frequencies without context of neighboring words, and in order to incorporate additional context into our model, we created bigram and trigram variables that combine adjacent terms.

In order to reduce the dimension of the dictionary, we performed a pre-processing task to convert to lowercase every tweet, and we set the stopwords list for the vectorizers<sup>4</sup>.

NLP feature extraction was done using Scikit Learn<sup>5</sup> and NLTK<sup>6</sup>.

### Stopwords

As a way to reduce the number of features, we can use stopwords to filter out some common words from the text to be processed.

In order to create our stopwords set, we used the collection provided by NLTK due to the fact that Scikit Learn does not have stopwords for Spanish.

Later, we want to expand this list with the most frequent words that

---

<sup>4</sup>Recall that we keep track of the number of uppercase letters in a tweet in the stylistic feature `qtyUppercase`.

<sup>5</sup><http://scikit-learn.org>

<sup>6</sup><http://www.nltk.org/>

appear in all age groups meaning they do not identify any group in particular.

However, the highest frequency words collected from a corpus may not be a good distinguishing feature, since they might contain contextual information. Also, different age groups may use mutual n-grams, but one age group may use them more often than another [9].

Instead, we analyzed the words that are highest occurring in each age group, and compared the results.

In order to do that, we used `TfidfVectorizer` from `Scikit` package which converts a collection of raw documents to a term frequency–inverse document frequency (TFIDF) matrix features [24]. It is equivalent to `CountVectorizer` followed by `TfidfTransformer`. TFIDF is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. The TFIDF value increases proportionally to the number of times a word appears in the document, but is often offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general. Nowadays, TFIDF is one of the most popular term-weighting schemes <sup>7</sup>.

In our case, we used it to extract the 200 most important words per age group. Then, if the word appeared in three or more age groups, it would be added to the stopwords list, hence, it would not be considered a feature. To see the full list of stopwords, refer to Appendix 7.

10-17	18-24	25-34	35-49	50-xx
si	si	gracias	gracias	si
hoy	hoy	si	si	ahora
mejor	siempre	hoy	bien	gracias
siempre	hacer	hacer	hoy	hoy
vida	quiero	años	día	mejor
ahora	mañana	día	año	ser
bien	ser	buena	años	año
feliz	bien	hace	cada	años
vos	mejor	ser	hace	bien
voy	va	siempre	mejor	día

Table 3.7: 10 Most Frequent words across the age groups

---

<sup>7</sup><https://en.wikipedia.org/wiki/Tf\0T1\textendashidf>



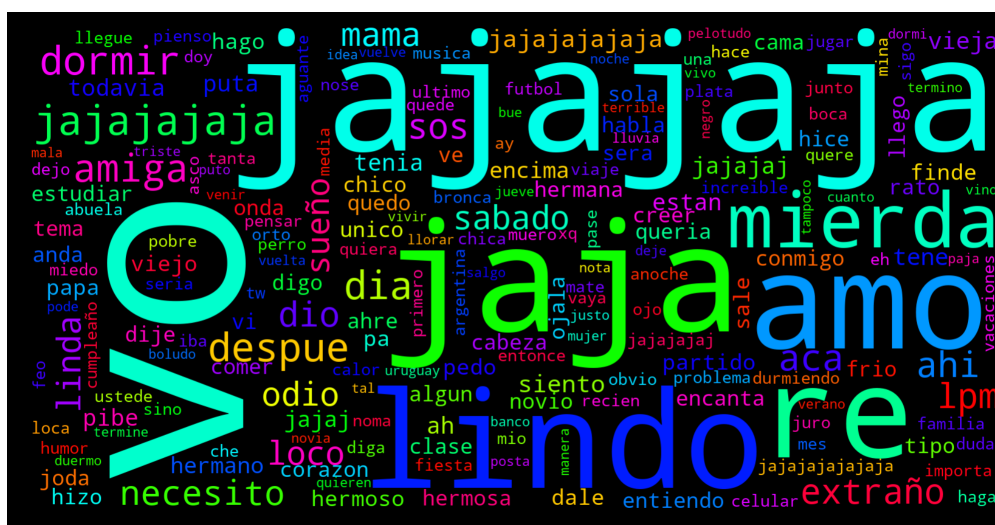


Figure 3.3: Word cloud for Age group 18-24



Figure 3.4: Word cloud for Age group 25-34



Figure 3.5: Word cloud for Age group 35-49



Figure 3.6: Word cloud for Age group 50-xx

Firstly, if we take a closer look to the word cloud corresponding to the youngest group, we see that the most frequent words are variants of the laugh onomatopoeia (“jaja”, “jajaja”, “jajajaja”). Also, we see some slangs such as “vo”, “lpm”, “re”. This lies on the fact that younger users tend to use more informal language. It is worth mentioning the occurrence of words like “mama”, “dormir”, “cama”, “papa” since the majority are children and young teenagers that do not work yet.

The reality changes when we focus on older users, since they refer to businesses, companies, the government, politics, e-business, pymes, science, technology, among others.

As we can see, the topics of the tweets vary significantly across the different age groups.

### 3.3.2 Subscription Lists

We used CountVectorizer from Scikit package to extract bigrams and trigrams from Subscription Lists names. In this case we only considered the default Spanish stopwords inside NLTK package and not custom ones, since it is a very small dataset with smaller texts, so there is no need to prune the feature tree with custom stopwords. To limit the vocabulary dimensionality, we only considered the 5,000 most frequent words.

Related to subscription lists, we also design word clouds since they come handy to understand the interests of the users across the different age ranges.

