Empirical Study of Machine Learning Based Approach for Opinion Mining in Tweets

Grigori Sidorov[1], Sabino Miranda-Jiménez[1], Francisco Viveros-Jiménez[1],
Alexander Gelbukh[1], Noé Castro-Sánchez[1], Francisco Velásquez[1],
Ismael Díaz-Rangel[1], Sergio Suárez-Guerra[1],
Alejandro Treviño[2], and Juan Gordon[2]

[1] Center for Computing Research,
Instituto Politécnico Nacional,
Av. Juan de Dios Bátiz, s/n, esq. Mendizabal,
Col. Nueva Industrial Vallejo, 07738, Mexico City, Mexico
[2] Intellego SC, Mexico City, Mexico
www.cic.ipn.mx/~sidorov, sabino_m@hotmail.com

Abstract. Opinion mining deals with determining of the sentiment orientation—positive, negative, or neutral—of a (short) text. Recently, it has attracted great interest both in academia and in industry due to its useful potential applications. One of the most promising applications is analysis of opinions in social networks. In this paper, we examine how classifiers work while doing opinion mining over Spanish Twitter data. We explore how different settings (n-gram size, corpus size, number of sentiment classes, balanced vs. unbalanced corpus, various domains) affect precision of the machine learning algorithms. We experimented with Naïve Bayes, Decision Tree, and Support Vector Machines. We describe also language specific preprocessing—in our case, for Spanish language—of tweets. The paper presents best settings of parameters for practical applications of opinion mining in Spanish Twitter. We also present a novel resource for analysis of emotions in texts: a dictionary marked with probabilities to express one of the six basic emotions—Probability Factor of Affective use (PFA)—Spanish Emotion Lexicon that contains 2,036 words.

Keywords. Opinion mining, sentiment analysis, sentiment classification, Spanish Twitter corpus, Spanish Emotion Lexicon.

1 Introduction

Opinion mining (or sentiment analysis[1]) has attracted great interest in recent years, both in academia and industry due to its potential applications. One of the most

[1] The terms "opinion mining" and "sentiment analysis" usually are used to denote essentially the same phenomenon, thus, they can be considered synonyms. It should be mentioned, though, that if we say "opinion", we can refer to much broader sense, appealing to

promising applications is analysis of opinions in social networks. Lots of people write their opinions in forums, microblogging or review websites. This data is very useful for business companies, governments, and individuals, who want to track automatically attitudes and feelings in those sites. Namely, there is a lot of data available that contains much useful information, so it can be analyzed automatically. For instance, a customer who wants to buy a product usually searches the Web trying to find opinions of other customers or reviewers about this product. In fact, these kinds of reviews affect customer's decision.

Opinion mining in a broad sense is defined as the computational study of opinions, sentiments and emotions expressed in texts [1]. Opinions exist on the Web for any entity or object (person, product, service, etc.), and for the features or components of these objects, like, a cell phone battery, keyboard, touch screen display, etc.

Detecting sentiments is considered a difficult task. Say, in the example '*la aplicación responde muy rápido* (*the application responds very fast*)'; the sentiment of the opinion is positive, because the word '*rápido* (*fast*)' implies a good thing—it is good that applications run fast. However, the same word in other context, like in the sentence '*la batería se descargó muy rápido* (*the battery discharged very fast*)', implies a negative sentiment—it is bad that batteries reduce their power quickly. So, the problem implies using of world knowledge, which is very vast and complex problem.

Formally, we say that an **opinion** of a feature ***f*** has a **sentiment** attached, commonly positive or negative. The person who emits the opinion is known as **opinion holder**. Thus, an opinion is defined as a quintuple (o_j, f_{jk}, oo_{ijkl},h_i, t_l) [2], where:

- o_j is the object of the opinion.
- f_{jk} is a feature of the object o_j about which the opinion is expressed. When no feature is detected, we use "**general opinion**" as the object feature.
- oo_{ijkl} is the sentiment polarity of the opinion about the feature f_{jk} of the object o_j—positive, negative, neutral.
- h_i is the opinion holder.
- t_l is the time when the opinion is expressed by h_i.

