A Sentiment Analysis Model of Spanish Tweets

Case Study: Colombia 2014 Presidential Election

Jhon Adrián Cerón-Guzmán

Universidad Nacional de Colombia
Facultad de Ingeniería
Departamento de Ingeniería de Sistemas e Industrial
Bogotá D.C., Colombia
2016

A Sentiment Analysis Model of Spanish Tweets

Case Study: Colombia 2014 Presidential Election

Jhon Adrián Cerón-Guzmán

Thesis presented in partial fulfillment of the requirements for the degree of
Master in Computer and Systems Engineering

Advisor
Elizabeth León-Guzmán, Ph.D.

Universidad Nacional de Colombia
Facultad de Ingeniería
Departamento de Ingeniería de Sistemas e Industrial
Bogotá D.C., Colombia
2016

Title in English

A Sentiment Analysis Model of Spanish Tweets. Case Study: Colombia 2014 Presidential Election

Título en español

Modelo de análisis de sentimientos de mensajes de Twitter en español. Caso de estudio: Elecciones Presidenciales en Colombia de 2014

Abstract: What people say on social media has turned into a rich source of information to understand social behavior. Sentiment analysis of Twitter data has been widely used to capture trends in public opinion regarding important events such as political elections. However, current research in social media analysis in political domains faces two major problems, namely: sentiment analysis methods implemented are often too simple, and most of the researches have assumed that all users and their tweets are trustworthy. This thesis is aimed at dealing with these problems to achieve more reliable public opinion measurements. Colombia 2014 presidential election was proposed as case study. First, a research on social spammer detection on Twitter was carried out by following machine learning approaches to distinguish spammer accounts from non-spammer ones. Because of the brevity of tweets and the widespread use of mobile devices, Twitter is also a rich source of noisy data containing many non-standard word forms. Since this is a task that exploits the large amount of user-generated texts, the performance of sentiment analysis may drop significantly if several lexical variation phenomena are not dealt with. For that reason, a lexical normalization system of Spanish tweets was developed to improve the quality of natural language analysis, using finite-state transducers and statistical language modeling. Lastly, a sentiment analysis system of Spanish tweets was developed by implementing a supervised classification approach. The system was applied in the Colombian election to infer voting intention. Experimental results highlight the importance of denoising in Twitter data to achieve more reliable public opinion measurements. Together with this, results show the potential of social media analysis to infer vote share, obtaining the lowest mean absolute error and correctly ranking the highest-polling candidates in the first round election. However, such an important method cannot be put forward as a substitute of the traditional polling.

Resumen: Lo que las personas dicen en plataformas de *social media* se ha convertido en una fuente valiosa de información para entender el comportamiento social. Análisis de sentimientos de datos de Twitter se ha utilizado ámpliamente para capturar tendencias en la opinión pública con respecto a temas importantes como los son las elecciones políticas. Sin embargo, la investigación actual sobre aplicaciones de análisis de *social media* en contextos políticos enfrenta dos grandes problemas, a saber: se han empleado los métodos más simples de análisis de sentimientos, y se ha asumido que todos los usuarios y sus *tweets* son dignos de confianza. Esta tesis tiene como objetivo hacer frente a estos problemas con el fin de alcanzar mediciones más fiables de la opinión pública. Las elecciones presidenciales en Colombia de 2014 se propusieron como caso de estudio. En primer lugar, se llevó a cabo una investigación sobre la detección de *spammers* en Twitter, implementando enfoques de aprendizaje automático para distinguir cuentas *spammers* de las que no lo son. Debido a la brevedad de los *tweets* y al ámplio uso de dispositivos móviles, Twitter se ha convertido en una fuente de datos ruídosos que

contiene muchas formas de palabra que no son estándar. Al tratarse de una tarea que explota la gran cantidad de texto generado por los usuarios, el desempeño de análisis de sentimientos podría degradarse si no se abordan varios fenómenos de variación léxica presentes en los *tweets*. Por esta razón, se desarrolló un sistema de normalización léxica de *tweets* en español, el cual emplea transductores de estado finito y modelado de lenguaje estadístico, a fin de mejorar la calidad del análisis del lenguaje natural. Por último, se desarrolló un sistema de análisis de sentimientos de *tweets* en español siguiendo un enfoque de clasificación supervisada, el cual se aplicó en el contexto de las citadas elecciones para realizar inferencia de intención de voto. Los resultados experimentales resaltan la importancia de eliminar el ruído de los datos de Twitter que se utilizan para realizar mediciones de la opinión pública. Junto con esto, los resultados muestran el potencial del análisis de *social media* para inferir la distribución de los votos, obteniendo la media del error absoluto más baja y correctamente clasificando los candidatos de mayor votación en la primera vuelta electoral. Sin embargo, dicho método no puede plantearse como un sustituto del sondeo electoral tradicional.

Keywords: social media, Twitter, Spanish tweets, spammer detection, lexical normalization, sentiment analysis, voting intention inference, politics, presidential election, Colombia

Palabras clave: social media, Twitter, tweets en español, detección de spammers, normalización léxica, análisis de sentimientos, inferencia de intención de votación, política, elecciones presidenciales, Colombia

Dedication

To the only wise God be glory forever through Jesus Christ! (Romans 16:27)

To my mom, Yasmín.

Acknowledgments

This thesis work would not be possible without the invaluable support from several people. First of all, I want to thank Rodríguez-Roncancio Family for their kindness and hospitality. I also thank my thesis advisor Elizabeth León Guzmán, who trusted and supported me during the course of the master's degree. Camilo López, Róbinson Alvarado, and Arles Rodríguez contributed with their reviews to improve the quality of this work; I thank them greatly.

Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

In an increasingly connected world taking advantage of what people say about factual or subjective issues might bring gains not only in the economic and political arena, but also in the social one. However, finding and monitoring such information is a formidable task due to the large amount of user-generated content that is spread on the web [47]. And, not least, language diversity in the web [39] becomes a major issue to be considered.

Social media platforms such as Facebook[1] and Twitter[2] have led to deep changes in the paradigm of information generation and consumption. For example, real-time Twitter content on natural disasters has been exploited to support disaster management teams [46]. Twitter is nowadays a popular microblogging site where users receive and exchange information instantaneously with a global audience; this is, users are not limited to their friendship networks, as it happens in Facebook. 'Tweeting', therefore, has become an activity *par excellence* to say what one thinks or feels, because of the brevity of tweets[3] and the widespread use of mobile devices [75].

What people say on social media about issues of their everyday life, the society, and the world in general has turned into a rich source of information to understand social behavior [69]. This large amount of user-generated content has brought new research opportunities to explore the human subjectivity at large scale, which was not feasible using traditional methods. However, analyzing this content also presents several challenges, including: distinguishing noisy, useless, and irrelevant information from valuable data; and developing text analysis approaches based on Natural Language Processing (NLP) techniques, which properly adapt to the informal genre and the free writing style of these platforms. Addressing these challenges would lead to more reliable results, because new forms of spam have been spread to manipulate social media discourse [26] and the performance of traditional NLP tools degrades on social media data [32, 16].

An appealing application of social media analysis is to determine the opinion orientation expressed in text streams. Sentiment analysis or opinion mining, as this application is known, deals with the task of rating the opinion orientation as either positive, negative, or neutral [47, 31]. This computational approach has been implemented in a diversity of domains ranging from marketing to politics [69]. The latter has caught the attention

[1] `https://www.facebook.com/`

[2] `https://twitter.com/`

[3] User posts on Twitter are known as tweets, which have a 140-character limit.

of researchers, whom have investigated the feasibility of supplementing or substituting traditional electoral polling with sentiment classification of text streams [55, 80]. Despite the relative success reported in the literature, the following problems have been identified [50, 28]: most of the works implemented the simplest of sentiment analysis methods, whose performance is only slightly better than that of a random classifier; and they have assumed that all users and their tweets are trustworthy. These problems, therefore, need to be tackled in order to achieve more reliable public opinion measurements from social media data.

This thesis work deals with the challenges and the problems described above. Colombia 2014 presidential election was proposed as case study. First, a Twitter user classification system was developed by following supervised and unsupervised learning approaches, in order to distinguish spammer accounts from non-spammer ones. Next, a lexical normalization system of Spanish tweets was built to normalize out-of-vocabulary (OOV) words to their canonical form. This normalization corresponds to an application of the spell checking task in NLP, and its development was mainly carried out by using finite-state transducers and a statistical language model. Lastly, a sentiment analysis system of Spanish tweets was developed by implementing supervised learning techniques in order to apply it for voting intention inference.

Because of the case study of the thesis is the Colombian election, a brief background on it is presented below.

1.1 Background on the Colombian Election

In race for the presidency in 2014 five candidates competed for the most important Colombian political office, including the incumbent President Juan Manuel Santos. Óscar Iván Zuluaga, Marta Lucía Ramírez, Clara López, and Enrique Peñolosa were the other candidates. Santos was supported by the grand center-right coalition called "Unidad Nacional", composed by the political parties "Partido de la U", "Cambio Radical", and "Partido Liberal Colombiano". Zuluaga was the "Centro Democrático" right-wing party candidate, founded by the former President of Colombia Álvaro Uribe. Ramírez, also a right-wing alternative, was chosen by the "Partido Conservador Colombiano". The main opposition party "Polo Democrático Alternativo" supported the López's left-wing candidacy. Lastly, Peñalosa was candidate by the "Partido Alianza Verde".

The presidential election was held under a two-round voting system. In the first round, held on May 25, 2014, no candidate received an absolute voting majority, and for that a run-off took place 21 days later between Zuluaga and Santos, whom were the highest-polling candidates with 29.28% and 25.72% support from voters, respectively. In the run-off election, Santos was re-elected President with 50.98% support. Table 1.1 shows important events of the Colombian election. Regarding presidential debates, those where all contenders participated and national television broadcasted them are cited.

TABLE 1.1. Schedule of important events of the presidential election

Date	Event
May 6, 2014	A person hired by the Zuluaga campaign was captured and accused of illegally obtaining classified information [22]
May 8, 2014	Accusation against the Santos campaign for the presidency in 2010, for allegedly received funds from drug trafficking activities[18]
May 17, 2014	A video revealed publicly shows that Zuluaga met with the person captured and accused of spying [20]
May 22, 2014	First presidential debate
May 25, 2014	Election day
Jun 5, 2014	Second presidential debate
Jun 9, 2014	Third presidential debate
Jun 15, 2014	Run-off election day

1.2 Goal

The main goal of the thesis is to design and implement a sentiment analysis model of Spanish tweets using machine learning techniques. In order to achieve this goal, the following specific objectives were proposed:

1. To design and implement a data extraction model from Twitter.
2. To design and implement a Twitter user classification system to distinguish spammer accounts from non-spammer ones, using machine learning techniques.
3. To design and implement a text preprocessing model for Spanish tweets.
4. To study, select and implement machine learning techniques to rate the opinion orientation in Spanish tweets.
5. To apply the sentiment analysis system of Spanish tweets in order to gauge public opinion regarding Colombia 2014 presidential election.

1.3 Contributions

The contributions of the thesis are mainly grouped into technical and data-based contributions and publications. These are listed below.

Technical and Data-based Contributions

1. A dataset of 513,324 tweets dealing with Colombia 2014 presidential election, contributed by 149,831 different users.
2. A dataset of 3,455 Twitter users classified into spammer and non-spammer.
3. A Twitter user classification system that distinguishes spammer accounts from non-spammer ones.
4. A lexical normalization system of Spanish tweets.

5. A dataset of 1,030 Spanish tweets classified into positive, negative, and neutral. The domain from which the dataset was extracted, is the Colombian election.

6. A sentiment analysis system of Spanish tweets.

Publications

1. Jhon Adrián Cerón-Guzmán and Elizabeth León, *Detecting Social Spammers in Colombia 2014 Presidential Election*. Paper accepted for publication at: 14th Mexican International Conference on Artificial Intelligence MICAI 2015. This work received the "Best Paper Award, First Place."

2. Jhon Adrián Cerón-Guzmán and Elizabeth León-Guzmán, *Lexical Normalization of Spanish Tweets*. Paper accepted for publication at: 2nd International Workshop on Natural Language Processing for Informal Text (NLPIT 2016). In conjunction with 25th International World Wide Web Conference (WWW 2016).

3. Jhon Adrián Cerón-Guzmán and Elizabeth León-Guzmán, *A Sentiment Analysis System of Spanish Tweets and Its Application in Colombia 2014 Presidential Election*. Paper pending publishing.

1.4 Thesis Outline

This document is organized as follows:

- **Chapter 2** presents a literature review on the topics that the thesis deals with.
- **Chapter 3** discusses the spammer detection problem on Twitter in a political domain.
- **Chapter 4** describes a lexical normalization system of Spanish tweets.
- **Chapter 5** presents a sentiment analysis system of Spanish tweets and its application in Colombia 2014 presidential election.
- **Chapter 6** concludes the thesis and shows directions for future work.

CHAPTER 2

Literature Review

The goal of this chapter is to present a literature review on the topics that the thesis deals with. First, the spammer detection problem on Twitter is briefly described. Next, as an application of the spell checking task in Natural Language Processing, literature on lexical normalization of tweets is presented. Then, a revision on sentiment analysis in Twitter is discussed. As a final point, the recent and appealing line of work that deals with election outcome prediction from Twitter data, is discussed.

2.1 Detecting Social Spammers on Twitter

The spammer detection problem on Twitter has been widely studied in the literature. Mainly, there have been proposed several detection models using machine learning techniques to classify users into two classes: spammer and non-spammer. Benevenuto et al. [7] proposed a set of features, grouped into content-based features and behavior-based features, to support an automatic classification system of Twitter users. Chu et al. [12] aggregated a new user class: cyborg, a mix between spammer and non-spammer. To conduct the classification, they designed a four-component system to detect patterns in posting behaviors, identify spam content in tweets, and capture spammer-like behaviors. Instead, Yang et al. [92] and Amleshwaram et al. [4] focused on studying evasion tactics and designing new and more robust features to achieve a high spammer detection rate, while a negligible fraction of non-spammers were misclassified. In the research presented in Chapter 3, the features used for distinguishing spammer accounts from non-spammer ones, have been widely used in the literature by their proven highly effectiveness. Likewise, while most of the works have followed a supervised learning approach for tackling the spammer detection problem, in the research, in addition to a supervised classification, an approximation of solution is implemented under a semi-supervised setting.

The spammer detection has also been approached as an anomaly detection problem. Miller et al. [52] used clustering techniques to predict a class of a Twitter account, treating outliers as spammers. Despite all those efforts, the spammer detection problem on Twitter is still an open challenge [79].

2.2 Lexical Normalization of Tweets

A large body of literature exists on lexical normalization of tweets written in English. Han and Baldwin [32] characterized the types of OOV words present in tweets, finding that these correspond to a heterogeneous collection of ill-formed words and proper nouns. As a critical step in the process of lexical normalization, they proposed an automatic method to distinguish between correct OOV words and ill-formed OOV words, and for the latter normalization candidates were suggested, from which the best were selected based on contextual inference, dependency features, and string similarity measures.

Han, Cook, and Baldwin [33] proposed a dictionary-based approach to normalize OOV words that fails to adapt to new domains, thus recording low values of recall, and does not normalize complex cases of OOV words with two or more possible standard lexical forms, for which contextual information may be used in order to resolve ambiguities.