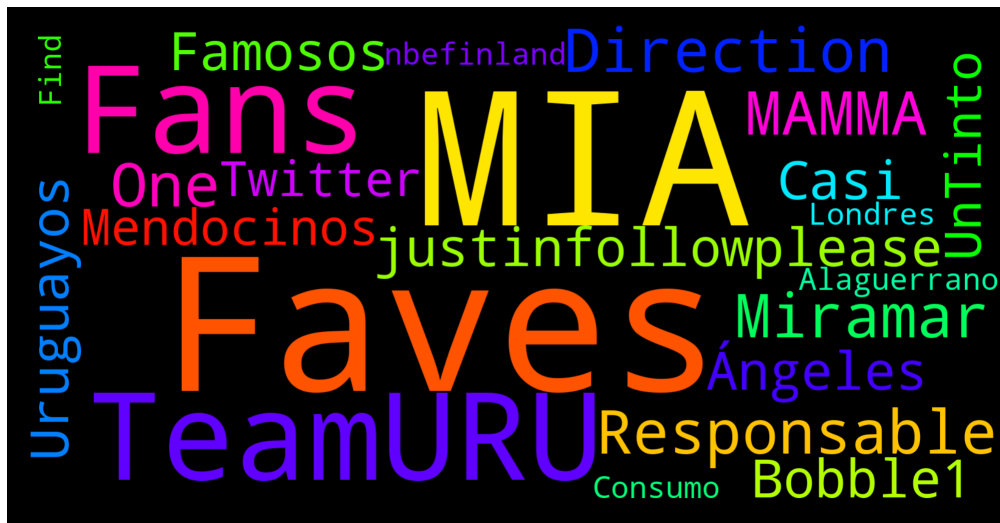


Figure 3.7: Word cloud for Age group 10-17



Figure 3.8: Word cloud for Age group 18-24

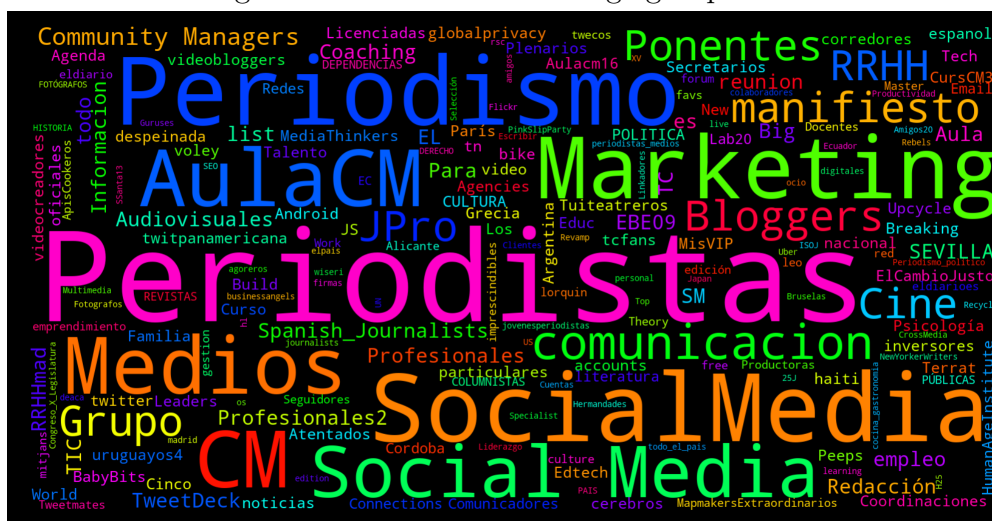


Figure 3.9: Word cloud for Age group 25-34

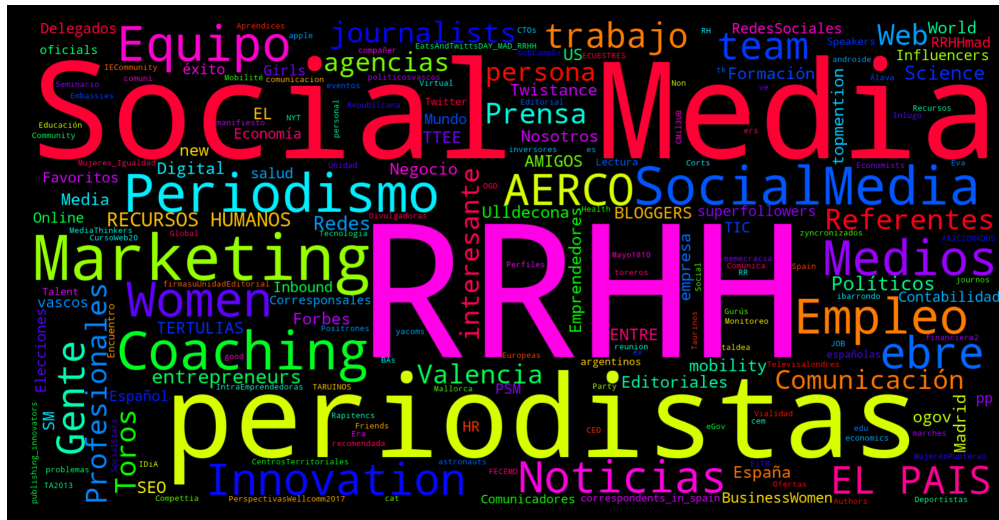


Figure 3.10: Word cloud for Age group 35-49

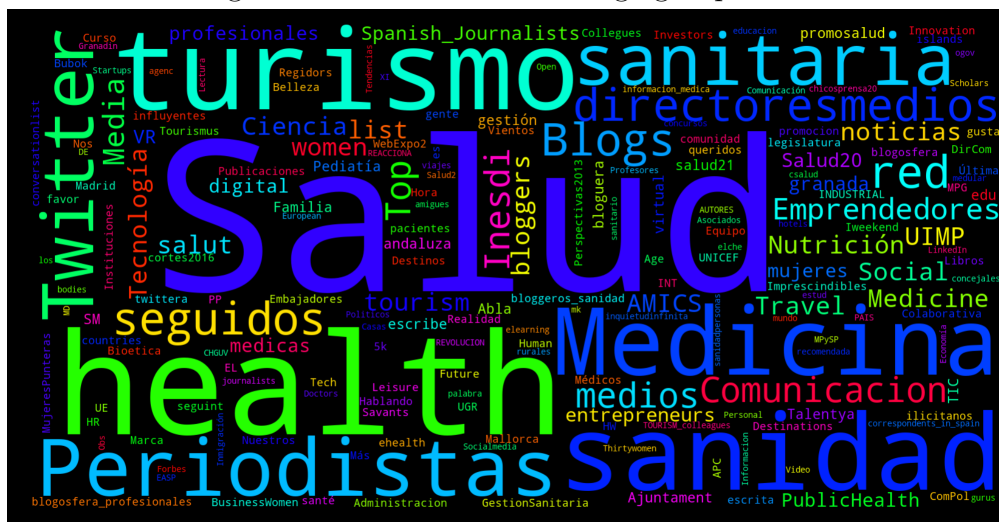


Figure 3.11: Word cloud for Age group 50-xx

If we take a closer look at younger users (10-17 and 18-24), their mostly subscribed to topics like the pop band One Direction and its members (Louis Tomlinson, Harry Styles, Liam Payne, Zayn Malik). We also see words that indicate location, such as Uruguay and Argentina, which also makes sense since we extracted users from these countries. We also see “Fans” and “Casi Angeles”, which is the name of a very famous series intended for young audiences.



Next, it is worth mentioning the big change of topics when we move to the word cloud corresponding to ages 25-34 and 35-49. Here, most users are interested in Marketing, Human Resources, Journalism, Innovation and Social Media. This also makes sense since probably these topics will be related to their professional careers.

Finally, the word cloud that represents the interests of the oldest users (50-xx) shows that the users are concerned about Health topics (Medicine, Nutrition, among others). Also, the word “turismo” appears a significant amount of times.

## 4 Experimental evaluation

In this chapter we present the different experiments carried out and the results obtained. As we mentioned earlier, we treat age prediction as a multi-class classification problem.

We compare the performance of our model with a majority baseline, which consists in setting the majority class occurring in the dataset as the predicted class for all instances in the test data set. We also compare it against the age predicted by the Microsoft Face API made through the analysis of the profile picture.

### 4.1 Classifiers

To model age, we tested four different supervised learning models: Multinomial Naive Bayes, Support Vector Machines (SVM), Stochastic Gradient Descent (SGD) and Random Forest. We also included a dummy classifier to assess majority baseline performance.

#### Multinomial Naive Bayes

Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

Multinomial Naive Bayes is a specific version of Naive Bayes, designed specifically for text documents. While Naive Bayes models a document as the presence and absence of particular words, Multinomial Naive Bayes models the word counts and adjusts the underlying calculations to deal with it. [25].

## Support Vector Machines

Support Vector Machines (SVMs) are a set of supervised learning methods used for classification, regression and outliers detection. They are effective in high dimensional spaces, even if the number of dimensions is greater than the number of samples.

Given a set of data samples, SVM finds a linear separation between the differently labeled datasets that maximizes the margin between the two classes. Then it can classify unseen samples by determining which side of the separating hyper-plane they are in. But since datasets are not always linearly separable, SVM uses a kernel function to map all samples to higher-dimensional spaces, which are then checked for linear separability [11].

We opted to use Support Vector Classifications (SVC), since it is one of the classes capable of performing multi-class classification on a dataset using a one-vs-one (OVO) scheme <sup>1</sup>.

## Stochastic Gradient Descent

It is a simple and very efficient approach to discriminative learning of linear classifiers under convex loss functions.

SGD is usually applied to large-scale and sparse machine learning problems often encountered in text classification and Natural Language Processing tasks. Since that data is sparse, the classifiers in this module easily scale to problems with more than  $10^5$  training examples and more than  $10^5$  features <sup>2</sup>.

## Random Forest

Random forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a set of decision trees at training time and outputting the class that is the most predicted class among all trees. Random forests are a way to avoid the potential overfitting incurred by a single decision tree [26].

---

<sup>1</sup><http://scikit-learn.org/stable/modules/svm.html>

<sup>2</sup><http://scikit-learn.org/stable/modules/sgd.html>

## Hyperparameter tuning

Hyperparameter tuning was performed on the models (except Multinomial Naive Bayes since it does not have parameters) to explore the feature space and experiment with different modeling assumptions.

We employ a holdout-validation approach to our data and analysis, according to which we divide our dataset into 2:

- A training dataset containing the 80% of the users for parameter estimation and to train the models.
- A test dataset containing the remaining 20% of the users, to generate the final model evaluation metrics.

## 4.2 Experiments

We carried out four experiments:

**Experiment 1** : Prediction between the five age groups

**Experiment 2** : Prediction for cyber pedophilia detection

**Experiment 3** : Prediction with a balanced dataset

**Experiment 4** : Our Predictions vs. Microsoft Face API

Each experiment was replicated using four different sets of features:

**Feature set 1** : User metadata and tweet stylometric features

**Feature set 2** : Tweet n-grams

**Feature set 3** : User metadata, tweet stylometric features, and tweet n-grams

**Feature set 4** : Subscription Lists n-grams

For each experiment, we ran several tests using different features, to see which ones are the most adequate and meaningful to solve this task.