For our work we use messages posted in Spanish Twitter. In this work, the opinion quintuple matches a message as follows:

- o_j is the entity the tweet deals with. A tweet contains one or more entities. Entities are sets of synonyms defined by a user.
- f_{jk}, feature is ignored for the moment, i.e., general opinion is used as the object feature.
- oo_{ijkl} is the message global polarity: positive, negative, neutral, or informative (news).
- h_i is the user who posted the message.
- t_l is the time when the message was posted.

substantial characteristics of our object, like, for example, size of a product, its weight, etc. While saying "sentiment", we mean only positive or negative feelings. If we would like to analyze more detailed feelings, we would say "emotion analysis/mining".

In the following message '*@user: Mi iPhone se calienta mucho (@user: My iPhone gets overheated*)'. The object (o) is *iPhone*; the feature (f) is related to the temperature, but in this work we will not try to detect it; the assigned polarity (oo) is negative, because it is bad that a cellphone gets overheated; the holder (h) is @user; and time (t) is the Twitter publication time. We use this formalism because it suits our domain (Twitter, see section 2.1). In case of Twitter, we use short text in contrast to reviews that are longer texts [3].

Many systems and approaches have been implemented for detecting sentiments in texts [1, 4]. We can distinguish two main methodologies used in opinion mining: machine learning approaches and the so-called symbolic approaches—approaches that use manually crafted rules and lexicons [5, 6]. This paper focuses on machine learning approaches.

Opinion mining task can be transformed into classification task, so machine learning techniques can be used for opinion mining. Machine learning approaches require a corpus containing a wide number of manually tagged examples, in our case, tweets with a sentiment assigned manually by a human annotator. In our corpus, text is represented as a set of features for classification. These features are traditional word n-grams extracted from each tweet in the corpus.

Let us explain briefly the n-gram representation for the sentence '*battery discharges very fast*'. When using n-gram features, an opinion is represented as independent n-grams of various orders: unigram (*battery, discharge, very, fast*), bigrams (combination of two words: *battery-discharge, discharge-very, very-fast*), trigrams (combination of three words: *battery-discharge-very, discharge-very-fast*), and so on. Note that we use morphologically normalized representation. When using POS n-grams, a POS-tag is used instead of each text word. For example, when using POS unigrams in the features set *battery* is changed to *Noun, discharge* is changed to *Verb, very* is changed to *Adverb, fast* is changed to *Adjective*, etc.

For the classification task, the most frequently used machine learning methods are: Naïve Bayes (NB), Maximum Entropy (MaxEnt), and Support Vector Machines (SVM). Machine learning approaches report relatively high results, i.e., Pang *et al*. [4] obtained 81.0% (NB), 80.4% (MaxEnt), and 82.9% (SVM). They used two classes: positive and negative, and worked using product reviews that are longer texts than tweets. In the domain of English Twitter, Go *et al*. [7] report similar results 81.3%, 80.5%, and 82.2% for the same classifiers. They use three classes: positive, negative, and neutral.

In this research, our aim is to find out what are the best settings of parameters for classifying of Spanish tweets. Our research questions are:

(1) Proper size for the training corpus,
(2) The best classifier for the task,
(3) Optimal size of n-grams,
(4) How using more classes—positive, negative, neutral, and informative (news)—affect~~s~~ precision,
(5) If balanced corpus improves precision (vs. unbalanced),
(6) How classifiers perform when the testing set and the training corpus belong to different domains, and
(7) The cases where the best classifier fails, i.e., the reason behind the errors.

There is little research on opinion mining in Spanish [8, 9], so we also describe specific preprocessing techniques for this language. This is the first research that uses Spanish Twitter corpus.

The paper is organized as follows. In section 2, we describe the used corpus, data preprocessing, and selected classifiers. In section 3, we present the results of the analysis for different settings of parameters. In section 4, the common errors are analyzed. In section 5, we describe novel resource for analysis of emotions on texts: a dictionary marked with probabilities to express one of the six basic emotions—Spanish Emotion Lexicon that contains 2,043 words. In section 6, conclusions are presented.

2 Opinion Mining Method

The general scheme of our processing is composed of several stages. First, a Spanish Twitter corpus is compiled (see section 2.1). After this, the data is modified in order to prepare the necessary information for classifiers (see section 2.2). Finally, the classifiers are trained with different settings as shown in section 2.3.