A common shortcoming of these cited works is that they both have focused on one-to-one normalization, i.e., one OOV word is normalized to one standard lexical form. However, OOV words may also be normalized by splitting fused words, which is why a one-to-many normalization approach is required.

As an initiative to foster research in the field, the TweetNorm 2013 shared task [2] was organized to create a benchmark for lexical normalization of Spanish tweets. The resources provided by the organizing committee were used to conduct experiments and evaluate the performance of the system presented in Chapter 4. The highest ranked participating systems are described below.

Porta and Sancho [60] used several weighted finite-state transducers that were applied in cascade to generate the confusion set of each OOV word. The standard lexical forms were suggested by their similarity to the graphemes or phonemes that make an OOV word, and the candidate selection was made by the application of a trigram language model.

Gamallo, Garcia, and Pichel [27] distinguished normalization candidates between primary and secondary variants. The former correspond to candidates that only differ from an OOV word with regard to one of several linguistic phenomena (i.e., uppercase/lowercase confusion, character repetition, or frequent spelling errors); otherwise, secondary variants were generated by using the Levenshtein distance. They also used a language model in the candidate selection. Without using contextual information, Ageno et al. [1] selected the normalization candidate from a confusion set generated by a set of expert modules, through a weighted voting scheme. Saralegi and Vicente [68] assumed that all the named entities recognized by a third-party language analyzer were correct OOV words; however, as it will be proved in the Chapter 4, these must be carefully treated because users misuse uppercase in tweets, e.g., to denote emphasis, thereby producing false positive of named entities.

Finally, Cotelo et al. [16] conducted a complete study of the types of OOV words present in Spanish tweets. They proposed a modular architecture for lexical normalization, in which each module addressed a specific error phenomenon. Thus, each module suggested normalization candidates, and the best was selected through a weighted voting scheme.

2.3 Sentiment Analysis of Twitter Data

Sentiment analysis is the computational approach that deals with the task of determining the opinion orientation expressed in a text [47]. In this way, an opinion, which may well be either a sentiment, attitude, emotion or appraisal about an entity or its attributes, is rated as positive, negative, or neutral [31]. Because of the brevity of tweets, given their 140-character limit, sentiment classification of them is mainly made at the document level, assuming that the tweet text expresses opinions on a single entity.

Sentiment classification is not a recent task. To the best of our knowledge, the seminal works on sentiment analysis were carried out by Pang et al. [57] and Turney [82]. They proposed the approaches typically used in sentiment analysis implementations, whatever the source of the textual data: classification based on unsupervised learning [82] and classification based on supervised learning [57]. The former, also known as lexicon-based classification, uses a lexicon to filter words of known polarity in a document in order to assign it a label class: positive, if the number of positive words is greater than the number of negative words; negative, if the number of positive words is less than the number of negative words; otherwise neutral. Although the lexicon-based classification is appealing due to its simplicity and the ease of implementing it, it is not able to understand subtle expressions (e.g., sarcasm), and the different meanings that a same word may acquire in nonidentical domains [47]. Instead, state-of-the-art in sentiment analysis of tweets follows the supervised classification approach [54, 31, 37]. Below, the literature review on sentiment analysis of Twitter data is presented.

Mohammad et al. [54] used a Support Vector Machine (SVM) with a large number of features, which are grouped into: word ngrams, character ngrams, all-caps, part-of-speech (POS), hashtags, lexicons, punctuation, emoticons, elongated words, clusters, and negation. To classify a tweet, they first normalized it by replacing URLs and user mentions by placeholders, and then tokenized and POS tagged it. The vectorization [48], this is, the process of transforming textual data into numerical feature vectors of fixed length, was based on the bag-of-words (BOW) representation.

Miura et al. [53] developed a sentiment analyzer based on supervised text classification. They used the Logistic Regression algorithm to predict the label class of a tweet; however, because the class distribution was unbalanced, they introduced an weighting factor w_l to adjust a probability output $Pr(l)$ of class l, and thus the class with the highest updated probability was chosen. The groups of features was inspired by Mohammad et al. [54], namely: word ngrams, character ngrams, lexicons, clusters, and word senses. They also used a spelling corrector and a word sense disambiguator, which were applied in the text preprocessing.

Amir et al. [3] proposed the following three groups of features: word-based, lexicon, and syntactic. In order to compute the word-based features, they used, in addition to the BOW representation, the word2vec method [51]. Under this method, neural networks are trained to learn vector representation of words from a (commonly) large dataset; this vector representation is characterized by full density and low dimensionality. To assign a class label to a tweet, they implemented the Logistic Regression algorithm. As in [53], they introduced class weights set to be inversely proportional to the class distribution.

Hagen et al. [31] reproduced four state-of-the-art approaches to sentiment analysis in Twitter and combined them in an ensemble. The ensemble combination was not based on

the final decision of each approach (reimplemented as a classifier), but rather it requested the classifiers' probabilities for each class. Thus, the class with the highest average probability was chosen.

Saralegi and Vicente [68] developed a supervised system using three groups of features to support the classification of Spanish tweets. In order to transform the tweet text into a feature vector, they used the BOW representation filtered by the words of a polarity lexicon, in addition to the frequency of each POS tag, and the frequency of each emoticon and interjection type (positive and negative). Because most of the lexicons exist for English, they created a polarity lexicon for Spanish by semi-automatically translating an English polarity lexicon and automatically extracting the words most associated with a certain polarity from a training corpus.

Díaz-Galiano and Montejo-Ráez [17] used the word2vec and doc2vec methods for vector representation [51, 44]. Doc2vec, unlike word2vec, induces a vector representation for each paragraph. In order to represent a tweet as a feature vector, they concatenated the vector obtained by the doc2vec and the vector as the average of the word2vec vectors. In this way, a 500-dimensional vector fed a SVM to assign a class label to a tweet.

2.4 Predicting Voting Intention from Twitter Data

Predicting real-world events from social media data has turned into an appealing line of research from social sciences to computer science [69]. What people say about an electoral race or its contestants has been exploited to predict or forecast election outcomes, given the large amount of user-generated content by the ever-growing virtualization of human behavior. In this way, a new alternative to gauge public opinion has been developed, which also benefits from the increasing cost and difficulty of the traditional polling [35]. However, the numerous researches that claim to have successfully forecasted, face reproducibility problems [28]. More importantly, most of the works dealing with election outcome prediction were only post hoc analysis [50].

Tumasjan et al. [81] claimed that the proportion of tweets mentioning an electoral option can be considered as a plausible reflection of its voting share. They reported a very low error in forecasting the 2009 German Bundestag elections on the assumption that the larger number of tweets, the larger the vote. However, despite being aware of the low representativeness, such that a small number of users generated most of the tweets, they claimed that Twitter is a predictor of election outcomes. This proves that Twitter's user base is not a representative sample of the population [28]. Jungherr et al. [40] reproduced the research, finding that the apparent success was due to data manipulation on the part of researchers.

O'Connor et al. [55] correlated sentiment scores with opinion polls in order to determine if the sentiment classification would respond faster to changes in the consumer confidence or the presidential job approval, compared with the traditional opinion polling. They defined the sentiment score to be the ratio between the number of positive tweets and the number of negative tweets; tweets were labeled by a lexicon-based classifier. Based on the obtained results, they claimed that sentiment analysis is a substitute and supplement for the traditional polling. However, Metaxas et al. [50] concluded that a lexicon-based classifier wrongly interprets the subtleties of propaganda and disinformation, and that its performance is only slightly better than that of a random classifier.

In the same way that social media provides an appealing source of information, it could, however, contain noisy, useless, and irrelevant information. In this regard, Metaxas et al. [50] warned: "spammers and propagandists write programs that create lots of fake accounts and use them to tweet intensively, amplifying their message, and polluting the data for any observer." Those problems, therefore, need to be tackled in order to achieve reliable public opinion measurements.

Gayo-Avello [28] presented a comprehensive literature revision on electoral prediction from Twitter data. He concluded with recommendations for future research from which the following are highlighted: the state-of-the-art approaches of sentiment analysis in Twitter should be implemented, and spam and disinformation should be removed from the study data. These recommendations have greatly inspired this research.

Shi et al. [71] and Tsakalidis et al. [80] developed prediction models that did not strictly rely on sentiment analysis or Twitter volume. Shi et al. [71] correlated a set of 19 features with opinion polls in order to predict the Republican Party presidential primaries in 2012. They also claimed that the traditional electoral polling can be supplemented or supplanted with analysis of Twitter data. Tsakalidis et al. [80] developed regression models to predict elections for multiple countries. Their results in most of the cases were better than those of the poll-based prediction, although they used a lexicon-based classifier.

2.5 Summary

This chapter presented a literature review on the topics that the thesis deals with. First, a brief discussion on spammer detection on Twitter was held. Next, literature on lexical normalization of tweets was presented. Then, a revision on sentiment analysis in Twitter was discussed. Finally, the line of research that deals with election outcome prediction from Twitter data, was described.

CHAPTER 3

Detecting Social Spammers on Twitter

Social media has turned into a rich source of information about individuals, society, and the world in general [69]. Thus, the impressive amount of user-generated content on a diversity of issues and topics has brought new research opportunities for understanding social behavior. Among them, the one related to public opinion has caught the attention of current research. Data collected from social media are nowadays used to measure public opinion in regards with important events such as political elections. Around elections time, a significant number of research works have used Twitter data to predict election outcomes, based on opinions or possible voting intentions expressed by users [28].

In the same way that social media provides an appealing source of information, it could, however, contain noisy, useless, and irrelevant information. New forms of spam have been spread to manipulate social media discourse with rumors, misinformation, political astroturf, slander, or simply noisy messages [26]. Therefore, social media poses problems regarding data quality and credibility. As one of the ways to deal with the former, i.e., the quality of the data, a spammer detection system is presented in this chapter. Regarding the latter, several works have studied the credibility of newsworthy information propagated through Twitter [8, 9].

Although the spammer detection on Twitter has been addressed in the literature, most of the researches on measuring public opinion from Twitter data have ignored this problem assuming that all users and their tweets are trustworthy, while few works have recognized it and accordingly adopted measures for denoising in Twitter data [28]. Thus, the goal of this research is to shed light on the importance of noise removal when public opinion measurements are conducted using Twitter data.

This chapter is organized as follows: First, the methodology for collecting the data and creating a labeled collection of users is described in Section 3.1. Next, the proposed detection system and the evaluation of its performance are discussed in Section 3.2. In Section 3.3, results were obtained by applying the proposed system on the collected data. Finally, Section 3.4 concludes the chapter with a summary of the work presented.

TABLE 3.1. Summary of the collected Twitter data on the electoral process

Candidate	Collection Period	Query Terms	Tweets	Users
Santos	Apr 30–Jun 24, 2014	"Juan Manuel Santos", @JuanManSantos	332,575	117,783
Zuluaga	Apr 30–Jun 24, 2014	"Oscar Ivan Zuluaga", @OIZuluaga	202,405	81,979
Ramírez	Apr 30–May 29, 2014	"Marta Lucia Ramirez", @mluciaramirez	9,273	6,198
López	Apr 30–May 29, 2014	"Clara Lopez", @ClaraLopezObre, @ClaraPresidenta	13,711	9,457
Peñalosa	Apr 30–May 29, 2014	"Enrique Penalosa", @EnriquePenalosa	12,072	7,391
Blank vote	Apr 30–Jun 24, 2014	"Voto Blanco", Blanco	39,203	27,148

3.1 Data Collection and Ground Truth Creation

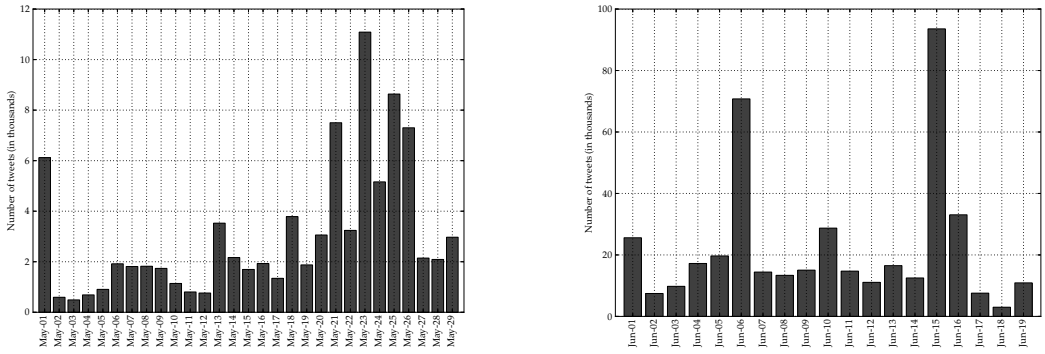
In this section, a Twitter dataset collected during the presidential election is described. Additionally, the strategy designed to create a labeled collection of spammers and non-spammers is discussed.

3.1.1 Dataset

During the course of the presidential election, in a two-month period from April 30, 2014 to June 24, 2014, a dataset ("COpres14") of 513,324 tweets contributed by 149,831 different users was collected from Twitter Search API [86]. To conduct the research relying on tweets referring to the aforementioned political context, a set of criteria was defined to filter tweets. Thus, only tweets containing at least one keyword or hashtag related to the presidential election (i.e., *elecciones* (`elections`), *presidenciales* (`presidential`), *#Elecciones2014* (`#2014Election`), *#ColombiaElige* (`#ColombiaChooses`), *#EleccionesColombia* (`#ColombianElection`), and *#ColombiaDecide* (`#ColombiaDecides`)) and full name or user mention that identifies a given candidate were collected. Table 3.1 shows the amount of collected data in terms of users and tweets per candidate, and the query terms related to each. Note that a larger amount of data were collected for the candidates Santos and Zuluaga, as well as for *blank vote*, because they were the contestants in the run-off election. Figure 3.1 shows the data collection activity on a daily basis divided into periods comprising the first round and the run-off of the election; Figure 3.2 shows the daily tweet mention distribution per candidate. As can be seen in Figure 3.1, local maximums were produced one day after the presidential debates and during and after the election days, being the global maximum the run-off election day with over 93K tweets; instead, Figure 3.2 could explain the challenging first round election, in terms of user mention, where the protagonists were Santos and Zuluaga, and likewise how Santos took advantage in race for the Colombian presidency in the run-off election.

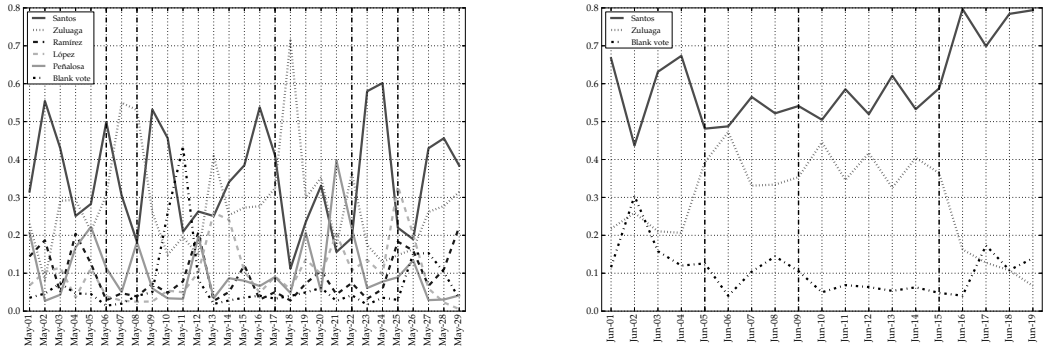
Because not enough tweets were collected per user,[1] up to the 40 most recent tweets were crawled from its timeline. From 149,831 users, in a second stage of data collection

[1]Average number was 1 tweet.