The original dataset is composed by 1,156 users, from which 925 (80%) belong to train set and 231 (20%) belong to test set.

To see the distribution of the different age groups, refer to Table 3.1.

The original dataset is used in Experiment 1. In experiments 2, 3, and 4, we use a subset of this dataset since some information is not present in all users.

#### 4.2.1 Experiment 1: Prediction between the five Age Groups

The objective is to predict the age of the users from the following age groups: 10-17, 18-24, 25-34, 35-49 and 50-xx.

##### Feature set 1: User metadata and tweet stylometric features

With these features, we obtained the results shown in Table 4.1.

Classifier	Accuracy
M. Naive Bayes	0.40
Random Forest	0.60
SVM	0.51
SGD	0.46

Table 4.1: Accuracy using 10-fold cross validation for Feature Set 1 - Experiment 1

One of the advantages of the Random Forest Classifier is that we can display the features ranked by importance according to this predictor:

Feature	Score
qtyUrls	0.132446
qtyEmojis	0.114787
qtyMentions	0.114541
followers_count	0.108915
tweets_count	0.103886
qtyHashtags	0.101795
qtyUppercase	0.092768
favourites_count	0.089030
friends_count	0.084164
profile_pic_gender	0.029943
instagram	0.013759
facebook	0.009583
snapchat	0.003721
linkedin	0.000663

Table 4.2: Importance of Features - Random Forest

According to the table above, the most meaningful attribute to predict the age is the quantity of URLs in the tweets. Next, we have qtyEmojis which is aligned to the fact that this quantity increased as users get younger (see Table 3.6).

Moreover, the least important ones are snapchat and linkedin. If we analyze thoroughly the dataset, we can see that this might be due to the fact that a very small proportion of users have their twitter profile linked with an account of these types. In fact, only 1.5% of the labeled users have LinkedIn in their profile.

Finally, if we compare the classifiers, we conclude that Random Forest outperformed the rest of the classifiers when considering user metadata and tweet stylometric features, achieving an **accuracy of 60%**.

## Feature set 2: Tweet n-grams

For this experiment we worked with unigrams, bigrams and trigrams extracted from the tweets.

In order to extract these features from the tweets text, we CountVector-

izer<sup>3</sup> and TfidfVectorizer (which is Equivalent to CountVectorizer followed by TfidfTransformer)<sup>4</sup>. The best performance was achieved using TfidfVectorizer.

In the first iteration we worked with the 5,000 most meaningful n-grams, obtaining the following accuracies:

Classifier	Accuracy
M. Naive Bayes	0.59
Random Forest	0.59
SVM	0.62
SGD	0.62

Table 4.3: Accuracy using 10-fold cross validation for Feature Set 2 - Experiment 1 with 5,000 features

Later, we performed a second iteration to verify if by considering more features the accuracy would improve.

Classifier	Accuracy
M. Naive Bayes	0.56
Random Forest	0.59
SVM	0.63
SGD	0.62

Table 4.4: Accuracy using 10-fold cross validation for Feature Set 2 - Experiment 1 with 50,000 features

As we can see, since the accuracy did not improve significantly by augmenting the feature quantity to 50,000, we concluded it is better to just work with 5,000 to reduce vocabulary dimensionality.

---

<sup>3</sup>[http://scikit-learn.org/stable/modules/generated/sklearn.feature\\_extraction.text.CountVectorizer.html](http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html)

<sup>4</sup>[http://scikit-learn.org/stable/modules/generated/sklearn.feature\\_extraction.text.TfidfVectorizer.html](http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html)

<b>n-gram</b>	<b>Score</b>
sale	1.22044
jajaja	1.2245
dormir	1.22721
sueño	1.22992
sigo	1.23265
pensar	1.23538
siento	1.23538
cabeza	1.23812
digo	1.23949
vos	1.245

Table 4.5: 10 Most frequent TFIDF terms in dataset 2 - Experiment 1

If we take a look at the table of most frequent TFIDF terms, we identify some words we saw in the wordclouds that characterized a specific age group such as “dormir” in the case of users between 10 and 17 years old.

<b>n-gram type</b>	<b>Quantity</b>
unigram	4734
bigram	215
trigram	51

Table 4.6: Quantity of different n-gram types present in 5,000 Most frequent TFIDF terms in dataset

Another thing to mention by looking at the 5,000 most frequent n-grams, is that the 95% of them are unigrams (4,734 entries), so we could only consider the unigrams instead of bigrams and trigrams as well. In fact, the first non unigram appears in position 1040 in the list.

One of the advantages of the Random Forest Classifier is that we can display the features ranked by importance according to this predictor:



Feature	Score
dormir	0.00646472
lindo	0.00634449
web	0.00511489
mierda	0.00492847
jajajaja	0.00482256
re	0.00470042
gestión	0.00450522
vos	0.00425784
extraño	0.00425715
jajajajaja	0.00403919

Table 4.7: Importance of Features - Random Forest

Some of the features with most TFIDF appear in the top ten of important features for Random Forest as well (i.e., “dormir” and “vos”)

Finally, if we analyze the accuracy, we conclude that the best approach is to consider the 5,000 most frequent unigrams, and train them with SVM or SGD, to achieve an **accuracy of 62%**.

### **Feature set 3: User metadata and tweet stylometric features + tweet n-grams**

This experiment set combines the features from the first set with the features from the second set: user metadata, stylometric features and tweet n-grams.

Classifier	Accuracy
M. Naive Bayes	0.40
Random Forest	0.60
SVM	0.51
SGD	0.42

Table 4.8: Accuracy using 10-fold cross validation for Feature Set 3 - Experiment 1

If we compare the results against the ones obtained with the first dataset, the accuracy achieved is very similar. In fact, it stayed practically the same

for Naive Bayes, SVM and Random Forest. However, the SGD classifier’s accuracy went from 0.46% to 42%.

If we now compare it against the second set of features, the tweet n-grams outperformed the combined set of features.

Feature	Score
re	0.008106
mierda	0.006186
lindo	0.006114
vosotros	0.005993
dormir	0.005655
web	0.004889
blog	0.004676
orto	0.004555
linda	0.004358
jajajaja	0.004210

Table 4.9: Importance of features - Random Forest

As we can see in the table above, neither stylistic nor metadata features appear in the 10 most relevant features, which are all tweet unigrams from the NLP approach. The first non-NLP feature –“qtyEmojis”– appears in position 16.

Finally, if we compare the classifiers, we conclude that, Random Forest outperformed the rest of the classifiers when considering user metadata, tweet stylometric and NLP features, achieving an **accuracy of 60%**, which is actually lower than the previous approach.

#### Feature set 4: Subscription Lists n-grams

In this particular case, since not all the users are subscribed to lists, and some others might have those lists private, we will work with a smaller dataset containing 164 users only (14% of the original dataset):

AGE	USERS
10-17	11
18-24	35
25-34	38
35-49	57
50-xx	23

Table 4.10: Age distribution in dataset for subscribed lists

Classifier	Accuracy
M. Naive Bayes	0.24
Random Forest	0.38
SVM	0.39
SGD	0.29

Table 4.11: Accuracy using 10-fold cross validation for Feature Set 4 - Experiment 1 with 5,000 features

In this experiment we also tested to consider 50,000 n-grams, but reached the same conclusion as with tweets: augmenting the vocabulary did not improve accuracy, so it is preferable to just work with 5,000 most frequent ones.

Feature	Score
my	0.0076397
justinfollowplease	0.0068823
consumo	0.0068053
teamuru	0.0065644
alaguerrano	0.0065174
londres	0.0063319
socialmedia	0.0062614
uruguayos londres	0.0058962
endocrinologos	0.0058676
miramar	0.0055258

Table 4.12: Importance of Features - Random Forest

n-gram type	Percentage
unigram	28%
bigram	37%
trigram	35%

Table 4.13: Percentage of different n-gram types present in most relevant terms in dataset (Random Forest)

As we can see, in the case of Subscription lists, bigrams and trigrams take more relevance in comparison with the results obtained for the tweets.