2.1 Corpus of Tweets

We chose to work with Spanish Twitter for our experiments. Twitter is a microblogging platform where users post their messages, opinions, comments, etc. Contents of the messages range from personal thoughts to public statements. A Twitter message is known as tweet. Tweets are very short; the maximum size of a tweet is 140 characters that usually correspond to a phrase. Thus, our work is limited to sentence level.

We use a global polarity rating due to shortness of messages in Twitter, and we do not process cases where tweets have more than one sentiment orientation. In addition, we do not extract features of products as in case of the reviews, neither we use predefined sets of characteristics [4]. In case of free-form texts (our case), it is not easy to determine object features, as, for example, in case of movie reviews, where the predefined sets of characteristics of films are used.

We compiled a corpus based on data extracted form Twitter. The corpus was built using a list of predefined entities about cell phone brands. We collected 32,000 tweets, and around 8,000 tweets were annotated by hand determining one of four classes for each tweet: positive (P), negative (N), neutral (T), or informative (news, I). Each class is described as follows:

1. Positive, if it has a positive sentiment in general, like in '*la aplicación responde muy rápido* (*the application responds very fast*)'.
2. Negative, if it has a negative sentiment in general, like in '*mi iPhone se calienta mucho* (*my iPhone gets overheated*)'.
3. Neutral, if it has no sentiment '*Estoy tuiteando desde el iPhone* (*I am tweeting from my iPhone*)'.
4. Informative (news), if it contains news or advertisements '*Vendo mi Samsung Galaxy, nuevo en caja* (*I sell my Samsung Galaxy, new in the box*)'.

Note that it is common to use just two classes—positive and negative. However, we used these four categories because one of our aims is to find out how the number of classes affects precision of classifiers.

2.2 Preprocessing

Analysis of tweets is complex task because these messages are full of slang, misspellings [7] and words borrowed from other languages. Some examples of errors are shown in Table 1. In order to tackle the problems mentioned in Table 1 and to deal with the noise in texts, we normalize the tweets before training the classifiers with the following procedures:

- Error correction,
- Special tags,
- POS-tagging,
- Negation processing.

Table 1. Common errors in Spanish tweets

Type of error	Example
(1) Slang	**(x fa/please)** *olvidé un celular en un Matiz, x fa que lo devuelvan* (*I forgot a cell phone in a Matiz, please give it back*)
(2) Misspelling	**(muerto~~oo~~/dead)** *tu celular estaba muertooo!* (*your cellphone was dead*!)
(3) Mixed languages	**(bonito/nice)** *ya está aquí, más* nice*, más rápido, el Nokia Lumia* (*It's here, It's very nice, It's faster, the Nokia Lumia*)

Error Correction

In case of orthographic errors like in (1) '*muertooo* (*dead*)', we use an approach based on a Spanish dictionary and a statistical model for common double letters in Spanish. Also, we developed a set of rules made by hand for slang and common words borrowed from the English language. The rules were made after manual analysis of the data from our corpus. We do not detect orthographic mistakes ('*dies*' instead of '*diez* (*ten*)'); '*sincronizacion*' instead of '*sincronización* (*synchronization*)') or split the words that are agglutinated ('*padrepero*' instead of '*padre pero*' (*nice but*)).

Special Tags

For usernames, hash tags, emoticons, and URLs in a tweet, we use an approach similar to [7]. We use special tags (USER_TAG, WINK_TAG, HASH_TAG, and URL_TAG) to replace the word by the corresponding tag, so that POS-tagger could tag correctly each word of the tweet. For instance, in the tweet '*@user no me arrepiento, soy feliz con iPhone :)* (*I have no regrets, I am happy with iPhone* :))', the user is identified by @ and the wink by the symbols :). List of common winks was compiled manually. The normalized tweet would be '*USER_TAG no me arrepiento, soy feliz con iPhone WINK_TAG*'.