(a) First round (b) Run-off

FIGURE 3.1. Number of tweets collected on a daily basis



(a) First round (b) Run-off

FIGURE 3.2. Fraction of tweets mentioning each presidential candidate. Vertical dotted lines represent the events highlighted in Table 1.1

conducted from February 19, 2015 to March 26, 2015, a dataset of 134,625 users and 1,765,225 tweets was collected using Twitter User Timeline API [85], since 4,805 users set their profile as private, 8,462 users changed their *@username*, and 1,939 users were suspended by Twitter. For the work presented in the remainder of this chapter, unless otherwise stated, was used the dataset collected in this second stage.

3.1.2 Ground Truth

For the purpose of this work, a labeled collection of Twitter accounts was needed to support the ability of the detection system to distinguish spammer accounts from non-spammer ones. To this end, a random sample of 49,358 users was drawn from the dataset, and a three-part strategy was designed to label them in a semi-automatic way. As a result, a labeled collection of 3,455 users was created, including 2,660 spammers and 795 non-spammers.

3.1.2.1 Harmful Link Detection

The first part of the strategy consisted in automatically identifying spam using five URL blacklists. From the 1,765,225 tweets in the dataset, 341,352 URLs were extracted and resolved of obfuscation using a web crawler developed to follow chain of redirects until reach target pages. The web crawler was able to resolve HTTP status codes, META tags and Javascript code used for redirects.

While the second data collection proccess ran, URLs in tweets were extracted and their target pages, whole chain of redirects, and tweet ids containing them, were saved in a database. Then, a batch script was developed to check the URLs crawled against the following blacklists: Google Safe Browsing, PhishTank, SURBL, Spamhaus, and URIBL [29, 59, 76, 74, 88]. The first one enables to check URLs against Google's constantly updated lists of suspected phishing and malware pages. Phishtank is a crowdsourcing service in which phishing sites are submitted, verified, and tracked in a semi-automatic way. The last three provide constantly updated lists of domain names that have appeared in unsolicited emails. Because of the slow detection rate of these services [30], the blacklist detection process was performed multiple times until April 20, 2015. If an URL was marked as harmful by two blacklists, no more checks were needed.

As result of this first part of the strategy, 3,302 URLs were detected as malicious links. However, during a manual revision of these URLs, it was found that only 12 of them, shared in 34 tweets, corresponded to true positives; therefore, the 7 users tweeting them were labeled as spammers. To explain this high false positive rate, it is important to note that 2,576 URLs from 9 different domains were (possibly) erroneously marked because spammers abuse of them (e.g., URL shortener services, such as *bit.ly*), or DNSBL services mark domains as spam when they appeared in unsolicited messages and not because they hosted malicious or harmful content. In this regard, the scientific community is encouraged to define a clear spam policy when URL blacklists are used, as it was done in [30] where an URL whitelist was created to minimize false positive rate.

In addition to the 7 spammer accounts resulting of this process, 86 accounts were labeled as malicious because URLs shared by them were detected by Twitter's anti-spam filter, and in a further revision of their profile, enough evidence was found to label them as

such. From these accounts, 56 users belonged to a political astroturf campaign intended to artificially inflate support for the Zuluaga's campaign.

3.1.2.2 Suspended Accounts

The first part of the strategy detected spam if an URL shared in a tweet was marked by blacklists. However, spam is not limited to posting harmful links, but rather, in a broader meaning, to any unsolicited, repeated actions that negatively impact other users [84]. This includes aggressive following behavior, posting unsolicited mentions and duplicate tweets, abusing of trending topics to grab attention, and share links unrelated to tweet content [87]. Based on Twitter's algorithm for suspending accounts that fall into some of prohibited behaviors, the second part of the strategy labeled as spammers to the suspended accounts identified in the second stage of data collection. From the 49,358 Twitter accounts, 674 were suspended, while 1,939 were found in the population of 149,831 users.

By labeling in this way, an assumption was made that the suspended accounts were manipulated to spam purposes, and they were not legitimate accounts, e.g., belonging to real people. Although any false positive can be resolved by an user requesting to be unsuspended [83], a further process was performed to verify that the suspensions were caused by prohibited behaviors. Thus, a random sample of 100 accounts was drawn from the 674 suspended, and their *timeline* was reconstructed from all tweets collected in the first stage of data collection. Considering tweet content, shared URLs, tweeting sources, numbers of followers and friends, longevity of account, and number of tweets posted, suspended accounts were manually investigated.

From the 100 accounts under analysis, 89 of them were labeled as spammers, while for the remaining there was not enough evidence because few tweets were collected. It is important to highlight that, in contrasts to results in [78], no harmful links were found in 356 tweets from the *timeline* reconstructed for the suspended accounts sample. However, 30 accounts were suspended because, in addition to automation behavior, their tweeting source pointed to a same spam URL.

From these results, it is possible to conclude that most of the accounts manually investigated were presumably well suspended, although more evidence was necessary to label the entire sample as spammers. In this way, all suspended accounts were used as spammers in the labeled collection of users.

The authors acknowledge that the discussed strategy may not be potentially useful, because few tweets were collected per user in the first stage of data collection and Twitter has already identified what characterize these accounts. However, a hypothesis is made in the direction that some (additional) knowledge could be extracted from Twitter's anti-spam algorithm, and this would be acquired by the proposed detection system.

3.1.2.3 Manual Labeling

The common strategy used in the literature to create a ground truth is to manually label a sample of users [91, 7, 12]. Given this, the third part of the strategy consisted in drawing a random sample of 1,245 users from the 49,358 Twitter accounts, and labeling them as either spammer or non-spammer based on their profile data and up to the 100 most recent tweets from each user timeline.

To conduct the manual labeling, a set of criteria was defined to classify users in the sample. In this way, every user was analyzed taking into consideration its tweet contents, shared URLs, tweeting sources, numbers of followers and friends, and number of statuses posted. Thus, an user was labeled as spammer if there was not evidence of original, intelligent, or human-like contents; URLs posted were spam or unrelated to tweet content; it abuses of trending topics to grab attention, or its tweet contents are unrelated to hashtags; posting duplicate content; or automation predominates account's behavior, such as tweeting from automated sources or automatic statuses from news sites or blogs. This set of criteria was motivated by the work in [12]. Otherwise, an user was classified as non-spammer. From the set of 1,245 users, 599 were classified as spammers, while 117 were not labeled because doubt predominated to assign a class or few tweets had been posted by them, so they were excluded from the labeled collection.

While the manual labeling process was performed, duplicate tweets were found over multiple accounts,[2] including 29 users that were not in the sample. In particular, 243 of these accounts were grouped into two political astroturf campaigns. The first one consisted of 150 spammers accounts used during the election to artificially inflate support for the Santos's campaign. Likewise, the second one was intended to create the appearance of wide support for the Zuluaga's campaign. Although this second campaign shared the same goal that the aforementioned one in the harmful links detection, they differ each due to tweet contents and because the activity of the former was based on mainly retweets, while the latter posted some tweets with human-like contents.

From this third part of the strategy, a labeled collection of 1,128 users was created, including 628 spammers and 500 non-spammers, plus 295 verified accounts in the dataset of 149,831 users that were used as non-spammer instances.

3.2 The Spammer Detection System

In this section, the proposed system for detecting spammers on Twitter is discussed. Firstly, a set of features is defined to support the discriminative power in spammer detection. Secondly, a first approximation of the detection system is implemented by following a semi-supervised learning approach, where in addition to the ground truth set, the system infers a classification function on the entire data space, including unlabeled data. Lastly, a supervised learning approach is followed to implement a second approximation of the detection system. Here, it is also discussed the least number of tweets used for spammer detection, and the importance of the features to achieve this goal.

3.2.1 Features

Unlike non-spammer accounts, spammer ones are presumably designed to infiltrate social media without being detected by security systems such as spam filters, and mimic human behavior to gain confidence of real people in order to obtain benefits for which they were made. These can be of kind commercial (e.g., advertising), harmful (e.g., malware or phishing), or even political (e.g., artificially inflating support for a candidate) [26]. That is why it would be expected that spammers differ from non-spammers on what they post,

[2]Using Twitter's search page: `https://twitter.com/search-home`

TABLE 3.2. List of features

Category	Feature
User	user has a profile description
User	account is verified
User	age of the user account, in days (AGE)
User	number of followings ($NFing$)
User	number of followers ($NFers$)
User	reputation ($\frac{NFers}{NFing+NFers}$)
User	number of tweets
User	fofo rate ($NFers/NFing$)
User	following rate ($NFing/AGE$)
Content	user mention (@) ratio
Content	unique user mention (@) ratio [45]
Content	URL ratio
Content	hashtag (#) ratio
Content	average of tweet content similarity [45]
Behavior	retweet rate
Behavior	reply rate
Behavior	mean of inter-tweeting delay
Behavior	standard deviation of inter-tweeting delay [4]
Behavior	average of tweets per day
Behavior	average of tweets per week
Behavior	number of tweets from manual devices [12]
Behavior	number of tweets from automated devices [12]
Behavior	distribution of tweets in each of the 8-3 hour periods within a day [49]

such as shared URLs, user mentions, hashtags, and content originality; on their behavior, such as devices used, tweeting frequency, etc.

Based on these assumptions, a set of features was proposed to support the ability of the detection system to distinguish spammer accounts from non-spammer ones; these are grouped into three categories. The features were collected from different works in the literature, and most of them have been widely used for detecting spammers on Twitter.

3.2.1.1 User-based Features

The first group of features is based on account's information that an user provides, summarizes its lifetime on Twitter, or describes its friendship network. These features are extracted from tweet's metadata, and comprise a list of 9 attributes presented in Table 3.2.

3.2.1.2 Content-based Features

These features are computed from tweet content, and defined to distinguish content that spammers usually post from original, intelligent, or human-like contents. To determine which is the least amount of data required to successfully distinguish spammer from non-spammer instances, the attributes of this category (as seen in Table 3.2) are computed

from the 5, 10, 20, and 40 most recent tweets from each user. In particular, the following text preprocessing technique was applied to normalize tweet content and compute the *average of tweet content similarity* feature: removing URLs, user mentions, hashtags, emoji unicode,[3] and HTML symbols; replacing time patterns with a placeholder (e.g. "HORA" (`time`)); normalizing character repetition (based on grammatical rules of the Spanish language, e.g. "holaaa" → "hola" (`hello`)); replacing emoticons with textual portrayals; normalizing and replacing laughs (e.g. "jajaja" → "RISA" (`laugh`)); unification of punctuation marks [89]; replacement of numeric patterns with a standard text; handling negation [54]; whitespace-based tokenization, and stop words removal. Once tweets are normalized, they are represented as vectors using the TF-IDF weighting scheme [48], and the cosine similarity is computed between them. Final value of the aforementionated feature is the average of similarity between the set of unique pairs of tweets.

3.2.1.3 Behavior-based Features

The last category of features was proposed to capture the behavior that characterizes each user class. Like content-based features, the attributes of this category (as seen in Table 3.2) are computed from the 5, 10, 20, and 40 most recent tweets from each user.

To count number of tweets posted from manual and automated devices, 773 different sources found in the 20 most recent tweets collection were manually classified. Thus, if a device requires human participation, it was classified as manual. Otherwise, the device was classified as automated.

3.2.2 Semi-Supervised Detection

The first approximation of solution for the spammer detection problem on Twitter was developed under a semi-supervised learning approach. Semi-supervised learning is halfway between supervised and unsupervised learning, where in addition to unlabeled data, the algorithm is provided with some supervision information [11]. Inspired by this approach, a two-stage study was conducted. In the first stage, a clustering algorithm was applied, and using labeled samples of the ground truth, a class was assigned to each cluster. Here, it was assumed that if points are in the same cluster, they are like to be of the same class [11]. In the second stage, a non-generalizing machine learning technique was used to predict a class for a given Twitter account based on the clustered data space.

3.2.2.1 Clustering

To conduct the clustering analysis, a dataset of 46,074 users from the sample of 49,358 was created from their 20 most recent tweets, including 1,350 samples (658 spammers and 692 non-spammers) of the ground truth, since for 3,080 users were not collected enough tweets, and 204 (101 spammers and 103 non-spammers) were excluded to test the prediction system (to be discussed later).

The CHAMELEON algorithm [41] and the set of proposed features were used to find groups of users with similar characteristics, which could be clustered instances of the spammer and non-spammer classes. CHAMELEON is a clustering algorithm that finds clusters

[3] `http://apps.timwhitlock.info/emoji/tables/unicode`

of diverse shapes, densities, and sizes, modeling data items in a sparse graph, where several subclusters are found using a graph-partioning algorithm, and then, repeatedly combining subclusters using an agglomerative hierarchical technique. As a result of the analysis, a 12-way clustering solution was found, using the 1,350 labeled samples like clue about what is the cluster tendency towards a class. However, in a further statistical analysis conducted per feature, and a manual labeling of samples randomly drawn from each cluster, 7 clusters were discovered like the best candidates to potentially contain spammer and non-spammer instances, including 4 clusters that sum 2,046 samples with spammer-like behaviors, and the remaining of 17,211 samples with non-spammer's behaviors. Table 3.3 shows the results of clustering validation. External measures, *Entropy* and *Purity* [77] were computed using 837 labeled users in the 7 clusters, plus 354 users manually classifed (108 spammers and 246 non-spammers) of random samples drawn from each cluster. Overall *Entropy* and *Purity* for the cluster solution are 0.153 and 0.975, respectively. *Samples labeled* column shows percentage of instances per class that a cluster contains of the 1,191 users labeled, while I_1 is an internal measure computed using the Euclidean distance.

Table 3.3. Clustering validation results

Cluster	Tendency	Size	Samples Labeled		I_1	Entropy	Purity
			Spammer	Non-Spammer			
1	Spammer	45	11.48%	0.00%	0.01	0.0	1.0
2	Spammer	224	26.79%	0.00%	0.01	0.08	0.99
3	Spammer	756	20.41%	0.29%	0.75	0.1	0.99
4	Spammer	1,021	37.5%	0.29%	0.42	0.06	0.99
5	Non-spammer	1,665	0.5%	33.24%	0.28	0.13	0.98
6	Non-spammer	5,161	1.28%	25.3%	0.21	0.31	0.95
7	Non-spammer	10,385	2.04%	40.88%	0.33	0.3	0.95

3.2.2.2 Predicting

Using the labeled collection of 19,257 users (the clustered data space), and the set of features discussed, a non-generalizing machine learning technique was implemented to predict a class for a given Twitter account. Here, it was investigated the feasibility of applying an inductive method inferred on the (partial) data space, instead of one that only takes into account labeled points. The k-Nearest Neighbors (k-NN) algorithm was chosen to classify unseen data points because, in addition to it has been commonly used in previous researches on spammer detection, it possibly best matches with the CHAMELEON algorithm, where a k-nearest neighbor graph is used to cluster the dataset.