As the results show, model training with n-grams from subscription lists achieved a low **accuracy of 39%**. However, it might be due to the fact that only the 14% of the users have public Subscription Lists in our dataset.

### Majority baseline classifier

We want to compare the performance of our model with a majority baseline. In the majority baseline, the majority class occurring in the data set is assumed to be the predicted class for all instances in the test data set. In our case, the majority class is 18-24, since 587 users belong to it. Hence, the **accuracy of this dummy predictor will be 51%**.

## Results

The objective of the present experiment was to assess the separate and joint predictive validity of metadata, stylistic and NLP approaches to age prediction.

We thought that the combination of approaches should increase the age prediction validity in Twitter data at a rate that is significantly higher than either approach alone.

However, we arrived to the conclusion that tweet unigrams perform better, without the need to include extra features.

As it turns out, the addition of more features does not always produce better results.

On the other hand, the n-grams obtained from Subscription lists, achieved the lower accuracy. But it might be affected by the small size of the dataset (only 14% of the users have public Subscription Lists in our dataset).

Finally, when comparing the accuracy against the majority baseline, we obtained better results in all cases except for subscription lists n-grams.

#### 4.2.2 Experiment 2: Prediction for Cyber Pedophilia Detection

Since our motivation is to develop a useful component in a pedophile detection system, instead of classifying the users in different age groups, we also want to focus on classifying adults versus adolescents. This way, the system should be able to detect adults posing as adolescents and flag their profiles for monitoring.

In Uruguay, the legal minimum age for sexual interactions is set at 15 and the legal age of majority at 18 <sup>5</sup>. Because the illegality of e.g., relationships between an 18 year old and a 15 year old is often very difficult to determine without a thorough police investigation, we also decided to create a class for an age group for which there could be no doubt about the illegal character of a sexual interaction with adolescents under 15. Following the idea of [2], we merged all age groups greater than 25 into one same age range 25-xx (25 and older).

Hence, for this matter we work with the following dataset:

AGE	USERS
10-17	237
18-24	587
25-xx	332

Table 4.14: Age distribution in dataset for forensic analysis

#### Feature set 1: User metadata and tweet stylometric features

Table 4.15 shows the results obtained.

---

<sup>5</sup><https://www.impo.com.uy/bases/codigo-penal/9155-1933/272>

Classifier	Accuracy
M. Naive Bayes	0.45
Random Forest	0.70
SVM	0.51
SGD	0.57

Table 4.15: Accuracy using 10-fold cross validation for Feature Set 1 - Experiment 2

Feature	Score
qtyUrls	0.175996
qtyEmojis	0.12768
qtyMentions	0.117611
followers_count	0.095925
qtyHashtags	0.0928547
tweets_count	0.0917871
qtyUppercase	0.0874878
favourites_count	0.08623
friends_count	0.0708661
profile_pic_gender	0.0224607
instagram	0.0146384
facebook	0.0104129
snapchat	0.00504708
linkedin	0.00100411

Table 4.16: Importance of Features - Random Forest

If we compare the ranking of features against the ones obtained in the Experiment 1 with the same feature types (See subsection 4.2.1), we realize that in both cases qtyUrls, qtyEmojis, qtyMentions and followers\_count are positioned in the top five.

When assessing the classifiers performance, we reach to the same result as in the Experiment 1 for metadata and stylometric features: Random Forest outperformed the rest of the classifiers, achieving an **accuracy of 70%**. The accuracy increased 17% compared with the previous experiment.

If we compare the confusion matrix of Experiment 1 and 2 for this set of features, we see that when switching from five classes to three, the dataset becomes more balanced.

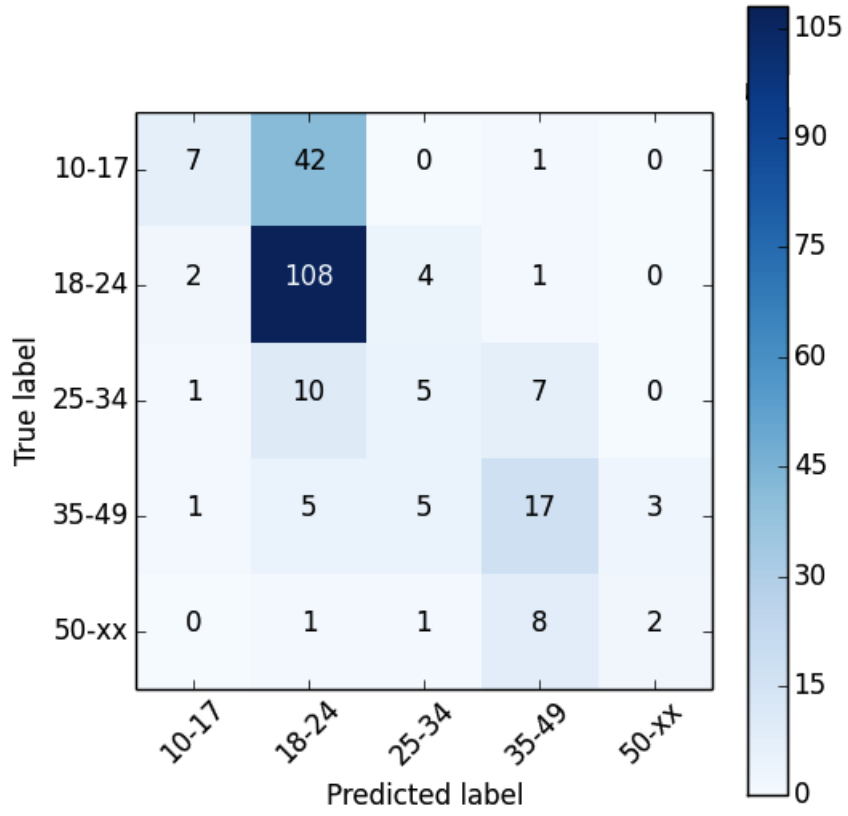


Figure 4.1: Confusion Matrix not normalized for metadata and stylometric features in Experiment 1

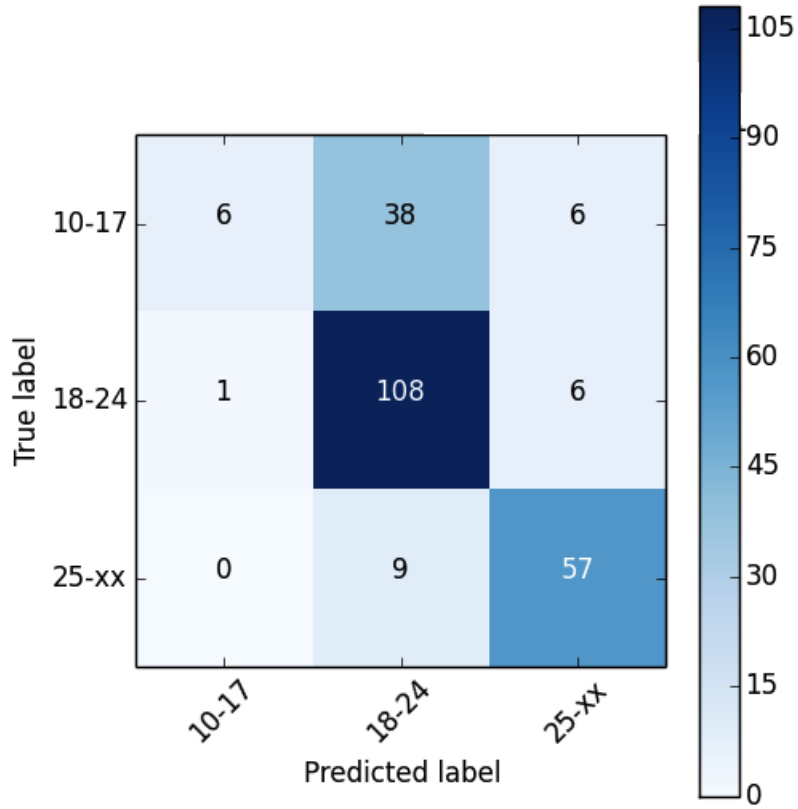


Figure 4.2: Confusion Matrix not normalized for metadata and stylometric features Experiment 2

For this experiment in particular, the confusion matrix is very useful to study sexual predators. We analyze the one corresponding to Random Forests since it is the classifier that worked better for this task.