POS-tagging

After text normalization, we applied a POS-tagger for Spanish using Freeling tool [10]. Freeling is a system for linguistic analysis of texts, like tagging, lemmatization, etc. After applying the POS-tagger, we obtain for each word its corresponding part of speech: verb, adjective, adverb, etc. Freeling follows the EAGLES recommendations for morphosyntactic tag set [13]. Also, we use the lemmatized words in order to reduce the number of word forms, which is important for morphologically rich Spanish language. For example, the tweet mentioned above is tagged as '*USER_TAG_NT000* (noun) *no_RN* (adverb) *me_PP1CS000* (pronoun) *arrepentir_VMIP1S0* (verb) *,_Fc* (punctuation) *ser_VSIP1S0* (verb) *feliz_AQ0CS0* (adjective) *con_SPS00* (preposition) *iPhone_NCMS000* (noun) *WINK_TAG_NT000* (noun)'.

Processing of Negation

Negation affects the value of an opinion. We use a similar approach as in [11] to handle negations. We search and remove the adverb 'no' from opinion, and attach the prefix 'no_' to next word (verb or adjective) to build one unit. For example, '*no_RN* (adverb) *tener_VMIP1S0* (verb) *uno_DI0MS0* (article) *iPhone_NCMS000* (noun) (*no tengo un iPhone* / I do not have an iPhone)' is transformed into '*no_tener_VMIP1S0 uno_DI0MS0 iPhone_NCMS000*'. Rules of transformation were made by hand according to the patterns detected in our corpus.

2.3 Selected Classifiers

Our method uses various machine learning classifiers. The machine learning classifiers we selected were: Naïve Bayes (NB), C4.5 (Decision Tree) and Support Vector Machines (SVM). NB and SVM were used in several experiments with good results for English language [4, 7].

We use WEKA API that implements all above mentioned algorithms [12]. WEKA implements SVM as SMO, and C4.5 as J48 algorithms.

Our input data are two sets of vectors. Each entry in the vector corresponds to a feature. We use the part of speech tags as filters for features. The part of speech tags that we consider as features are verbs, nouns, adjectives, adverbs, and interjections.

A set of 8,000 tweets were manually marked with one of the four categories mentioned above. We use 7,000 tweets as training set and 1,000 tweets as test set. The test set has 236 positive tweets, 145 negative, 342 neutral, and 257 news or advertisements.

3 Experiments and Evaluation

In this section, we describe the experiments that we carried out to determine the influence of corpus size, n-gram size, number of the classes, and balanced vs. unbalanced corpus on machine learning based sentiment classification. The models were trained using different sizes of n-grams. Let us remind that we consider the word and its POS tag together as features, for example, '*trabajar_verbo* (*work_verb*)' is one feature. We compute precision of the classifier on the whole evaluation dataset using equation 1.

$$precision = \frac{correct\ answers}{total\ answers} \qquad (1)$$

Our further analysis is based on the following experiments:

- Effect of the corpus size,
- Effect of the n-gram size,
- Effect of the number of the classes,
- Effect of an unbalanced corpus.

We conduct the experiments using the best setting obtained in the previous tests. All precision values of the following tables are given as percentage.

3.1 Effect of the Corpus Size

We tested different corpus sizes for training the three classifiers. In Table 2, we show how the corpus size affects precision. The precision was improved when using more training samples.

Table 2. Precision observed when using 12 different training corpus sizes

Classifier	**Part I. Corpus size (tweets)**						
	1,000	**1,500**	**2,000**	**2,500**	**3,000**	**3,500**	**4,000**
Naïve Bayes	42	46	42	42	45	44	45
J48	43	49	46	49	50	52	54
SVM	48	55	55	59	**61**	59	60

Classifier	**Part II. Corpus size (tweets)**					
	4,500	**5,000**	**5,500**	**6,000**	**6,500**	**7,000**
Naïve Bayes	42	43	44	43	44	46
J48	53	55	52	54	54	57
SVM	59	60	59	60	**61**	**61**

However, it can be observed that after 3,000 tweets precision is improving very slowly. Also, it can be noted that precision in the interval 3,500-6,000 slightly fluctuates. Thus, test results suggest that 3,000 samples are enough as a training set for a selected topic (cell phones in our case).