The Scikit-learn [58] implementation of the k-NN algorithm was used to classify Twitter accounts into spammer and non-spammer. Firstly, the number of neighbors (i.e., the k parameter) was searched using cross validation with 5-folds on the clustered data space. Secondly, unseen data points were classified based on the class the majority of their 7 closest samples. Table 3.4 shows the results of applying the semi-supervised system on the test dataset of 204 users (101 spammers and 103 non-spammers). This result indicates that the system correctly identifies 86.14% of spammers (true positive rate), at a cost of misclassifying 11.65% of non-spammers (false positive rate).

Manually examining the users being classified by mistake, it was observed that for non-spammers misclassified as spammers, synchronization of their activity on other social

TABLE 3.4. Confusion matrix for the semi-supervised detection

		Predicted	
		Spammer	Non-Spammer
Actual	Spammer	87	14
	Non-Spammer	12	91

media platforms (such as YouTube[4] and Instagram[5]), which generates a high number of tweets from automated devices, and posting a significant number of their tweets to trending topics or with several mentions, corresponded to typical behaviors of spammers [4]. Regarding false negatives (spammers misclassified as non-spammers), some of them were legitimate accounts hijacked by spammers to spread spam without permission of their owners, while others were occasional spammers and a large number of tweets (e.g., 100) would be required to make a correct classification.

3.2.3 Supervised Detection

The common strategy to tackle the spammer detection problem on Twitter is based on a supervised learning setting. Under this setting, a machine learning algorithm infers a classification model from a labeled collection, and then the extracted knowledge is applied to classify an unseen user as either spammer or non-spammer. Following this approach, a spammer detection system was developed. Firstly, it is discussed the classification algorithm selection between a range of machine learning techniques widely used in the literature. Secondly, it is determined the least amount of information (i.e, number of tweets) required to the proposed system detects a large number of spammers in early stages, at a cost of misclassifying a small number of non-spammers. Lastly, the importance of the features is discussed.

3.2.3.1 Selecting the Classification Technique

To conduct the selection, three machine learning techniques were implemented on the ground truth set, and then the performance of them were compared using the standard information retrieval metrics of recall, precision, and F1-score. In this stage, the features were computed from the 20 most recent tweets from each user. Thus, a dataset of 1,554 users was created (759 spammers and 795 non-spammers), since for the remaining of the ground truth no enough tweets were collected. This dataset was splitted into training and test sets. The first one, which consists of 66% of the samples, was used to optimize the hyperparameters for each technique using 5-fold cross validation. The second one was used to perform independent tests and accordingly select the best classification technique.

The Scikit-learn implementations of the Support Vector Machines (SVM), Random Forest (RF), and Gaussian Naive Bayes (NB) were used to train the techniques and

[4] `https://www.youtube.com/`
[5] `https://instagram.com/`

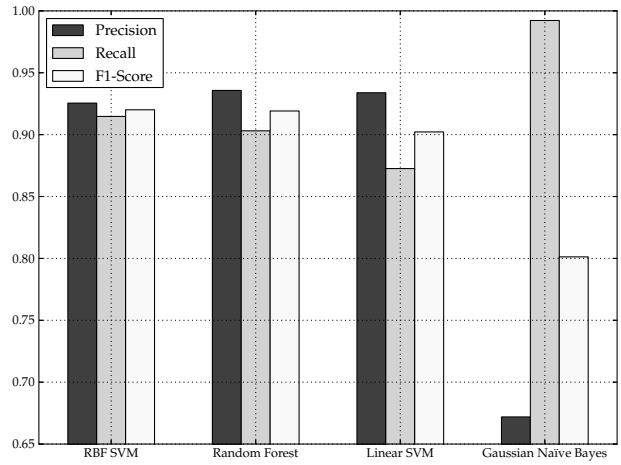


FIGURE 3.3. Performance comparison of the classification techniques

conduct the evaluation for each. In particular, two flavors of the SVM technique were implemented. In the first one, which is based on the LIBLINEAR library [25], only the complexity parameter was optimized, since the 'linear' kernel is fixed. In the second one, which is based on the LIBSVM library [10], both the kernel and the complexity parameters were optimized. Figure 3.3 shows the performance of each algorithm on the test dataset. To select between SVM with 'radius basis function' (RBF) kernel and RF, which achieve the highest performance, the following tiebreaker rule was applied: for each spammer correctly classified, 1 point was added, while for each non-spammer misclassified as spammer, 1.5 points were subtracted. Thus, the RF was selected as the best classification technique, and therefore the system implementation was based on it.

3.2.3.2 Number of Tweets Required for Detecting Spammers

A critical limitation of previous works [73] is related to the number of tweets required to detect spammers before they achieve the purpose for which they were designed, e.g., to spread viruses and malwares, and a large amount of Twitter accounts may be harmed. In this stage, the goal was to determine the least number of tweets required to detect a large number of spammers, at a cost of misclassifying a small number of non-spammers. Here, an assumption was made that a small number of tweets could reduce the delay between spammer account creation and its detection.

Figure 3.4 shows the numbers of tweets used for detecting spammers and true positive and false positive rates for each.[6] This result was obtained by implementing the RF algorithm on the ground truth set, and computing the features from the 5, 10, 20, and 40 most recent tweets from each user in it. The experimental setup described in the previous section was also applied. From this result, it is determined that 10 tweets is the least amount of information required for achieving a balance between detecting a large number of spammers and misclassifying a relatively small number of non-spammers, with true positive and false positive rates of 93.02% and 7.78%, respectively. Table 3.5 shows the performance of the detection system using 10 tweets on the test dataset. In order to improve the

[6]Note that when no tweets are used to detect spammers, only the user-based features are computed to make classification.

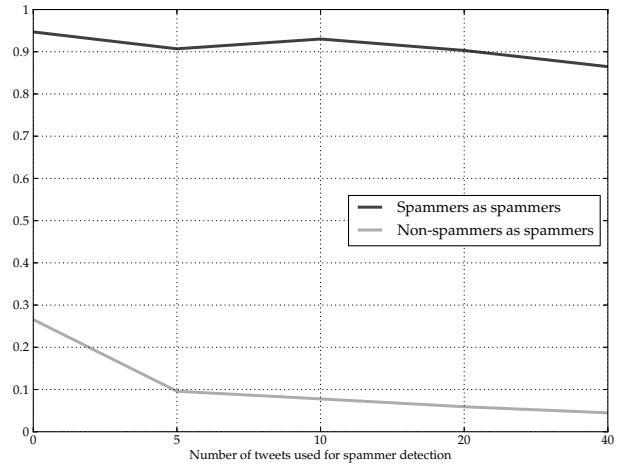


FIGURE 3.4. Number of tweets required for detecting spammer

TABLE 3.5. Confusion matrix for the supervised detection using 10 tweets

		Predicted	
		Spammer	Non-Spammer
Actual	Spammer	253	19
	Non-Spammer	21	249

performance of the RF algorithm, the standard boosting and bagging techniques [77] were applied to build ensemble classifiers. However, no significant improvement was obtained.

In a further manual revision of the users misclassified as spammers, it was observed that the following reasons could cause the unwanted performance: several tweets were generated from automated devices because users had granted permissions to third parties apps for automatic tweeting; time interval between tweet posting was short, which could seem to a regular timing pattern; and a significant number of tweets were posted with several mentions and URLs. Regarding spammers misclassified as non-spammers, randomness on the inter-tweet time interval and posting tweets from manual devices (mainly from Twitter's web interface), caused spammers acquire non-spammer's characteristics.

3.2.3.3 Importance of the Features

The detection ability of the proposed system depends on the relative power of its features in discriminating between spammer instances and non-spammer ones. Thus, to identify which features contribute the most at time of discriminating each user class, the effectiveness of the 30 features was evaluated. In every test, only one feature was used to implement the Random Forest algorithm under the experimental setup described above, and thus evaluate its importance. Table 3.6 shows the top 10 features sorted by following the tiebreaker rule described in the classification technique selection. Surprisingly, the *url ratio* is not among the most important features, as it has been reported in [7, 45, 12],

which would indicate that spammers have evolved their tactics, and possibly redefined their objectives. However, devices used for tweeting are still highly discriminative features [12]. This is because spammers are more enticed by automated devices, due to the cheap and practical, instead of interact with devices that require human participation, e.g., when logging into Twitter's web interface. The other features correspond to the results reported in the literature [7, 45, 12], among them, short lifetime that characterizes spammer accounts.

TABLE 3.6. Ranking of the detection performance using only one feature

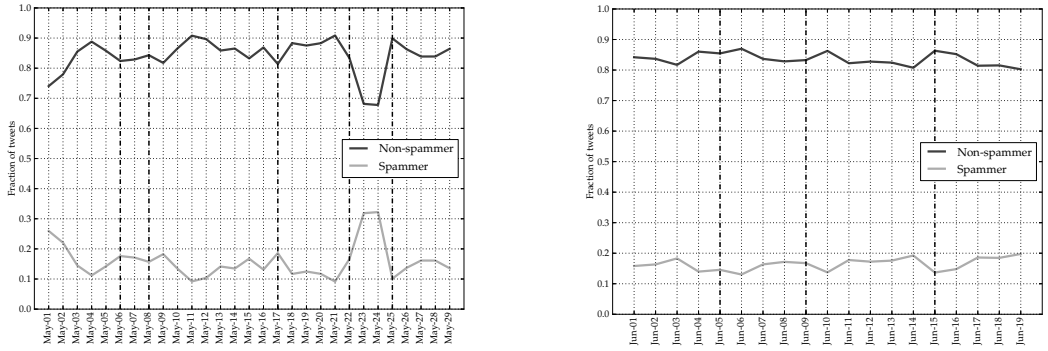
Category	Feature	TPR (%)	FPR (%)
Content	user mention (@) ratio	68.02	17.41
User	age of the user account	77.57	26.67
Behavior	number of tweets from manual devices	41.91	8.15
Behavior	number of tweets from automated devices	41.91	8.15
Behavior	retweet rate	81.99	35.19
User	fofo rate	69.12	30.74
Content	unique user mention (@) ratio	68.02	31.85
Behavior	distribution of tweets between 3 and 5 am	36.03	13.33
Behavior	mean of inter-tweeting delay	69.12	35.56
User	number of followers	70.96	38.52

3.3 Discussion

So far, it has been discussed that social media has turned into an appealing source of information due to the large amount of user-generated content on a diversity of issues and topics [69]. However, new forms of spam have been spread to manipulate social media discourse with rumors, misinformation, political astroturf, slander, or just noisy messages [26]. Because of this, it is imperative to distinguish noisy, useless, and irrelevant information from valuable data. To this end, it has been proposed two approaches of solution to the spammer detection problem on Twitter based on machine learning approaches: semi-supervised and supervised learning. In the first approach, it was investigated the feasibility of applying an inductive method inferred on the entire data space, including unlabeled data. Although the performance of the semi-supervised detection is good, even outperforming to other works in the literature [7] (in terms of accuracy), it is not better that the performance of the supervised detection, the second approach. In this one, an achievement of this research was to obtain an overall accuracy of 93%, using 10 tweets and only one Twitter API method.[7] Instead, other works have required a larger amount of data (e.g. 40 tweets [92, 4], and up to 100 tweets [12]) and several API methods (e.g., to compute features such as *bi-directional links* [92], and *unsolicited mentions* [4]) to achieve an overall accuracy ranging from 96% [12] to 98% [4].

Moreover, to quantify the importance of adopting measures to filter noise in Twitter data, the proposed detection system was applied on the COpres14 set. As result, 22.01% of users in the dataset were classified as spammers, whom generated 15.67% of tweets. Figure 3.5 shows the daily fraction of tweets generated by each user class during the course of

[7] *GET statuses/user_timeline*: `https://dev.twitter.com/rest/reference/get/statuses/user_timeline`



(a) First round (b) Run-off

FIGURE 3.5. Daily fraction of tweets generated by each user class. Vertical dotted lines represent the events highlighted in Table 1.1

the presidential election; as can be seen, the fraction of tweets generated by spammers remained above 10%, achieving a maximum peak of 32% one day before the election day. These findings prove that spammers could significantly affect public opinion measurements from Twitter data, and thus, for example, it can be stated that the mere number of tweets is not a reliable source of voting preferences, as it was reported in [81]. In this way, with relatively high fractions of spammers and tweets generated by them, it is emphasized the importance of adopting measures to distinguish noisy, useless, and irrelevant information from valuable data in order to achieve reliable measurements.

3.4 Summary

In this chapter, the spammer detection problem on Twitter has been studied. Colombia 2014 presidential election has been proposed as case study to shed light on the importance of noise removal when public opinion measurements are conducted using Twitter data. As the most important result of the research, it is concluded that adopting measures for denoising in Twitter data becomes a major issue in order to achieve more reliable measurements, with approximately 22% of accounts in the dataset classified as spammers and 15% of tweets contributed by them.

To classify a given Twitter user as either spammer or non-spammer, two detection systems were developed by following the machine learning approaches: semi-supervised and supervised learning. Both systems were implemented on a labeled collection of users semi-automatically classified into spammer and non-spammer, and built by considering common practices to create a ground truth for spammer detection. Although the performance of the semi-supervised system outperforms other proposals in the literature, supervised learning is the most appropriate approach to deal with the problem, taking into consideration the results obtained in this research and those reported in the literature. Likewise, the Random Forest algorithm is the best classification technique to predict the class of a given Twitter user.

As a final point, by using few resources for the detection task, in terms of number of tweets and Twitter API methods, the proposed system was able to achieve a high detection rate of spammers and its overall accuracy is competitive to the state-of-the-art.

CHAPTER 4

Lexical Normalization of Spanish Tweets

Twitter data have brought new opportunities to know what happens in the world in real-time, and conduct studies on the human subjectivity on a diversity of issues and topics at large scale, which would not be feasible using traditional methods. However, as well as these data represent a valuable source, a vast amount of noise can be found in them. Because of the brevity of tweets and the widespread use of mobile devices [75], Twitter is also a rich source of noisy data containing many non-standard word forms [32]. That is why several lexical variation phenomena that occur on the content generation, need to be tackled in the pipeline of a Natural Language Processing (NLP) task, in order to improve the quality of natural language analysis [32, 16].

Initial lexical normalization approaches of tweets have focused on English [32, 33]. However, Twitter content in languages such as Spanish rapidly increases [70], for which normalization strategies to deal with lexical variation phenomena (e.g., initialisms, shortenings, homophonic confusion, character repetition, and misuse of uppercase) present in Spanish tweets becomes a major issue in order to boost NLP applications that exploit user-generated content in that language.

In this chapter, a lexical normalization system of Spanish tweets is presented. The overall process of lexical normalization follows a sequential approach that goes from the detection of out-of-vocabulary (OOV) words in a tweet, to the correction candidate selection for a word from a set of normalization proposals. In contrast to [32], where a one-to-one normalization approach was developed, this is, one OOV word is normalized to one standard lexical form, this work proposes a one-to-many normalization approach to deal with word segmentation problems such as lack of spacing between words.