If we focus on the upper right corner of Figure 4.2, we have 6 users that indicated in their profile description that they were in between 10 and 17 years old. However, after analyzing the metadata and stylistic features from the tweets, the classifier believes that those users write like a much older user belonging to 25-xx class. This might indicate these users are pretending to be someone younger, which was precisely what we wanted to detect in this Experiment.



## Feature set 2: Tweet n-grams

For this experiment we extracted the 5,000 most meaningful unigrams, bigrams and trigrams. Table 4.17 shows the performance results.

Classifier	Accuracy
M. Naive Bayes	0.70
Random Forest	0.71
SVM	0.73
SGD	0.74

Table 4.17: Accuracy using 10-fold cross validation for Feature Set 2 - Experiment 2 with 5,000 features

Feature	Score
re	0.0120827
extraño	0.0116176
blog	0.00759291
sos	0.0071604
vos	0.00685758
empleo	0.00657081
puta	0.00644849
lindo	0.00594509
dale	0.00573223
amo	0.00558721

Table 4.18: Importance of Features - Random Forest

If we compare this table with the one in the first experiment (see Table 4.7) for the same feature type, most of the words appear in both lists (i.e., “lindo”, “re”, “vos”, “extraño”), but in different order.

On the other hand, if we perform a TFIDF analysis, the ten most frequent TFIDF terms are the same as in the first experiment for the Set 2 of features (see Table 4.5). This makes sense since we are using the same tweets (and same vocabulary) for both experiments.

Finally, if we analyze the accuracy, we conclude that the best approach is to consider the 5,000 most frequent unigrams, and train them with SGD

to achieve an **accuracy of 74%** (Same features and same classifier from Experiment 1).

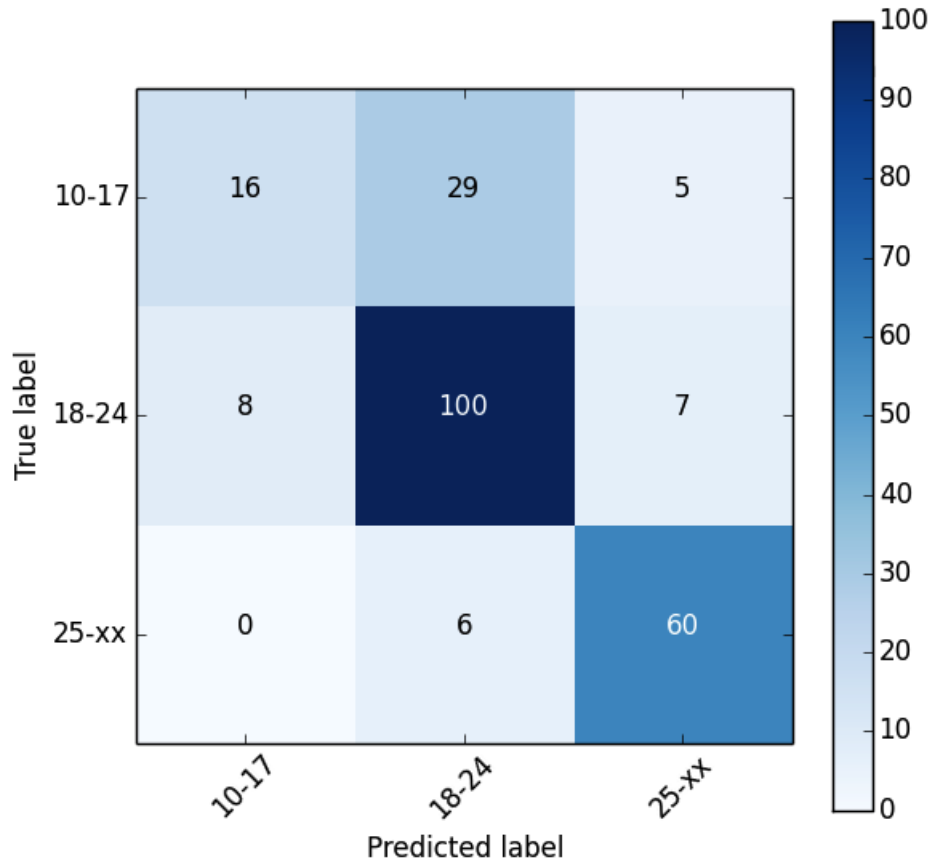


Figure 4.3: Confusion Matrix not normalized for tweet n-grams

In this case, if we pay attention to the upper right corner, there are 5 users that indicated in their profile description that they belong to 10-17 age group, but the classifier predicted a much older class. This might indicate these users are posing as younger users.

**Feature set 3: User metadata and tweet stylometric features +  
Tweet n-grams**

Classifier	Accuracy
M. Naive Bayes	0.45
Random Forest	0.71
SVM	0.51
SGD	0.53

Table 4.19: Accuracy using 10-fold cross validation for Feature Set 3 - Experiment 2 with 5,000 features

Similarly to what happened in Experiment 1, tweet n-grams accuracy exceeded the accuracy of the combined feature types. In fact, the accuracy achieved is almost the same as the accuracy obtained from metadata and stylistic features only.

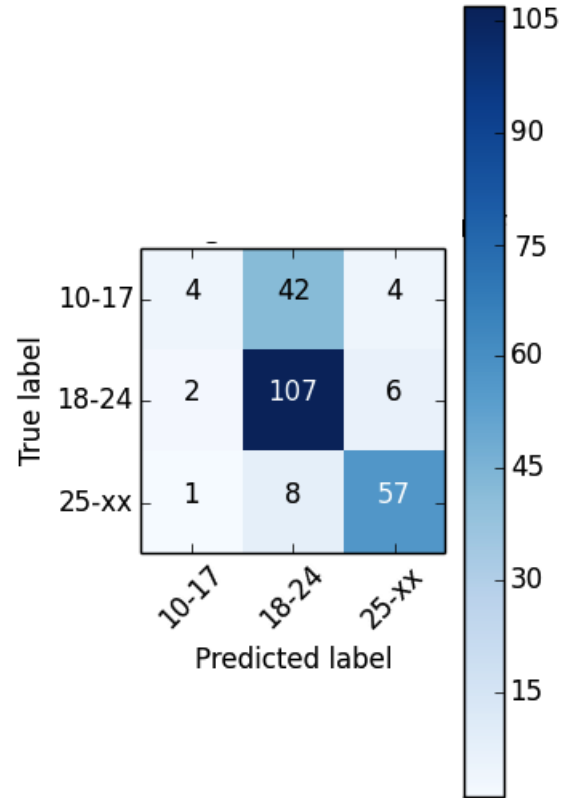


Figure 4.4: Confusion Matrix not normalized for tweet n-grams and custom fields

The upper right corner of the confusion matrix follows the line of the rest of the dataset, since it identified 4 users that they are supposed to be young, but the classifier predicts they are much older.

#### Feature set 4: Subscription Lists n-grams

In the case of subscription lists analysis, as we mentioned before, the dataset is only 14% of the original, and for this experiment it has this distribution:

AGE	USERS
10-17	11
18-24	35
25-xx	118

Table 4.20: Age distribution in dataset for forensic analysis in subscribed lists

Classifier	Accuracy
M. Naive Bayes	0.29
Random Forest	0.74
SVM	0.75
SGD	0.73

Table 4.21: Accuracy using 10-fold cross validation for Feature Set 4 - Experiment 2 with 5,000 features

An interesting conclusion that can be deducted from the previous table is that by reducing the age groups from 5 to 3, the accuracy almost doubled (using same feature types and same classifiers).

This means the analysis of subscription lists can become a meaningful feature to predict age, if the lists are better balanced across the different age ranges.