This chapter is organized as follows: The system architecture, which is divided into three components and considers a post-processing step, as well as the set of lexical resources employed by the system to suggest normalization candidates for OOV words, may be read in Sections 4.1 and 4.2, respectively. In Section 4.3 the experimental development of the system and the evaluation of its performance are presented. Finally, Section 4.4 concludes the chapter with a summary of the work presented.

4.1 The System Architecture

The overall process of lexical normalization follows a sequential approach that goes from the detection of OOV words in a given tweet, to the correction candidate selection for a word. The approach is structurally divided into three components that are discussed below: in the first one, a third-party language analyzer is used for performing tokenization of tweets and lexical analysis of in-vocabulary (IV) words, while non-standard word forms are detected (i.e., OOV words); the second one generates normalization candidates for each OOV word (i.e., the confusion set); finally, the third one selects the best candidate from the confusion set of each OOV word, taking into account contextual information. After the selection, a post-processing is applied to uppercase the correction for a word when one of several conditions is satisfied.

4.1.1 Detecting OOV Words

The morphological analyzer of FreeLing [56] is used for detecting OOV words. Once a given tweet has been tokenized, each resulting token is passed through a set of basic modules (e.g., dictionary lookup, suffixes check, detection of numbers and dates, named entity recognition, etc.) for identifying standard word forms and other valid constructions. If a token is not recognized by any of the modules, it is marked as OOV. In this step, specific Twitter terms like user mentions (e.g., @twitter), hashtags (e.g., #twitter), and "RT" (retweet) and other expressions such as URLs are treated as valid constructions.

While some experiments were conducted on a development set, an unexpected behavior of the Named Entity Recognition module of FreeLing was observed.[1] Specifically, tokens starting with a capital letter or completely written in uppercase were mostly wrong recognized as named entities, because the capitalization rules [62] are not taken into account by users who write tweets misusing uppercase, e.g., to denote emphasis, thereby producing false positives of named entities that must be carefully treated. For example, given the tweet *"Lo mejor es que me da igual todo SOI FELIZ"* (`The best is that i do not care anything, I AM HAPPY`), the tokens *"SOI FELIZ"* (`i am happy`) are wrongly recognized as an entity, being *"SOI"* a typo of the standard word form *"soy"* (`i am`) and *"FELIZ"* (`happy`) a standard word form. Therefore, each token recognized as an entity is looked up in the dictionary of standard words, and if there is not an entry matching the token, it is marked as OOV.

4.1.2 Confusion Set Generation

Once the OOV words have been detected, a first issue to be tackled is to determine if a given OOV is either a correct word that is not in the standard dictionary, or a token requiring to be normalized to its canonical form. That is, it is explicitly necessary to distinguish between correct OOV words and ill-formed OOV words [32]. For the former, the OOV itself would remain unchanged, while for the latter, several lexical variation phenomena should be dealt with, including: character repetition (e.g., *claseeeesss* → *clases* (`classes`)) and alteration of valid onomatopoeia (e.g., *ajajajjaja* → *ja*); language-dependent orthographic errors [27, 16]: missing of diacritical marks (e.g., *tendre* → *tendré* (`i will have`)), uppercase/lowercase confusion (e.g., *francia* → *Francia* (`France`)), and letter confusion

[1] Using the "basic" recognizer.

(v → b, ll → y, h → 0); initialisms (e.g., *xk* → *porque* (`because`)), shortenings (e.g., *pa* → *para* (`for`)), and letter omissions [60]; homophonic confusion (e.g., *pokitin* → *poquitín* (`little bit`)) [16] and standard non-correct endings (e.g., *mercao* → *mercado* (`market`)) [60]; and word segmentation problems (e.g., *alomejor* → *"a lo mejor"* (`at best`)) [60]. Thus, in order to determine if an OOV token is a correct word, it is first included in its confusion set; if the OOV token is which best fits a language model, it is then considered as correct. The confusion set generation is discussed below.

A confusion set is generated by either one of two sequential phases. The first one involves a set of simple rules intended to tackle some of the most common lexical variation phenomena present in Spanish tweets. If an OOV word is recognized by one of these rules, its canonical form is provided; otherwise, the second one generates normalization candidates that are identical or similar to the graphemes or phonemes that make the OOV word. During the entire process, the consecutive repetition of a same letter is reduced to one and two occurrences, thus generating three different versions of the OOV word (the first one being the OOV itself, the second one with no letter repetition,[2] and the third one with at most two consecutive repetitions); the repetition reduction is inspired by the approach proposed in [1]. Likewise, the treatment of unknown characters, taking as reference the Spanish alphabet [61], is conducted by representing them to their closest ASCII variant, using the *unidecode*[3] module for the mapping.

The confusion set generation comprises a set of finite-state networks developed to deal with the foregoing lexical variation phenomena. These networks are computationally efficient for tasks such as natural-language morphological analysis, and for their mathematical properties, which are well understood, it is allowed to manipulate and combine them in ways that would be impossible using traditional algorithmic programs [6].

4.1.2.1 Matching Simple Rules

As discussed above, a set of simple rules was designed to tackle some of the most common lexical variation phenomena present in Spanish tweets, namely: alteration of valid onomatopoeia, missing of diacritical marks, initialisms, and shortenings. These rules are described as regular expressions compiled into finite-state transducers using the Foma library [36]. Thus, if an OOV word is accepted by the language of a network, i.e., a transducer, its canonical form is provided; otherwise, if the OOV word is rejected by the set of transducers, the process of the confusion set generation is applied.

Note that in this phase, two or more target words can be suggested, instead to be directly provided the normalization of an OOV word. For example, let the OOV word be *"siii"*, and the network be the composition of the transducers to deal with the missing of diacritical marks,[4] and accept all the valid Spanish words, the following normalization candidates are returned by the process of generation: *"si"* (`if`) and *"sí"* (`yes`).[5] However, in most of the cases, the normalization of an OOV word is directly provided. In this regard, initialisms and shortenings are dealt with a network whose language consists of frequent OOV words that may be included within these phenomena, and their normalization can be provided unambiguously. The language is a normalization dictionary that was built

[2] Considering the formation of the digraphs "rr" and "ll" as valid repetitions in the Spanish language.

[3] `https://pypi.python.org/pypi/Unidecode`

[4] This transducer generates other versions of the OOV word by accentuating its vowels (only one vowel is accentuated per version).

[5] The composition is made in the order in which the transducers are stated.

from the most frequent OOV words observed in a development set, and Internet slang used in Spanish tweets.

4.1.2.2 Generating the Confusion Set

To tackle the remaining lexical variation phenomena, in this phase a set of normalization candidates is generated. The candidates are elements of the union of the standard dictionary and the gazetteer of proper nouns, which are identical or similar to the graphemes or phonemes that make an OOV word.

Firstly, the OOV word is converted into its phonetic transcription using the International Phonetic Alphabet (IPA). The phonetic transcription makes the IPA phonemes /ʝ/ and /ʎ/ equivalent, which is a phenomenon that occurs in many dialects of the Spanish language [65]. The linguistic phenomenon of *seseo* [63], homophonic confusion, and standard non-correct endings are also modeled by the transducer that makes the transcription; the phenomena of uppercase/lowercase confusion and letter confusion are implicitly modeled. Thus, normalization candidates are suggested by their phonetic similarity to the OOV word. Likewise, a suffixes search is performed to recognize inflected forms that are not found in the standard dictionary, namely: clitics attached to verbal forms of infinitive, imperative, and gerund, i.e., enclitic pronouns; adverbs ending in *-mente*; and diminutive forms of adjectives, adverbs and nouns [93]. Therefore, if the OOV is recognized as an inflected word form, it is suggested as a candidate with the proper accentuation.

Secondly, if no candidates are generated by the above approach, in this one the most complex cases of OOV words, mainly characterized by the phenomena of letter omissions and word segmentation problems, are tackled. To deal with the first phenomenon, a transducer inserts only one vowel in any position of the OOV word, as it was proposed in [60]. Inspired by [60] and [1], the second phenomenon is dealt with the composition of the transducers that insert blanks (_) between letters, and accept the language $L(\_L)^+$, where L is the language of all entries in the standard dictionary. Also, candidates within a Levenshtein distance of one are generated. Finally, the Longest Common Subsequence is calculated between the OOV word and each normalization candidate, thus removing candidates whose ratio is below a threshold.

4.1.3 Candidate Selection

To select the best normalization candidate from the confusion set of an OOV word, contextual information is taken into account. However, because in a context can co-occur several non-standard forms, the selection of the normalization candidates corresponds to the candidates combination that maximizes an objective function. Therefore, the combinations are evaluated against a language model implemented with the Kenlm tool [34], and the one that obtains the highest log probability of sequence of words is selected. The model was estimated from the Spanish Wikipedia corpus.

4.1.4 Post-processing

Even though the best normalization candidates have been selected, it may still be required a post-processing for the proper capitalization of them. In the Spanish language [62], capital letters are used to differentiate proper nouns from common nouns; however, the

case is also required by the punctuation. For proper nouns, their capitalization is selected by the application of the language model. Otherwise, a selected candidate is uppercased if one of the following conditions is satisfied:

1. If the OOV word is in initial position of tweet.
2. If the OOV word is preceded by one of the following punctuation marks: ". ! ?".[6]
3. If the previous token is an ellipsis mark, and the OOV word begins with an uppercase letter.

4.2 Resources

The system employs a set of lexical resources to suggest normalization candidates for OOV words. The set consists of a dictionary of Spanish standard words, a normalization dictionary, and a gazetteer of proper nouns, which are described below. While the normalization dictionary was entirely handcrafted, the other lexical resources have been built in an automatic way.

4.2.1 Standard Dictionary

The dictionary of Spanish standard words was built from the FreeLing Spanish dictionary of 556,509 forms. This one was expanded with the entries in the Dirae lexicon[7], and by generating verbal forms of *voseo* [64]. The final dictionary consists of 619,550 standard word forms. Note that the inflected forms of enclitic pronouns, adverbs ending in *-mente*, and diminutives were not added as entries in the dictionary; they are recognized during the process of confusion set generation by applying a set of morphological rules.

4.2.2 Normalization Dictionary

The normalization dictionary consists of 529 entries that correspond to initialisms, shortenings, and other Internet slang expressions frequently used in Spanish tweets. This resource was mainly built from the most frequent OOV words observed in a development set, for which can be provided their normalization unambiguously. In this way, for each OOV word in the dictionary, its canonical form is included.

4.2.3 Gazetteer of Proper Nouns

The list of proper nouns was built by following the approach in [68]. The Spanish Wikipedia corpus was morphologically analyzed using FreeLing, being the forms categorized as named entities considered as candidates to build the gazetteer. These forms were tokenized and those unigrams whose frequency was greater than 100 and higher than their lowercased variant, and which were not found in the standard dictionary, were taken as secure proper nouns. In this way, a gazetteer of 53,531 unigrams was built.

[6] The double quotes are used to enclose the punctuation marks.
[7] http://dirae.es/

4.3 Experiments and Evaluation

In this section the experimentation with the system to set its parameters and the evaluation of its performance are discussed. To carry out these processes, the resources provided by the organizing committee of the TweetNorm 2013 shared task [2], which are a benchmark for lexical normalization of tweets written in Spanish, have been employed. These resources comprise a set of 937 tweets divided into two collections, i.e., the development corpus (475 tweets) and the test corpus (462 tweets), with 653 and 572 OOV words manually annotated, respectively.[8] The RAE dictionary[9] was taken as reference to determine the standard word forms.

In Section 4.3.1 the metrics used to evaluate the performance of the system are described. The experimentation conducted on the development corpus to set the parameters of the system, regarding the ability of OOV words detection and the contextual information required to select normalization candidates, is discussed in Section 4.3.2. Finally, the evaluation of the system on the test set, conceiving it as a whole and by isolating its components, is discussed in Section 4.3.3.

4.3.1 Metrics

The detection rate metric evaluates the ability of the system to detect OOV words. The candidate coverage metric [16] measures how many times the confusion set of an OOV word includes the proper correction, regardless of the candidate selection. The standard information retrieval metrics of precision, recall, and F1-score have been also used to evaluate the performance of the system. These five metrics are described below:

$$Detection\ rate = \frac{\sum_{t \epsilon T} \sum_{oov \epsilon OOV'_t} [oov \epsilon OOV_t]}{\sum_{t \epsilon T} |OOV_t|}$$

$$Candidate\ coverage = \frac{\sum_{t \epsilon T} \sum_{oov \epsilon OOV'_t} [corr^t_{oov} \epsilon C^t_{oov}]}{\sum_{t \epsilon T} |OOV'_t|}$$

$$Precision\ (P) = \frac{\sum_{t \epsilon T} \sum_{oov \epsilon OOV'_t} [sel^t_{oov} = corr^t_{oov}]}{\sum_{t \epsilon T} |OOV'_t|}$$

$$Recall\ (R) = \frac{\sum_{t \epsilon T} \sum_{oov \epsilon OOV'_t} [sel^t_{oov} = corr^t_{oov}]}{\sum_{t \epsilon T} |OOV_t|}$$

$$F1\text{-}score\ (F) = \frac{2 \times P \times R}{P + R}$$

Where,

- T is the collection of tweets, OOV_t the set of OOV words in tweet $t\ \epsilon\ T$, and OOV'_t the set of detected OOV words.

[8] At the time of tweets collection retrieval, on July 2015, several tweets had been removed from the Twitter historical data. Therefore, of 1,162 tweets provided, 937 were retrieved by using the Twitter REST APIs.

[9] `http://dle.rae.es/`

TABLE 4.1. Performance of the system on the test set with different isolated components. All values are given in percentages

Active components	Candidate coverage	P	R	F1
All	79.65	69.65	69.41	69.53
All – Matching simple rules	68.95	55.96	55.77	55.86
All – Confusion set generation	63.68	61.40	61.19	61.29
All – Phonetic transcription – Suffixes search	80.35	64.39	64.16	64.27
All – Vowels insertion – Edit distance – Split words	74.21	69.30	69.06	69.18
All – Post-processing	72.46	62.11	61.89	62.00

- C^t_{oov} is the confusion set of an OOV word, sel^t_{oov} the normalization candidate selected from the confusion set, and $corr^t_{oov}$ the proper correction of the OOV word.

4.3.2 Setting the System

A critical step of the process of lexical normalization has to do with the ability of the system to detect OOV words. Thus, if several OOV words are not detected, recall could significantly drop. For this reason, two approaches of OOV words detection have been proposed: in the first one, the tokens without analysis by the morphological analyzer of FreeLing are treated as OOV words; in the second one, in addition to the tokens without analysis, the named entities are also treated as OOV words, as it was discussed in Section 4.1.1. To select between these approaches, several experiments were conducted on the development set. With a detection rate of 98.77%, and 23 percentage points higher than that of the first approach, the second one was selected to detect OOV words.

Likewise, the amount of contextual information required by the candidate selection component was determined. In this way, different orders of the language model were evaluated. As result, the highest precision was obtained by a 3-grams language model, 71.78%, above the 71.32% that both 2- and 4-grams achieve.

4.3.3 Results and Evaluation

Table 4.1 shows the performance values of the system on the test set. Here, the system was evaluated by activating all its components; likewise, a further evaluation was conducted by isolating each component in order to determine its contribution to the overall performance. In general, the system achieves a F1-score of 69.53%, with a precision of 69.65% and recall of 69.41%.