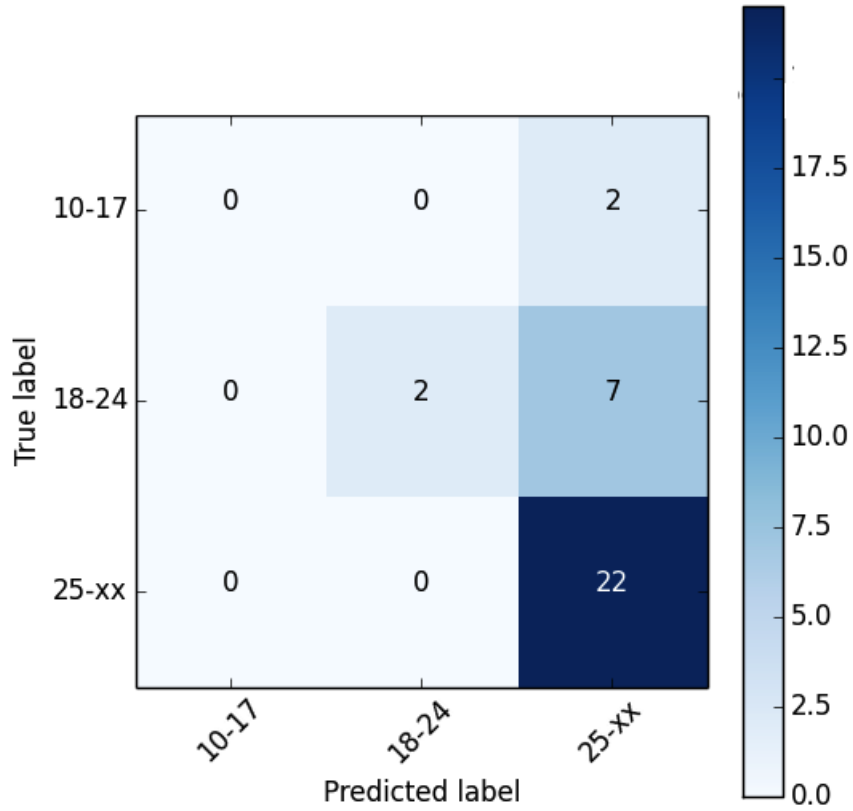


Figure 4.5: Confusion Matrix not normalized for subscription n-grams

In this case, the SVM classifier detected 2 users that indicated to be 10-17 years old, but later was predicted to belong to a much older class.

### Majority baseline classifier

In this experiment the majority baseline predictor shares the same accuracy of 51% with the previous experiment.

Since we only merged the age ranges 25-34, 35-49, 50-64, and 65-xx into one new age class 25-xx, but left unchanged the majority class, 18-24, the true positives and the total stay the same, hence, the accuracy as well.

## Results

Same as in Experiment 1, the combined approach did not outperform the n-gram analysis of tweets. Hence, it is not worth it the overhead dedicated to merge the features after extracting NLP ones.

Another key point to mention is the significant increase in the accuracy of subscription lists n-grams, which went from 39% (even lower than the majority baseline) to 75%, only by grouping the 3 older age groups into one.

### 4.2.3 Experiment 3: Prediction with a Balanced Dataset

Some authors like Tam et.al [9] mentioned the possibility that some models might not do as well because of an unbalanced dataset. Therefore, we will run experiments to compare the performance of the models using unbalanced and balanced datasets. In order to do that, we checked the age range with fewer users, and only exported that quantity of users for the remaining age groups.

As we can see in the previous tables, the age class 50-xx is the group with less entries (59), and for subscribed lists is 10-17 (11). In conclusion, the first dataset will contain 59 users per class, and the subscribed list dataset will include 11 instances per age group.

#### Feature set 1: User metadata and tweet stylometric features

Classifier	Accuracy
M. Naive Bayes	0.30
Random Forest	0.49
SVM	0.19
SGD	0.27

Table 4.22: Accuracy using 10-fold cross validation for Feature Set 1 - Experiment 3

In the case of user metadata and tweet stylometric features, balancing

the data did not yield better predictions, in fact, the accuracy dropped significantly for most classifiers.

### **Feature set 2: Tweet n-grams**

For this experiment we extracted the 5,000 most meaningful unigrams, bigrams and trigrams.

<b>Classifier</b>	<b>Accuracy</b>
M. Naive Bayes	0.43
Random Forest	0.47
SVM	0.46
SGD	0.45

Table 4.23: Accuracy using 10-fold cross validation for Feature Set 2 - Experiment 3 with 5,000 features

When comparing these results with the ones obtained in the first experiment, we reach to the same conclusion as before: balancing the data did not yield better predictions for this type of feature either, since the accuracy dropped for all the classifiers.

### **Feature set 3: User metadata and tweet stylometric features + Tweet n-grams**

<b>Classifier</b>	<b>Accuracy</b>
M. Naive Bayes	0.30
Random Forest	0.45
SVM	0.19
SGD	0.30

Table 4.24: Accuracy using 10-fold cross validation for Feature Set 3 - Experiment 3 with 5,000 features

The results follow the same line as the rest of the data sets for this experiments. Neither in this case, dataset balancing helped improving the predictions.



#### Feature set 4: Subscription Lists n-grams

In this particular case, since not all the users are subscribed to lists, and we want to have a balanced dataset, we ended up working with only 55 users (4.75% of the original dataset).

Classifier	Accuracy
M. Naive Bayes	0.22
Random Forest	0.26
SVM	0.22
SGD	0.24

Table 4.25: Accuracy using 10-fold cross validation for Feature Set 4 - Experiment 3 with 5,000 features

This approach produced by far the worst accuracy. Clearly, the initial dataset reduction to consider only the users with subscription lists, and the subsequent reduction to balance it provoked a major loss of information, which generated poor results. For instance, in the case of SVM, the accuracy dropped a 44%, from 0.39 (which was already pretty low) to 0.22.

#### Results

For this experiment, dataset balancing failed unanimously for all the feature sets when trying to increase the accuracy of the classifiers predictions.

#### 4.2.4 Experiment 4: Our Predictions vs. Microsoft Face API

Microsoft Face API was able to identify the age of 741 users from their profile picture, which is a 64% of the original dataset.

Having said that, we want to test its accuracy and compare it against our model.

On one hand, the Microsoft platform achieved an accuracy of 45% while predicting the exact age.

On the other hand, we compared against the model that worked better in Experiment 1, which was achieved by using SGD/SVM classifier, and training it with tweet n-grams (we used SGD in this case). With this configuration, we train the dataset with users with valid predicted age and gender through Face API, and obtained an accuracy of 64%, which means we performed more accurate predictions.

Nevertheless, Microsoft Face API uses a regression approach, while we do classification prediction in our experiments. Therefore, if the API predicted 24 and the actual age was 25, from the regression point of view is a good estimation. However, in the classification approach, the age 24 belongs to the class 18-24 and 25 to 25-34, so in this case will be considered a false positive.

Age group	True Qty	FaceAPIQty
10-17	132	31
18-24	374	258
25-34	88	272
35-49	112	143
50-xx	35	37

Table 4.26: Actual age distribution vs. Face API age distribution

In order to objectively compare both performances, we should take a regression approach as well, and measure the Mean Absolute Error (MAE) of each predictor.

## 5 Conclusions

In summary, we find that the performance of our best model to predict the age from 5 different age ranges obtained by training the SVM or SGD models with tweet n-grams outperformed the results achieved by Arroju et al [10], which is one of the very few that studied age prediction in Twitter for Spanish speakers. In that investigation, they achieved an accuracy of 48% which is even less than the majority baseline. It is worth to mention that that result was obtained on the PAN Clef Dataset, which is 5 times smaller than our generated dataset, and only considered tweet text.

In our case, the best result obtained for the 5-class problem was 62%, which is a significant increase. We also outperformed the majority baseline, as well as the predictions made by Microsoft Face API.

Furthermore, if we group the users of the 3 oldest ranges into one age range, the accuracy of the prediction improved to 74%, as the Experiment 2 shows.

On the other hand, we find that examining tweet stylistic features and Twitter metadata combined with NLP features did not improve our predictions made with tweet n-grams only. Hence, they only added overhead and in some cases even degraded our predictions.

In general, it is challenging to compare model performance across studies because of differences in age groups examined, sampling, and annotation methods used.

At the same time, the present study makes a significant contribution to the state-of-the-art since we compared the performance of metadata, stylistic and Natural Language Processing features independently and later combined for the task of age prediction in Twitter. We also identified and generated novel features that model the presence of other social network profiles (i.e.,

Facebook, Instagram Snapchat) linked to the Twitter profile of a user. Likewise, we include the analysis of subscription lists, as a way to understand the different interests among the age groups.

Another contribution of this investigation is, in fact, the software Tweet-Lab, a custom pipeline streamer that allows the user to collect new Twitter users, and annotate them with their age, in case it is mentioned in the profile description. It also generates and persists all user metadata and stylistic features mentioned in this research. In addition, no other investigation has studied the subscription lists in Twitter as a way to understand the user's interests.