TABLE 4.2. Performance comparison with participating systems in the TweetNorm 2013 shared task

Rank	System	R
1	RAE [60]	78.32%
2	**ours**	**69.41%**
3	Citius-Imaxin [27]	66.43%
4	UPC [1]	65.56%
5	Elhuyar [68]	63.81%

Clearly the results show that the greatest room for improvement is in the candidate selection component. The language used in the Spanish Wikipedia corpus, from which the language model was estimated, is characterized by being more formal than that used in Twitter, where predominates a free writing style. Therefore, it should be considered a language model that adapts to informal genres. Despite the prevalence of OOV words in Twitter data, it is not difficult to build a large corpus of tweets with only standard word forms [32]. As future work, it is planned to build a large corpus of tweets from user accounts who, in theory, write correctly, e.g., journalists and mass media.

With regard to the contribution of each component to the overall performance, it is observed that the matching simple rules and the post-processing are which contribute the most, thus when these are deactivated, the performance of the system drops significantly. The most complex cases of OOV words are mainly dealt with the phonetic transcription and the suffixes search; instead, deactivating the normalization candidates generation through the vowels insertion, edit distance, and splitting of words, causes a negligible drop in the overall performance.

Finally, the performance of the system was compared with the participating systems in the TweetNorm 2013 shared task. This comparison was made by considering only the 462 tweets of the test set that were retrieved. The best five results sorted by recall are shown in Table 4.2.[10] For reference, recall average of the 13 participating systems was 56.52%, with the lowest score being 33.93%.

4.4 Summary

In this chapter, a lexical normalization system of tweets written in Spanish was proposed. The system correctly detects OOV words in tweets and suggests normalization candidates that are identical or similar to the graphemes or phonemes that make an OOV word. To select the best normalization candidate for an OOV word, contextual information is taken into account. However, because in a context can co-occur other OOV words, the selection corresponds to the candidates combination that best fits a trigram language model. Although most of the cases the correct normalization of an OOV word is suggested, there is a room for improvement in the candidate selection, which is not properly adapted to the informal genre and the free writing style of Twitter.

[10] Recall was the official metric used to evaluate the performance of the systems in the shared task.

CHAPTER 5

Sentiment Analysis of Spanish Tweets and Its Application in the Colombian Election

What people say about issues of their everyday life, the society, or the world in general has turned into a rich source of information to support processes of decision making, in both businesses interested in positioning a brand or a product and governments that seek to understand people's needs in order to define public policies. The large amount of user-generated content on social media platforms such as Twitter has brought new opportunities to explore the human subjectivity at large scale. However, the analysis of this information through manual means becomes an impractical task, which is why automated methods for processing large amounts of user-generated content are required. Sentiment analysis or opinion mining is the computational approach of studying "people's opinions, appraisals, attitudes, and emotions toward entities, individuals, issues, events, topics and their attributes" [47]. Unlike other text classification tasks, the goal of sentiment analysis is to rate the sentiment expressed as positive, negative, or neutral [31].

There is a large number of works on sentiment analysis in the literature. Typically, these have implemented one of the following approaches: lexicon-based classification [82], machine learning classification [57], and a combination of both. Although the first one is appealing due to its simplicity and the ease of implementing it, it is not able to understand subtle expressions (e.g., sarcasm) and the different meanings that a same word may acquire in nonidentical domains [47]. Motivated by these reasons, the system presented in this chapter follows the machine learning classification approach, which corresponds to the state-of-the-art of sentiment analysis [54, 31, 37].

As an application of the sentiment analysis system, the voting intention inference in Colombia 2014 presidential election is presented. However, the inference was not strictly relied on sentiment analysis or Twitter volume, which are the methods commonly used to address this task [28]. Although most of the researches on voting intention inference have included, as part of the study data, lots of fake accounts that produce noisy tweets [50, 28], spammer accounts and their tweets were removed from the data used in this research.

This chapter is organized as follows: First, the datasets used in the research are described in Section 5.1. Then, the sentiment analysis system of Spanish tweets and the voting intention inference are discussed in Sections 5.2 and 5.3, respectively. Finally, Section 5.4 concludes the chapter with a summary of the work presented.

5.1 Data

Throughout this section the datasets used in the research are described. First, the dataset used to train and evaluate the sentiment analysis system is discussed in Section 5.1.1. Then, the aggregated polling data to infer voting intention are presented in Section 5.1.2.

5.1.1 Sentiment Labeled Dataset

A sentiment analysis system is highly sensitive to the domain from which the data used to train it were extracted [47]. For this reason, a system may obtain poor results when it is applied on a dataset whose domain differs from the one learned [72]. Although an important and large resource exists to build sentiment analysis systems of Spanish Twitter data [90], it was decided to create a dataset by labeling a random sample of tweets drawn from the COpres14 set (described in Chapter 3), because the context of extraction of the cited resource has a Spain-focused bias and its domain deals with topics of general interest from politics to celebrities; instead, a dataset whose domain exclusively deals with the topic of interest of this research, results more appropriate because the sentiment analysis system will be applied on the COpres14 set to infer voting intention in the Colombian election.

A random sample of 1,170 tweets was drawn from the COpres14 set. In order to label a tweet as either positive, negative, or neutral, two volunteers assigned it a label according to the sentiment they understood was conveyed.[1] If there was no agreement among volunteers regarding the polarity label of a tweet, a third independent volunteer was heard. Volunteers agreed in 40,38% of tweets, thus supporting the statement with respect to humans often disagree on the sentiment of a text [90]. In total, 1,030 tweets were labeled by 234 different volunteers, each of whom labeled 10 tweets. The class distribution is as follows: positive, 22.43%; negative, 41.1%; and neutral, 36.47%.

The sentiment labeled dataset was splitted into two sets: 80% of tweets were used as the training set and the remaining as the test set. The splitting of the data was performed in a stratified way.

5.1.2 Opinion Polls

Opinion polls were collected and then aggregated by hand. These were filtered by their survey period in order that they corresponded with the collection period of Twitter data. Thus, polls conducted between May 03, 2014 and May 15, 2014 were collected to infer voting in the first round election (as seen in Table 5.1); likewise, those whose survey period was in the range from May 26, 2014 to June 04, 2014 were collected to infer voting in the run-off election (as seen in Table 5.2).

In order to aggregate the data from the opinion polls, the approach proposed by Tsakalidis et al. [80] was applied as follows: because a poll is usually conducted in a two- or three-day period, the voting share each candidate would receive is treated as if the election would have taken place on any of these days. If two or more polls were conducted on a same day, the voting share of each candidate is considered as the weighted average value, using the sample size of every poll as the weight. Finally, the votes of undecided voters were proportionally distributed to all contenders.

[1] A website was developed to help volunteers to manually label tweets.

TABLE 5.1. Opinion polls to gauge voting intention in the first round election

Pollster	Survey period	
	Start date	End date
Infométrika [38]	May 03, 2014	May 06, 2014
Centro Nacional de Consultoría [19]	May 06, 2014	May 10, 2014
Cifras y Conceptos [13]	May 09, 2014	May 12, 2014
Datexco [24]	May 10, 2014	May 13, 2014
Gallup [5]	May 10, 2014	May 13, 2014
Ipsos [42]	May 13, 2014	May 15, 2014

TABLE 5.2. Opinion polls to gauge voting intention in the run-off election

Pollster	Survey period	
	Start date	End date
Cifras y Conceptos [13]	May 26, 2014	May 27, 2014
Centro Nacional de Consultoría [23]	May 26, 2014	May 30, 2014
Datexco [21]	May 31, 2014	June 04, 2014
Gallup [14]	May 31, 2014	June 03, 2014
Cifras y Conceptos [13]	May 31, 2014	June 03, 2014
Ipsos [43]	June 02, 2014	June 04, 2014

5.2 The Sentiment Analysis System

In this section, the system architecture, which was conceived as a pipeline where a tweet is first preprocessed and then transformed into a numerical feature vector understood by the machine learning classifier, is described. Likewise, the experimental development of the system and the performance evaluation on the sentiment labeled dataset are discussed.

5.2.1 The System Architecture

The tweet text is passed through the pipeline of the sentiment analysis system in order to label it as either positive, negative, or neutral. Note that this is a multiclass classification task and only one label is assigned to a tweet. Firstly, the text is normalized by applying the common strategies of text cleaning and normalization, namely: URLs removal, word lengthening treatment, emoticons replacement to their canonical form, among others. However, in this stage a further normalization is applied by taking into account syntactic and lexical information. Secondly, the text is transformed into numerical features represented as a n-dimensional vector. Lastly, the machine learning classifier receives the feature vector as input and produces one label as output. This pipeline will be described below.

5.2.1.1 Preprocessing

The process of text cleaning and normalization is performed in two phases. The first one comprises a set of simple rules commonly used in the literature. The second one takes into account syntactic and lexical information to restore out-of-vocabulary (OOV) words to their canonical form and handle negation.

Basic Preprocessing:

- Removing URLs and emails.
- HTML entities are mapped to textual representations (e.g., "<" → "<").
- Specific Twitter terms such as mentions (@user) and hashtags (#topic) are replaced by placeholders.
- Unknown characters are mapped to their closest ASCII variant, using the *unidecode*[2] module for the mapping.
- Consecutive repetitions of a same character are reduced to one occurrence.
- Emoticons are recognized by simple regular expressions and classified into positive and negative according to the sentiment they convey (e.g., ":)" → "EMO_POS", ":(" → "EMO_NEG").[3]
- Unification of punctuation marks [89].

Advanced Preprocessing. Once the set of simple rules has been applied, the tweet text is tokenized and morphologically analyzed by FreeLing [56]. In this way, for each resulting token, its lemma and Part-of-Speech (POS) tag are assigned. Taking these data as input, the following advanced preprocessing is applied.

- **Lexical normalization.** Each token is passed through a set of basic modules of FreeLing (e.g., dictionary lookup, suffixes check, detection of numbers and dates, named entity recognition, etc.) for identifying standard word forms and other valid constructions. If a token is not recognized by any of the modules, it is marked as out-of-vocabulary (OOV) word. Then, a confusion set is formed by normalization candidates that are similar to the graphemes or phonemes of the OOV word. These candidates are elements of the union of a dictionary of Spanish standard words and a gazetteer of proper nouns. The best normalization candidate for the OOV word is which best fits a statistical language model. The language model was estimated from the Spanish Wikipedia corpus. Lastly, the selected candidate is capitalized according to the capitalization rules of the Spanish language [62]. Extensive research on lexical normalization of Spanish tweets can be read in Chapter 4.
- **Negation handling.** The common strategy for handling negation in tweets has followed the approach proposed by Pang et al. [57]. In accordance with that proposal, a negated context is defined as a segment of the text that starts with a negation word (e.g., *no*, *nunca* (`never`)) and ends with a punctuation mark (e.g., ",", ".", "!"). In a negated context, every token is affected by adding it the "_NEG" suffix.

  Inspired by the previous approach, in this work a negated context has been defined as follows: A segment of the tweet that starts with a (Spanish) negation word and ends with a punctuation mark (i.e., "!", ",", ":", "?", ".", ";"), but only the first token (from left to right) labeled with a specific POS tag (i.e., verb, adjective, or common noun) is affected. This definition was result of experimentation conducted on the training set.

[2] `https://pypi.python.org/pypi/Unidecode`
[3] A list of emoticons was retrieved from `https://en.wikipedia.org/wiki/List_of_emoticons`

5.2.1.2 Feature Extraction

In this stage, the normalized tweet text is transformed into a feature vector that feeds the machine learning classifier. The features are grouped into basic features and word-based features.

Basic Features:[4]

- All-caps (1): the number of words completely in uppercase.
- Elongated words (1): the number of words with more than two consecutive repetitions of a same character.
- Punctuation marks (2): the number of consecutive repetitions of exclamation marks, question marks, and both punctuation marks (e.g., "!!", "??", "?!") and whether the text ends with an exclamation or question mark.
- Emoticons (3): the number of occurrences of each class of emoticons (i.e., positive and negative) and whether the last token of the tweet is an emoticon.
- Lexicons (3): the number of positive and negative words, relative to the ElhPolar lexicon [67]. In a negated context the label of a polarity word is inverted (i.e., positive words become negative words, and vice versa). Additionally, a third feature labels the tweet with the class whose number of polarity words in the text is the highest.
- Negation (1): the number of negated contexts.
- POS (13): the number of occurrences of each Part-of-Speech tag.

Word-based Features. The fixed-length set of basic features is always extracted from tweets. However, the tweet text varies from another in terms of length, number of tokens, and vocabulary used. For that reason, a process that transforms textual data into numerical feature vectors of fixed length is required. Thus, the tweet text can be represented as a numerical feature vector understood by the machine learning classifier. This process, known as vectorization, is performed by following two approaches in the pipeline of the sentiment analysis system. The first one is based on the (traditional) tf-idf weighting scheme [48]. The second one, which is based on a more recent proposal [51], produces a vector representation for each word, and under this approach, a tweet is modeled as the average of its word vectors.

- Tf-idf: under this scheme, each document (i.e., a tweet text) is represented as a vector $d = \{t_1, \ldots, t_n\} \; \epsilon \; \mathbb{R}^V$, where V is the size of the vocabulary. In order to assign to term t a weight in document d, the *term frequency-inverse document frequency* is computed as follows [48]:

$$tf\text{-}idf_{t,d} = tf_{t,d} \times \log \frac{N}{df_t},$$

  where $tf_{t,d}$ is the number of occurrences of term t in document d, N represents the total number of documents in the collection, and df_t is the number of documents in the collection that contains term t.

[4] Some of these features are computed before the process of text cleaning and normalization is performed.

Through experimentation conducted on the training set, the vocabulary was built by considering unigrams with document frequency greater than eight. Unigrams that occurred in at least 90% of documents in the collection were ignored.

- Word2vec: in this approach, the vectors used to represent words are called *word embeddings*. Unlike representations such as tf-idf, where vectors have a high dimensionality and are also characterized by sparsity, word2vec produces dense vectors of much lower dimensionality that capture semantic relations between words [51].

  To learn vector representation of words, a model from the Spanish Wikipedia corpus was generated by using the Gensim [66] implementation of word2vec. Thus, each word is represented as a 480-dimensional vector under the *skip-gram* architecture. Once the representations of the words occurring in tweet t are induced, this is modeled as the average of its word vectors.

As result of applying the previous approaches, two feature vectors of fixed length for each tweet are generated: the first one corresponds to concatenate the set of basic features with the vectorization obtained by the tf-idf scheme; the second one is a 504-dimensional vector where the first 24 features represent the basic features, and the remaining to the average of word embeddings.

5.2.1.3 Machine Learning Classification

At the last stage, the sentiment analysis system classifies a given tweet as either positive, negative, or neutral. It is important to recall that the system deals with a multiclass classification task, and therefore it assigns only one label to the tweet. After receiving as input the feature vectors individually, two L2-regularized Logistic Regression classifiers (one for each feature vector) produce probabilities for each class, instead of predicting a class label. The classifiers were trained on the training set via cross validation, using the Scikit-learn [58] implementation of the Logistic Regression algorithm.

Taking into consideration the probability estimates, the final classification of a tweet may be made either by determining which is the best feature vector (i.e., the one that produces more accurate outputs), or by combining both outputs. For the combination, the approach proposed by Hagen et al. [31] is applied as follows: the probabilities of each class are averaged, and then the one with the highest average probability is chosen.

5.2.2 Experiments

Firstly, the three classification settings proposed above were evaluated. The first one, called *tfidf*, corresponds to a classifier that receives as input the feature vector formed by concatenating the set of basic features and the vectorization obtained by the tf-idf weighting scheme. The second one, called *word2vec*, corresponds to a classifier that is fed by a 504-dimensional vector where the first 24 dimensions represent the basic features, and the remaining correspond to the vector obtained by averaging word embeddings. The last one, called *ensemble*, receives the outputs (i.e., probability estimates for each class) of the two previous settings and combines them as it is described in [31]. In the first two settings, the class with the highest probability is chosen. Table 5.3 shows the results of evaluating the classification settings on the test set. From these results it was determined

TABLE 5.3. Performance of the classification settings on the test set

Classification setting	Accuracy	Macro-averaged F1-score
tfidf	0.6019	0.5824
word2vec	0.5680	0.5363
ensemble	0.5922	0.5715

TABLE 5.4. Discriminative power of the system for each class

Class	Precision	Recall	F1-score
Positive	0.65	0.43	0.52
Negative	0.62	0.74	0.67
Neutral	0.56	0.55	0.55

that the best setting to classify a tweet as either positive, negative, or neutral is the *tfidf*. In this way, the final system implementation was based on the *tfidf* classification setting.

Secondly, it was evaluated the discriminative power of the system for each class. The standard information retrieval metrics of precision, recall, and F1-score have been used to perform the evaluation. Table 5.4 shows the results.

Lastly, it is hypothesized that the low performance of the system is due to the process of creating the sentiment labeled dataset. Therefore, it is proposed as future work to train and evaluate a system on the general corpus provided by the organizing committee of the TASS workshop [90] in order to evaluate the previous hypothesis. As a result, it might be determined that a labeling process where a small number of volunteers participate, and it is also assisted by a classifier that suggests the polarity label of a tweet [90], produces a subjectivity easier to learn by the system, instead of a process where a large number of volunteers are involved, thus producing a subjectivity from different tendencies that is harder to learn.

5.3 Voting Intention Inference in the Colombian Election

In this section, the voting intention inference from Twitter data is presented. First, the independent variables used by the inference method, as well as the output variable it produces are listed. The voting intention inference was approached as a multiple linear regression analysis, such that several regression models were built to infer the voting share of each candidate in the first round and the run-off of the Colombian election. Lastly, the voting inference in the two electoral rounds are presented.

5.3.1 Features and Method

A common denominator in the literature on voting intention inference from Twitter data is either treat the proportion of tweets mentioning an electoral option as the reflection of its voting share [81], or employ the simplest of sentiment analysis methods (e.g., a lexicon-based classifier) and assume that the candidate with the highest sentiment score would result to be the chosen [50]. However, both inference methods have proven to be inconsistent [28]. Therefore, the feature selection has followed recommendations to deal with these problems [50, 28].

5.3.1.1 Features

The following features, which correspond to the independent variables used by the inference method, are computed from the COpres14 set in a daily basis per candidate to correlate them with the polling data. Note that spammer accounts and their tweets were removed from the COpres14 set.

1. Tweet volume: the number of tweets mentioning candidate c on day d.
2. Unique tweet volume: the number of tweets that only mentions candidate c on day d.
3. Twitter user count: the number of different Twitter users with at least one tweet mentioning candidate c on day d.
4. Unique Twitter user count: the number of different Twitter users whose tweets only mention candidate c on day d.
5. Positive (negative) tweet volume: the number of positive (negative) tweets that mentions candidate c on day d.
6. Positive- (negative-) based Twitter user count: the number of different Twitter users with at least one positive (negative) tweet mentioning candidate c on day d.

In total, 14 features have been proposed by additionally taking into account the *sentiment score* [55] and other ratios such as *tweets per user*. Tweets were classified by the sentiment analysis system to compute the sentiment-based features. Finally, the features are normalized by applying the moving average smoothing technique over a window of the past seven days, as it was proposed in [55].

5.3.1.2 Inference Method

The voting intention inference was approached as a multiple linear regression analysis. In this way, several regression models were built to infer the vote of each candidate in the two electoral rounds, using the aggregated polling data as the output variable of the models. In total, nine models were built, six of which were used to infer the voting in the first round election (one model per candidate, including the *blank vote* option).

In order to choose the best setting of each model, the first 80% observations were used as the training set, and the remaining as the test set. Under the proposed method, a regression model built to infer the voting of candidate c in either the first round or the run-off of the election, receives the feature vector computed for candidate c on day d and produces the voting share candidate c would receive if the election was held on day d. The Scikit-learn [58] implementations of the Ordinary Least Square, Ridge Regression, Lasso, and Support Vector Regression were used to train the different model settings via cross validation on the training set. Based on the performance of the settings on the test set, in terms of *mean absolute error*, the best one was chosen.

TABLE 5.5. Results and voting inferences per method in the first round election. Numbers in bold show the inference method with the lowest absolute error that correctly ranked a candidate

Candidate	Result	Polls	Twitter volume	Proposed method
Zuluaga	29.28%	27.53%	24.10%	**29.21%**
Santos	25.72%	28.99%	35.12%	**28.34%**
Ramírez	15.52%	9.43%	8.99%	9.23%
López	15.21%	10.56%	12.09%	10.15%
Peñalosa	8.27%	11.25%	8.13%	11.54%
Blank vote	5.98%	12.24%	11.65%	12.87%

TABLE 5.6. Results and voting inferences per method in the run-off election. Numbers in bold show the inference method with the lowest absolute error that correctly ranked a candidate

Candidate	Result	Polls	Twitter volume	Proposed method
Santos	50.98%	44.97%	**52.37%**	43.25%
Zuluaga	44.98%	**43.74%**	32.27%	46.29%
Blank vote	4.02%	**11.29%**	15.36%	12.94%

5.3.2 Results

Tables 5.5 and 5.6 show the official results and the voting intention inference in the first round and the run-off of the Colombian election, respectively. The *Result* column contains the official results of the election. Instead, the *Polls* and *Twitter volume* columns show the inferences from two baseline methods: the first one corresponds to the aggregated poll reports and the second one is based on Twitter volume [81]. The results of the proposed method are shown in the last column. In accordance to the Colombian law [15], polls must be conducted or published until eight days before the presidential election, which is why the results of Twitter volume and the proposed method for the ninth days before the election dates were used as the final inferences.

The inference results of the proposed method were good enough in the first round election, with a mean absolute error (MAE) of 4 percentage points (the lowest one) and the highest-polling candidates correctly ranked. However, the results of the proposed method in the run-off election were worse than those of the baseline methods, being the poll-based inference the one that correctly ranked all the candidates with the lowest MAE (4.84%). At last, although the Twitter volume method obtained the highest MAE in both rounds of the election, it correctly ranked the contenders in the run-off.

5.4 Summary

In this chapter, a sentiment analysis system of Spanish tweets was presented. The system was trained on a dataset whose domain is the Colombian election. However, it is hypothesized that the low performance that the system achieved was due to the process of creating the dataset. In order to assign a class label to a tweet, the machine learning classifier receives a feature vector formed by concatenating a set of basic features and the

vectorization obtained by the tf-idf weighting scheme. Before making the classification, a process of text cleaning and normalization is performed in two phases.

In addition to the sentiment analysis system, the voting intention inference in the Colombian election was presented. Under the proposed inference method, several regression models were built to infer the votes of each candidate in the first round and the run-off of the election, using data aggregated from opinion polls to train and evaluate the models. The results of the proposed method were good enough in the first round election; however, the inference results for the run-off election were worse than those of the baseline methods.

CHAPTER 6

Conclusions and Future Work

6.1 Conclusions

- Social media analysis represents a prolific research trend that demands a cautious handling. Its potential partly depends on the acknowledgment of its particularities and of the appropriate selection of the data it provides.
- In the same way that social media provides a rich source of information, it also contains noisy, useless, and irrelevant information in the form of spam, rumors, misinformation, political astroturf, slander, or simply noisy messages.
- Therefore, spam removal becomes one of the major issues in the search for more reliable measurements from social media data. For example, it was found that 22% of users tweeting about the Colombian election were spammers, whom generated 15% of the tweets in the retrieved collection. These findings prove that the mere proportion of tweets mentioning an electoral option cannot be considered as a plausible reflection of its voting share.
- The results obtained in this research support that Twitter user classification based on supervised learning is the most appropriate approach to deal with the spammer detection problem. Likewise, results demonstrate that the Random Forest is the best technique to distinguish spammer accounts from non-spammer ones.
- Although the problem was also tackled by following a semi-supervised learning approach, the results were worse than those of the supervised learning approach in terms of true positive (spammers classified as spammers) and false positive (non-spammers misclassified as spammers) rates.
- A collection of Twitter users was semi-automatically classified into spammer and non-spammer by applying common methods to create a ground truth for spammer detection. The most helpful method was based on manual labeling, which was a pretty time consuming task. At this point, it is important to note that Twitter was able to detect harmful links and correctly suspend accounts that fell into some of the prohibited behaviors.

- Because non-standard word forms abound in Twitter data, a lexical normalization system of Spanish tweets was developed to normalize out-of-vocabulary (OOV) words to their canonical form, using finite-state transducers and statistical language modeling. The system correctly detected OOV words in tweets, and suggested most of the cases the proper corrections. However, there is a great room for improvement in the candidate selection component, because the system was not properly adapted to the informal genre and the free writing style of Twitter.

- A sentiment analysis system of Spanish tweets was developed by implementing a supervised classification approach. In order to assign a label class to a tweet, several classification settings were evaluated, including one based on ensemble combination. From their performance on the test set, it was determined that the final decision of a single classifier was the most appropriate setting to deal with sentiment classification, using the Logistic Regression algorithm for the machine learning classification. Despite implementing state-of-the-art approaches, the system achieved a low performance probably due to the process of creating the sentiment labeled dataset used to train it. In view of this, it is hypothesized that the process produced a subjectivity from different tendencies that was not properly learned by the system, since a large number of volunteers were involved.

- The feature selection was performed by taking into account the features commonly used in the literature. In this stage, two vectorization methods were evaluated: *tf-idf* and *word2vec*. The former was the method that best supported the sentiment classification, even though the latter captured semantic relations between words and produced dense vectors of much lower dimensionality.

- In order to investigate the potential of social media analysis to infer voting intention, the sentiment analysis system was applied on the collection of tweets referring to the Colombian election. Despite obtaining results good enough in the first round election, with the lowest mean absolute error and correctly ranking the highest-polling candidates, such an important method cannot be put forward as a substitute of what professional pollsters have been doing for the last years.

6.2 Future Work

The following directions of future work are proposed to further develop this thesis:

- The Twitter user classification system achieved a high spammer detection rate and its overall accuracy was competitive to the state-of-the-art. Further research should focus on designing new features that lead to increase the spammer detection rate while keeping a lower false positive rate, using no more resources that those used in the research.

- The lexical normalization system of Spanish tweets has a great room for improvement in its candidate selection component, because the language used in the Spanish Wikipedia corpus, from which the statistical language model was estimated, is characterized by a more formal style than that used in Twitter. For this reason, it is planned to build a large Twitter corpus from users who, in theory, write correctly, e.g., journalists and mass media. In this way, the system could be adapted to the informal genre and free writing style of Twitter.

- It is hypothesized that the low performance of the sentiment analysis system was due to the process of creating the dataset used for the training. Therefore, it is proposed to train and evaluate a system on the general corpus provided by the organizing committee of the TASS workshop, which is a benchmark for sentiment analysis of Spanish tweets, in order to evaluate the previous hypothesis.

- The sentiment analysis system should learn to deal with figurative language such as irony or sarcasm, and thus understand when an apparently positive language is used to convey negative meanings. On the other hand, because the application of the system in the Colombian election was based on the aggregation of the opinion orientation, and not on whether a specific user had a positive or negative view of a candidate, sentiment analysis could be tackled as a quantification problem instead of a classification problem.

Bibliography

[1] Alicia Ageno, Pere R. Comas, Lluís Padró, and Jordi Turmo, *The TALP-UPC approach to Tweet-Norm 2013*, Proceedings of the Tweet Normalization Workshop at SEPLN 2013, September 2013.

[2] Inaki Alegria, Nora Aranberri, Pere R. Comas, Víctor Fresno, Pablo Gamallo, Lluís Padró, Inaki San Vicente, Jordi Turmo, and Arkaitz Zubiaga, *Tweetnorm: a benchmark for lexical normalization of spanish tweets*, Language Resources and Evaluation **49** (2015), no. 4, 883–905.

[3] Silvio Amir, Miguel Almeida, Bruno Martins, Joao Filgueiras, and Mário J. Silva, *Tugas: Exploiting unlabelled data for twitter sentiment analysis*, Proceedings of the eighth international workshop on Semantic Evaluation (SemEval 2014) (Dublin, Ireland), August 2014.

[4] Amit A. Amleshwaram, Narasimha Reddy, Sandeep Yadav, Guofei Gu, and Chao Yang, *Cats: Characterizing automation of twitter spammers*, 2013 IEEE Fifth International Conference on Communication Systems and Networks (COMSNETS), 2013.

[5] Natalia Arteaga and Esteban Guerra, *Las encuestadoras se lavan las manos en la recta final de la campaña presidencial*, `http://www.webcitation.org/6g3p9G1lE`, 2014, (accessed: December 15, 2015).

[6] Kenneth R. Beesley and Lauri Karttunen, *A Gentle Introduction*, Finite State Morphology, Center for the Study of Language and Information, April 2003.

[7] Fabrício Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgílio Almeida, *Detecting spammers on twitter*, Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS), 2010.

[8] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete, *Information credibility on twitter*, Proceedings of the 20th International Conference on World Wide Web (New York, NY, USA), ACM, 2011, pp. 675–684.

[9] ———, *Predicting information credibility in time-sensitive social media*, Internet Research **23** (2013), no. 5, 560–588.

[10] Chih-Chung Chang and Chih-Jen Lin, *LIBSVM: A library for support vector machines*, ACM Transactions on Intelligent Systems and Technology **2** (2011), 1–27.

[11] Olivier Chapelle, Bernhard Schlkopf, and Alexander Zien, *Semi-supervised learning*, 1st ed., The MIT Press, 2010.

[12] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia, *Detecting Automation of Twitter Accounts: Are You a Human, Bot, or Cyborg?*, IEEE Trans. Dependable Secur. Comput. **9** (2012), no. 6, 811–824.

[13] Cifras y Conceptos, *Polimétrica*, `http://www.webcitation.org/6g3oHiI58`, (accessed: December 15, 2015).

[14] Colprensa, *Resultados de la gran encuesta de los medios de junio de 2014*, `http://www.webcitation.org/6g3pXRoUX`, 2014, (accessed: December 15, 2015).

[15] Congreso de la República de Colombia, *Ley 996 de 2005*, `http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=18232`, 2005, (accessed: January 25, 2016).

[16] Juan M. Cotelo, Fermín L. Cruz, Jose Antonio Troyano, and F. Javier Ortega, *A modular approach for lexical normalization applied to Spanish tweets*, Expert Systems with Applications **42** (2015), no. 10, 4743–4754.