This paves the way for a deeper research along this line of work, such as, by adding track words like 'birthday', 'birthdate' as a means to annotate more users.

Nonetheless, individuals who specify their age in the description may constitute a specific subpopulation that reflects a selection bias compared with individuals who do not. However, all classification studies using social media data are to some extent biased since there is no comprehensive frame of all users to sample from [22].

Another difficulty is that studies of this nature may need continual updating. To illustrate, nowadays acronyms like "LOL" are being used more by adults than kids, and kids are replacing the use of abbreviations for emojis or Gifs. Also, five years ago, Facebook audience was a young public, and now millennials are switching to other networks like Instagram or Snapchat. There communication happens through visuals like emojis that can make up for the useful cues that are often missing from digital chit-chatting: the raise of an eyebrow, the shrug of the shoulders, the rolling of an eye <sup>1</sup>.

As per the experiments, we used a snapshot of the database and gathered a corpus of 1,156 users, in which the older age groups are smaller (aged 25 or older) compared to the rest, which likely explains the poorer performance in predicting this age group. One possible explanation for the smaller sample may be that older adults are less likely to be on Twitter. In 2016, 36% of adults aged 18 to 29 used Twitter, compared with 22% of adults aged 30 to 49<sup>2</sup>.

---

<sup>1</sup><http://time.com/4834112/millennials-gifs-emojis>

<sup>2</sup><http://www.pewinternet.org/fact-sheet/social-media/>

Additionally, older adults might be less likely to announce their age on their profile description. Future iterations should be able to produce larger samples of older ages to improve the classification accuracy for older Twitter users. To that end, we can combine TweetLab with other approaches like the idea of integrating external survey labeling games like the one proposed by [20]. Some users might include an age reference in their profile description that differs from their age, for instance, somebody that has X years of professional experience. Hence, we can also add a task to identify those wrong labeled users to avoid noise in the dataset.

Future work might be also done to provide some insight on the use of slang (non dictionary words) in Spanish language, by adding a Spanish lexicon to TweetLab. This might be useful, since it is one of the most meaningful features according to [3],[4],[5],[6],[7] when trying to identify a young user from an older one. We can also analyze the multilingual aspect of users, to find out if the fact of writing in another language as well helps with age profiling.

## 6 Bibliographical References

- [1] J. Caverlee and S. Webb, “A large-scale study of myspace: Observations and implications for online social networks,” in *Proceedings from the 2nd International Conference on Weblogs and Social Media (AAAI)*, 2008. [Online]. Available: <http://faculty.cs.tamu.edu/caverlee/pubs/caverlee08alarge.pdf>
- [2] C. Peersman, W. Daelemans, and L. Van Vaerenbergh, “Predicting age and gender in online social networks,” in *Proceedings of the 3rd International Workshop on Search and Mining User-generated Contents*, ser. SMUC ’11. New York, NY, USA: ACM, 2011, pp. 37–44. [Online]. Available: <http://doi.acm.org/10.1145/2065023.2065035>
- [3] S. Rosenthal and K. McKeown, “Age prediction in blogs: A study of style, content, and online behavior in pre- and post-social media generations,” in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, ser. HLT ’11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 763–772. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2002472.2002569>
- [4] D. Nguyen, R. Gravel, D. Trieschnigg, and T. Meder, “”how old do you think I am?” A study of language and age in twitter,” in *Proceedings of the Seventh International Conference on Weblogs and Social Media, ICWSM 2013, Cambridge, Massachusetts, USA, July 8-11, 2013.*, 2013. [Online]. Available: <http://www.aaai.org/ocs/index.php/ICWSM/ICWSM13/paper/view/5984>
- [5] D. Rao and D. Yarowsky, “Detecting latent user properties in social media,” in *In Proc. of the NIPS MLSN Workshop*, 2010.
- [6] S. Goswami, S. Sarkar, and M. Rustagi, “Stylometric analysis of bloggers’ age and gender.” in *ICWSM*, E. Adar, M. Hurst, T. Finin,

- N. S. Glance, N. Nicolov, and B. L. Tseng, Eds. The AAAI Press, 2009. [Online]. Available: <http://dblp.uni-trier.de/db/conf/icwsm/icwsm2009.html#GoswamiSR09>
- [7] F. Barbieri, “Patterns of age-based linguistic variation in american english1,” *Journal of Sociolinguistics*, vol. 12, no. 1, pp. 58–88, 2008. [Online]. Available: <http://dx.doi.org/10.1111/j.1467-9841.2008.00353.x>
- [8] PAN. Pan. [Online]. Available: <http://pan.webis.de/#>
- [9] J. Tam and C. H. Martell, “Age detection in chat,” in *2009 IEEE International Conference on Semantic Computing(ICSC)*, vol. 00, 09 2009, pp. 33–39. [Online]. Available: [doi.ieeecomputersociety.org/10.1109/ICSC.2009.37](http://doi.ieeecomputersociety.org/10.1109/ICSC.2009.37)
- [10] M. Arroju, A. Hassan, and G. Farnadi, “Age, gender and personality recognition using tweets in a multilingual setting: Notebook for PAN at CLEF 2015,” in *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015.*, 2015. [Online]. Available: <http://ceur-ws.org/Vol-1391/57-CR.pdf>
- [11] G. Hirst and O. Feiguina, “Bigrams of syntactic labels for authorship discrimination of short texts,” *LLC*, vol. 22, no. 4, pp. 405–417, 2007. [Online]. Available: <https://doi.org/10.1093/llc/fqm023>
- [12] A. Glover, “Automatically detecting stylistic inconsistencies in computer-supported collaborative writing [microform],” ser. Canadian theses. Thesis (M.A.)—University of Toronto, 1996. [Online]. Available: [https://books.google.com.uy/books?id=n\\_ngjgEACAAJ](https://books.google.com.uy/books?id=n_ngjgEACAAJ)
- [13] J. Burrows, “‘delta’: a measure of stylistic difference and a guide to likely authorship,” *LLC*, vol. 17, no. 3, pp. 267–287, 2002. [Online]. Available: <https://doi.org/10.1093/llc/17.3.267>
- [14] N. Graham, G. Hirst, and B. Marthi, “Segmenting documents by stylistic character,” *Nat. Lang. Eng.*, vol. 11, no. 4, pp. 397–415, Dec. 2005. [Online]. Available: <http://dx.doi.org/10.1017/S1351324905003694>
- [15] N. Pendar, “Toward spotting the pedophile telling victim from predator in text chats,” in *Proceedings of the First IEEE International Conference on Semantic Computing (ICSC 2007), September 17-19,*

- 2007, Irvine, California, USA, 2007, pp. 235–241. [Online]. Available: <https://doi.org/10.1109/ICSC.2007.32>
- [16] J. Marquardt, G. Farnadi, G. Vasudevan, M.-F. Moens, S. Davalos, A. Teredesai, and M. De Cock, “Age and gender identification in social media,” in *CLEF 2014 working notes*, 2014.
- [17] M. Coltheart, “The mrc psycholinguistic database,” *Quarterly Journal of Experimental Psychology*, vol. 33, pp. 497–505, 1981.
- [18] J. Schler, M. Koppel, S. Argamon, and J. W. Pennebaker, “Effects of age and gender on blogging,” in *Computational Approaches to Analyzing Weblogs, Papers from the 2006 AAAI Spring Symposium, Technical Report SS-06-03, Stanford, California, USA, March 27-29, 2006*, 2006, pp. 199–205. [Online]. Available: <http://www.aaai.org/Library/Symposia/Spring/2006/ss06-03-039.php>
- [19] B. Perozzi and S. Skiena, “Exact age prediction in social networks,” in *Proceedings of the 24th International Conference on World Wide Web*, ser. WWW ’15 Companion. New York, NY, USA: ACM, 2015, pp. 91–92. [Online]. Available: <http://doi.acm.org/10.1145/2740908.2742765>
- [20] D. Nguyen, D. Trieschnigg, A. S. Dogruöz, R. Gravel, M. Theune, T. Meder, and F. de Jong, “Why gender and age prediction from tweets is hard: Lessons from a crowdsourcing experiment,” in *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, August 23-29, 2014, Dublin, Ireland*, 2014, pp. 1950–1961. [Online]. Available: <http://aclweb.org/anthology/C/C14/C14-1184.pdf>
- [21] F. A. Zamal, W. Liu, and D. Ruths, “Homophily and latent attribute inference: Inferring latent attributes of twitter users from neighbors,” in *Proceedings of the Sixth International Conference on Weblogs and Social Media, Dublin, Ireland, June 4-7, 2012*, 2012. [Online]. Available: <http://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/view/4713>
- [22] C. R. R. P. Morgan-Lopez AA, Kim AE, “Predicting age groups of twitter users based on language and metadata features,” *PLoS ONE*, 2017. [Online]. Available: <https://doi.org/10.1371/journal.pone.0183537>
- [23] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta, “Classifying latent user attributes in twitter,” in *In Proc. of SMUC*, 2010.



- [24] S. G. and B. C., “Weighting approaches in automatic text retrieval,” *Information Processing and Management*, vol. 24, pp. 513—523, 1988.
- [25] A. McCallum and K. Nigam, “A comparison of event models for naive bayes text classification,” in *AAAI-98 workshop on learning for text categorization*. AAAI Press, 1998, pp. 41–48.
- [26] T. Hastie, R. Tibshirani, and J. Friedman, “The elements of statistical learning,” ser. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.

## 7 Appendices

### Appendix A – Mapping between Twitter and Facebook account

Twitter allows the user to link a Facebook account to his Twitter profile, providing the option of automatically posting the tweets in Facebook as well.

Even if the age is not an attribute in the Twitter profile, it does appear on the Facebook profile. This is why it is interesting to try to access the linked Facebook profile of an account.

The age field might be public or private, if the user selected to keep it private, only the users friends will be able to see this information. Nevertheless, some users have that information available for everyone, which seemed a good way to build our corpus.

#### First Strategy

The first thing to do was extracting the linked Facebook account from the Twitter profile through the API. However, this information is not available in the response of the request `users/lookup` used to gather the user information.

Hence, we decided to work only with the profiles that explicitly mentioned the Facebook account in the description inside the Twitter profile:



This is a way to retrieve the linked Facebook account, now we only need a tool to get the information inside the Facebook profile.

In fact, Facebook provides a tool named “Facebook Graph API”<sup>1</sup> that allows to send requests and obtain user information, pages, places, etc, similar to what Twitter API does (see section 2.4.2).



Another key thing to mention is that the requests get executed from a specific user profile, and the results may vary depending on the information of that user in particular (location, friends, interests, among others). This is a big difficulty since the same query might return different results based on the user that does the request.

Nevertheless, as we can see in the image above, we can easily get the user birthday, which comes handy. But, at this point we encounter another

<sup>1</sup><https://developers.facebook.com/docs/graph-api/>

difficulty: since version 2.0 it is not possible anymore to look for a user by his username.

The reason behind this is that some bots were using the username to send mails through Facebook. For instance, if we have the URL: `http://www.facebook.com/sebastian.trug`, his facebook email will be `sebastian.trug@facebook.com`. Thus, if the bot sends an email to that email, the user will receive a new message in the “Messages” section.

Same will happen if we want to access from Python, when using the GraphAPI as well as the standard library “requests”.

However, you still can search for a user by his id, but most of the users have replaced this UID by a more user friendly text. To provide an example, if we access the URL: `https://www.facebook.com/449423951902909`, we will be automatically redirected to `https://www.facebook.com/anabrecontreras`, since “anabrecontreras” is the username corresponding to ID: 449423951902909.

Having said that, most users post their Facebook username instead of the UID. Therefore, it would be necessary to obtain the UID from the username. In order to do that, we investigated the command “search” of the Graph API. For instance, `graph.search(term='veronica tortorella',type='user')` simulates when the user searches by name another user in the Search Bar inside Facebook. Unfortunately, this command returns all the UIDs of users with similar name to the one typed, so this did not work either.

In case of still wanting to use this approach, we should traverse one by one the list of UIDs returned (might be  $n$ ), and check if the redirect URL matches their Facebook profile. Another difficulty lies on the fact that we cannot specify the parameters to return, so we should use the UID and request for the field=birthday, which might be defined or not, with private or public visibility).

This processing might provoke a significant overhead, in the case we look for common names like “Juan Perez”, this query might return millions of results, and traverse them one by one is expensive. Moreover, even after this processing, it might not even have birth date available.

## Second Strategy

According to the documentation, the only way to get information from Facebook is through Facebook Graph API.

After identifying the major drawbacks mentioned above, we started considering a whole new approach: Web Scraping in Facebook

Web scraping is a technique used by software programs to extract information from different web sites. Usually this programs simulate the human navigation in the World Wide Web by using manually HTTP protocol, or even embed a browser in an application.

The idea is to simulate the navigation towards a user profile, in order to get the “Birthday” field, in case this information is present, using the Facebook URL explicitly mentioned in the Twitter profile as a starting point.

In order to do this, we tested different libraries such as Selenium, PhantomJS, cookielib, mechanize, pycurl, as well as libraries to parse and handle html like HTMLParser, urllib2 and BeautifulSoup.

In our particular case, we want to retrieve the information inside the URL `<profile>/about?section=contact-info&pnref=about`, which contains a section with basic information of the user. This is the place where we should find the birth date.



The final solution was developed with Selenium and PhantomJS. We found an interesting fact: Facebook behaves differently if the profile URL contains ID or user name. If the profile URL contains the user ID, it is not necessary to be logged in Facebook to access the information. Nevertheless, if the URL is formed with the user name, an error will be displayed in case it is not connected to Facebook.

Hence, the first task consisted of simulating the user login with a test account. Once logged in through Selenium, we make a second observation: the page source gathered from the URL and from Selenium differ. As a direct consequence, it is necessary to base on the content obtained through Selenium since it is our test context.

We also used the commands `WebDriverWait` and `until` to block the execution until all the information is loaded in the webpage. This happens frequently in pages with javascript and with ajax calls, where sometimes we receive an empty response in Selenium due to the fact that the query was executed prior to the completion of the page load.

Another difficulty encountered is the lack of IDs inside the HTML. Instead, many css classes appear, which makes it really hard to track and get the elements of interest. In our case, we need to look for the id “pagelet\_basic”, which represents the Basic Information section, and that contains the Birth date.

This information might be inside a div with the text “Fecha de nacimiento”, or we can have “Año de nacimiento” separated from “Fecha de nacimiento”.

In our reality, we just need the year to calculate the user age, therefore, we will first look for the birth year, and if this is not present, we will search the birth date. We should also mention that, if this information is not public, we will not find anything here.

## Appendix B - Custom Stopwords List

<i>STOPWORDS</i>				
10	dar	hecho	parte	todas
11	debe	historia	pasa	trabajar
12	decir	hola	pasado	trabajo
15	deja	hora	pasar	twitter
20	dejar	horas	paso	va
30	dentro	hoy	país	vamos
acuerdo	después	igual	peor	van
ahora	dice	importante	persona	vas
ahí	dicen	ir	personas	veces
alguien	dijo	lado	poder	veo
amigo	domingo	llega	pone	ver
amigos	dos	llegar	poner	verdad
amor	día	lugar	primer	vez
así	días	lunes	primera	vida
aunque	equipo	madre	programa	viene
ayer	esperando	mal	puede	viernes
año	espero	mano	pueden	volver
años	falta	mas	puedo	voy
bien	favor	mañana	queda	vía
buen	feliz	medio	quiere	
buen	fin	mejor	quiero	
buenas	final	mejores	realidad	
bueno	forma	menos	sabe	
buenos	foto	mientras	saber	
cada	fotos	mil	sabes	
cambiar	futuro	minutos	salir	
cambio	ganas	mira	seguir	
cara	gente	misma	seguro	
casa	gracias	mismo	semana	
casi	gran	momento	ser	
claro	grande	mucha	si	
cosa	grandes	muchas	siempre	
cosas	grupo	mundo	sigue	
creo	gusta	nadie	solo	
cualquier	haber	noche	suerte	
cuenta	hablar	nueva	tan	
cumple	hace	nuevo	tarde	
cómo	hacen	nunca	tener	
da	hacer	palabras	tiempo	
dan	haciendo	parece	toda	