[17] M.C. Díaz-Galiano and A. Montejo-Ráez, *Participación de sinai dw2vec en tass 2015*, Proceedings of the Sentiment Analysis Workshop at SEPLN (TASS2015), September 2015, pp. 59–64.

[18] El Espectador, *"Uribe dice que J.J. Rendón entregó US$2 millones a campaña de Santos en 2010"*, `http://www.webcitation.org/6gLcDXDd2`, 2014, (accessed: May 23, 2015).

[19] El Tiempo, *Zuluaga ganaría en primera y segunda vuelta, dice encuesta del cnc*, `http://www.webcitation.org/6g3nwoz3z`, (accessed: December 15, 2015).

[20] ———, *"'Hacker' dijo a Zuluaga que tenía acceso a información de inteligencia"*, `http://www.webcitation.org/6gLcs40fC`, 2014, (accessed: May 23, 2015).

[21] ———, *Leve ventaja de santos en carrera con zuluaga*, `http://www.webcitation.org/6g3otoW9E`, 2014, (accessed: December 15, 2015).

[22] ———, *"Óscar Iván Zuluaga reconoce que hacker capturado trabaja en su campaña"*, `http://www.webcitation.org/6gLcPHtJJ`, 2014, (accessed: May 23, 2015).

[23] ———, *Óscar Iván Zuluaga, 47 %; Juan Manuel Santos, 45 %*, `http://www.webcitation.org/6g3p1tS1z`, 2014, (accessed: December 15, 2015).

[24] ———, *Santos y zuluaga disputarían la presidencia en una segunda vuelta*, `http://www.webcitation.org/6g3oqbXc2`, 2014, (accessed: December 15, 2015).

[25] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin, *LIBLINEAR: A library for large linear classification*, Journal of Machine Learning Research **9** (2008), 1871–1874.

[26] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini, *The rise of social bots*, CoRR **abs/1407.5225** (2014).

[27] Pablo Gamallo, Marcos García, and José R. Pichel, *A Method to Lexical Normalisation of Tweets*, Proceedings of the Tweet Normalization Workshop at SEPLN 2013, September 2013.

[28] Daniel Gayo-Avello, *A meta-analysis of state-of-the-art electoral prediction from twitter data*, Soc. Sci. Comput. Rev. **31** (2013), no. 6, 649–679.

[29] Google, *Safe Browsing API*, `https://developers.google.com/safe-browsing/`, 2015, (accessed: March 7, 2015).

[30] Chris Grier, Kurt Thomas, Vern Paxson, and Michael Zhang, *@Spam: The Underground on 140 Characters or Less*, Proceedings of the 17th ACM Conference on Computer and Communications Security (New York, NY, USA), CCS '10, ACM, 2010, pp. 27–37.

[31] Matthias Hagen, Martin Potthast, Michel Büchner, and Benno Stein, *Webis: an ensemble for twitter sentiment detection*, Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Association for Computational Linguistics, 2015, pp. 582–589.

[32] Bo Han and Timothy Baldwin, *Lexical Normalisation of Short Text Messages: Makn Sens a #Twitter*, Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (Stroudsburg, PA, USA), HLT '11, Association for Computational Linguistics, 2011, pp. 368–378.

[33] Bo Han, Paul Cook, and Timothy Baldwin, *Automatically constructing a normalisation dictionary for microblogs*, Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL '12, 2012, pp. 421–432.

[34] Kenneth Heafield, *KenLM: faster and smaller language model queries*, Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation (Edinburgh, Scotland, United Kingdom), July 2011, pp. 187–197.

[35] Mark Huberty, *Can we vote with our tweet? on the perennial difficulty of election forecasting with social media*, International Journal of Forecasting **31** (2015), no. 3, 992–1007.

[36] Mans Hulden, *Foma: a finite-state compiler and library*, Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2009, pp. 29–32.

[37] Lluís-F. Hurtado, Ferran Pla, and Davide Buscaldi, *Elirf-upv en tass 2015: Análisis de sentimientos en twitter*, Proceedings of the Sentiment Analysis Workshop at SEPLN (TASS2015), September 2015, pp. 75–79.

[38] Infométrika, *Encuesta de intención de voto elecciones presidenciales colombia 2014*, `http://www.webcitation.org/6g3mVY6pk`, 2014, (accessed: December 15, 2015).

[39] Internet World Stats, *Internet world users by language – top 10 languages*, `http://www.internetworldstats.com/stats7.htm`, 2015, (accessed: February 01, 2016).

[40] Andreas Jungherr, Pascal Jürgens, and Harald Schoen, *Why the pirate party won the german election of 2009 or the trouble with predictions: A response to tumasjan, a., sprenger, t. o., sander, p. g., & welpe, i. m. "predicting elections with twitter: What 140 characters reveal about political sentiment"*, Soc. Sci. Comput. Rev. **30** (2012), no. 2, 229–234.

[41] George Karypis, Eui-Hong (Sam) Han, and Vipin Kumar, *Chameleon: Hierarchical clustering using dynamic modeling*, Computer **32** (1999), no. 8, 68–75.

[42] La FM, *Última Gran Encuesta: Óscar Iván Zuluaga 29.5 %, Juan M. Santos 28.5 %*, `http://www.webcitation.org/6g3oYtJ2k`, 2014, (accessed: December 15, 2015).

[43] ———, *Última Gran Encuesta: Óscar Iván Zuluaga (49%) y Juan M. Santos (41%)*, `http://www.webcitation.org/6g3oiQRtJ`, 2014, (accessed: December 15, 2015).

[44] Quoc V. Le and Tomas Mikolov, *Distributed representations of sentences and documents*, Proceedings of the 31th International Conference on Machine Learning, ICML 2014, June 2014, pp. 1188–1196.

[45] Kyumin Lee, Brian David Eoff, and James Caverlee, *Seven months with the devils: a long-term study of content polluters on twitter*, AAAI Int'l Conference on Weblogs and Social Media (ICWSM), 2011.

[46] Rui Li, Kin Hou Lei, Ravi Khadiwala, and Kevin Chen-Chuan Chang, *Tedas: A twitter-based event detection and analysis system*, Data Engineering (ICDE), 2012 IEEE 28th International Conference on, April 2012, pp. 1273–1276.

[47] Bing Liu and Lei Zhang, *A survey of opinion mining and sentiment analysis*, Mining Text Data (Charu C. Aggarwal and ChengXiang Zhai, eds.), Springer US, 2012, pp. 415–463.

[48] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze, *Scoring, term weighting and the vector space model*, An Introduction to Information Retrieval, Cambridge University Press, New York, NY, USA, 2008.

[49] M. McCord and M. Chuah, *Spam detection on twitter using traditional classifiers*, Autonomic and Trusted Computing, Lecture Notes in Computer Science, vol. 6906, Springer Berlin Heidelberg, 2011, pp. 175–186.

[50] Panagiotis Takis Metaxas, Eni Mustafaraj, and Daniel Gayo-Avello, *How (not) to predict elections*, 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), Oct 2011, pp. 165–171.

[51] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, *Efficient estimation of word representations in vector space*, CoRR **abs/1301.3781** (2013).

[52] Zachary Miller, Brian Dickinson, William Deitrick, Wei Hu, and Alex Hai Wang, *Twitter spammer detection using data stream clustering*, Information Sciences **260** (2014), 64–73.

[53] Yasuhide Miura, Shigeyuki Sakaki, Keigo Hattori, and Tomoko Ohkuma, *Teamx: A sentiment analyzer with enhanced lexicon mapping and weighting scheme for unbalanced data*, Proceedings of the eighth international workshop on Semantic Evaluation (SemEval 2014) (Dublin, Ireland), August 2014.

[54] Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu, *Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets*, Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013) (Atlanta, Georgia, USA), June 2013.

[55] Brendan O'Connor, Ramnath Balasubramanyan, Bryan Routledge, and Noah Smith, *From tweets to polls: Linking text sentiment to public opinion time series*, International AAAI Conference on Weblogs and Social Media, 2010.

[56] Lluís Padró and Evgeny Stanilovsky, *FreeLing 3.0: Towards Wider Multilinguality*, Proceedings of the Language Resources and Evaluation Conference (LREC 2012) (Istanbul, Turkey), ELRA, May 2012.

[57] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan, *Thumbs up?: Sentiment classification using machine learning techniques*, Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10, EMNLP '02, Association for Computational Linguistics, 2002, pp. 79–86.

[58] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, *Scikit-learn: Machine learning in Python*, Journal of Machine Learning Research **12** (2011), 2825–2830.

[59] PhishTank, *PhishTank – Join the fight against phishing*, `https://www.phishtank.com/`, (accessed: March 7, 2015).

[60] Jordi Porta and José L. Sancho, *Word normalization in Twitter using finite-state transducers*, Proceedings of the Tweet Normalization Workshop at SEPLN 2013, September 2013.

[61] RAE, *Exclusión de ch y ll del abecedario*, `http://www.webcitation.org/6gLderGSk`, (accessed: October 16, 2015).

[62] ———, *Mayúculas*, `http://buscon.rae.es/dpd/srv/search?id=BapzSnotjD6n0vZiTp`, 2005, (accessed: October 15, 2015).

[63] ———, *Seseo*, `http://lema.rae.es/dpd/srv/search?id=IIUwJDU07D6XC2xEky`, 2005, (accessed: November 9, 2015).

[64] ———, *Voseo*, `http://lema.rae.es/dpd/srv/search?id=iOTUSehtID6mVONyGX`, 2005, (accessed: October 24, 2015).

[65] ———, *Yeísmo*, `http://lema.rae.es/dpd/srv/search?id=HK5DEyboyD6iOqnxZu`, 2005, (accessed: October 23, 2015).

[66] Radim Řehůřek and Petr Sojka, *Software framework for topic modelling with large corpora*, Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, ELRA, 2010, pp. 45–50.

[67] Xabier Saralegi and Inaki San Vicente, *Elhuyar at tass 2013*, Proceedings of the Sentiment Analysis Workshop at SEPLN (TASS2013), September 2013.

[68] ———, *Elhuyar at TweetNorm 2013*, Proceedings of the Tweet Normalization Workshop at SEPLN 2013, September 2013.

[69] Harald Schoen, Daniel Gayo-Avello, Panagiotis Takis Metaxas, Eni Mustafaraj, Markus Strohmaier, and Peter Gloor, *The power of prediction with social media*, Internet Research **23** (2013), no. 5, 528–543.

[70] Ashwin Seshagiri, *The languages of twitter users*, http://www.webcitation.org/6gLdPszkH, 2014, (accessed: December 4, 2015).

[71] Lei Shi, Neeraj Agarwal, Ankur Agrawal, Rahul Garg, and Jacob Spoelstra, *Predicting us primary elections with twitter*, 2012.

[72] Grigori Sidorov, Sabino Miranda-Jiménez, Francisco Viveros-Jiménez, Alexander Gelbukh, Noé Castro-Sánchez, Francisco Velásquez, Ismael Díaz-Rangel, Sergio Suárez-Guerra, Alejandro Trevino, and Juan Gordon, *Empirical study of machine learning based approach for opinion mining in tweets*, Advances in Artificial Intelligence (Ildar Batyrshin and Miguel González Mendoza, eds.), Lecture Notes in Computer Science, vol. 7629, Springer Berlin Heidelberg, 2013, pp. 1–14.

[73] Jonghyuk Song, Sangho Lee, and Jong Kim, *Spam filtering in twitter using sender-receiver relationship*, Recent Advances in Intrusion Detection (Robin Sommer, Davide Balzarotti, and Gregor Maier, eds.), Lecture Notes in Computer Science, vol. 6961, Springer Berlin Heidelberg, 2011, pp. 301–317.

[74] Spamhaus, *The Spamhaus project*, http://www.spamhaus.org/, (accessed: March 7, 2015).

[75] Jane Stecyk, *Study: Twitter users love mobile apps*, http://www.webcitation.org/6g3pmjP9k, 2015, (accessed: November 10, 2015).

[76] SURBL, *SURBL*, http://www.surbl.org/, (accessed: March 7, 2015).

[77] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, *Cluster analysis: Basic concepts and algorithms*, Introduction to Data Mining, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.

[78] Kurt Thomas, Chris Grier, Dawn Song, and Vern Paxson, *Suspended Accounts in Retrospect: An Analysis of Twitter Spam*, Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference (New York, NY, USA), IMC '11, ACM, 2011, pp. 243–258.

[79] Nitasha Tiku and Casey Newton, *Twitter CEO: 'We suck at dealing with abuse'*, http://www.webcitation.org/6gLe0isPw, 2015, (accessed: June 13, 2015).

[80] Adam Tsakalidis, Symeon Papadopoulos, Alexandra I. Cristea, and Yiannis Kompatsiaris, *Predicting elections for multiple countries using twitter and polls*, Intelligent Systems, IEEE **30** (2015), no. 2, 10–17.

[81] Andranik Tumasjan, Timm Sprenger, Philipp Sandner, and Isabell Welpe, *Predicting elections with twitter: What 140 characters reveal about political sentiment*, International AAAI Conference on Weblogs and Social Media, 2010.

[82] Peter D. Turney, *Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews*, Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, ACL '02, Association for Computational Linguistics, 2002, pp. 417–424.

[83] Twitter, *My account is suspended*, https://support.twitter.com/articles/15790, (accessed: March 17, 2015).

[84] ———, *Reporting spam on Twitter*, https://support.twitter.com/articles/64986-reporting-spam-on-twitter, (accessed: March 16, 2015).

[85] ———, *REST APIs*, https://dev.twitter.com/rest/public, (accessed: March 6, 2015).

[86] ———, *The search API*, https://dev.twitter.com/rest/public/search, (accessed: March 6, 2015).

[87] ———, *The Twitter Rules*, https://support.twitter.com/articles/18311-the-twitter-rules, (accessed: March 16, 2015).

[88] URIBL, *URIBL.COM - Realtime URI Blacklist*, http://uribl.com/, (accessed: March 7, 2015).

[89] David Vilares, Miguel A Alonso, and Carlos Gómez-Rodrıguez, *On the usefulness of lexical and syntactic processing in polarity classification of twitter messages*, Journal of the Association for Information Science and Technology (2014).

[90] Julio Villena-Román, Janine García-Morera, Sara Lana-Serrano, and José Carlos González-Cristóbal, *Tass 2013 - a second step in reputation analysis in spanish*, Procesamiento del Lenguaje Natural **52** (2014), no. 0, 37–44.

[91] Alex Hai Wang, *Don't Follow Me - Spam Detection in Twitter*, SECRYPT (Sokratis K Katsikas and Pierangela Samarati, eds.), SciTePress, 2010, pp. 142–151.

[92] Chao Yang, Robert Chandler Harkreader, and Guofei Gu, *Die free or live hard? empirical evaluation and new design for fighting evolving twitter spammers*, Recent Advances in Intrusion Detection, Lecture Notes in Computer Science, vol. 6961, Springer Berlin Heidelberg, 2011, pp. 318–337.

[93] Ramón Zacarías, *Formación de diminutivos con el sufijo /-ít-/. Una propuesta desde la morfología natural*, Anuario de Letras: Lingüística y Filología **44** (2006), 77–103.