3.2 Effect of n-gram Size

We perform tests to study the effect of the n-gram order (size) on the precision of classifiers. We tried six different sizes. Results are shown in Table 3. It confirms that unigram is the best feature size. This conclusion confirms the conclusions obtained in other studies for English language and different corpus domain such as Twitter and films reviews [4, 9].

Table 3. Precision observed when using 6 different n-gram sizes

Classifier	**N-gram size**					
	1	**2**	**3**	**4**	**5**	**6**
Naïve Bayes	46	37	35	35	35	35
J48	57	41	35	35	35	35
SVM	**61**	49	41	35	35	35

3.3 Effect of the Number of Classes

Table 4 describes the values of the number of classes and their composition. For example, in class number 2 there are two types of categories: *positive* and *negative*, but positive value also corresponds to *positive*, *neutral*, and *news* opinions. The class number represents the quantity of classes in the group.

Table 4. Possible combinations for classifying classes

Number of the classes	***Values in the classes***	***Clustering of the values***
2	positive, negative	I. positive: positive, neutral, or news II. negative
3	positive, negative, neutral	I. positive II. negative III. neutral: neutral or news
4	positive, negative, neutral, news	I. positive II. negative III. neutral IV. news

Table 5. Precision observed when using different number of target classes

Classifier	*Number of the classes*		
	2	**3**	**4**
Naïve Bayes	78.2	58.3	46.0
J48	83.6	60.2	57.0
SVM	**85.8**	69.0	61.0

Table 5 shows the effect of the number of classes on the classifier performance. We see that reducing the number of classes increases the classifiers precision. It is not surprising because we decrease the possibility of errors.

3.4 Effect of Balanced vs. Unbalanced Corpus

In this section, our goal was to analyze the effect of balanced vs. unbalanced corpus on classification. We selected 4,000 tweets from the annotated corpus in order to build a balanced subcorpus. Namely, 1,000 tweets were selected for each class, i.e., each class has equal representation. We classified according to the setup of Table 5. Tables 6 and 7 show the results obtained when using an unbalanced and a balanced corpus respectively. We can observe that the best precision was 85.8% for positive and negative classes with the unbalanced corpus. This result is slightly higher than the 82.2% reported for English Twitter in [7]. It is interesting to observe that the precision decreased when using a balanced corpus, though not very much. Perhaps, this behavior was due to the average number of adjectives (1.15) and adverbs (0.58) per tweet in the unbalanced corpus, which is higher than in the balanced corpus (adjectives: 0.98 and adverbs: 0.63). Adjectives and adverbs usually have more sentiment connotations. This phenomenon is part of our future research.

Another interesting point is that the Decision Tree classifier (J48) in general is more stable as far as the effects of balancing of the corpus are concerned.

Table 6. Precision observed when using an unbalanced corpus

Classifier	*Number of classes*		
	2	**3**	**4**
Naïve Bayes	78.2	58.3	46.0
J48	83.6	60.2	57.0
SVM	**85.8**	69.0	61.0

Table 7. Precision observed when using a balanced corpus

Classifier	*Number of classes*		
	2	**3**	**4**
Naïve Bayes	72.0	48.0	31.3
J48	80.9	62.6	46.6
SVM	**81.6**	62.2	54.6

3.5 Effect of Testing on Different Domains

For evaluation of the influence of the domain, the classifiers were trained using the domain of cell phones and were tested both in domain of cell phones and political domain. The data for the political domain were taken from other corpus of tweets, i.e., these are two very different domains. The political domain test set contains 1,400 tweets (positive: 255, negative: 725, neutral: 134, and news: 286). Table 8 shows that training with a corpus that has a domain different from the target domain affects precision very negatively, namely, it is two or three times worse.

Table 8. Precision observed on different domains

Classifier	*Same domain*	*Different domain*
Naïve Bayes	78.2	34.0
J48	83.6	17.0
SVM	**85.8**	28.0

3.6 Best Settings for Practical Applications

We conclude that the best settings for our practical application were:

- Using unigrams (i.e., n-gram size is equal to one),
- Having a training set containing at least 3,000 tweets,
- Using SVM classifier,
- Having two polarity classes (positive and negative) if possible,
- Having training and target sets within the same domain.

When using all the aforementioned settings together we observed a precision of 85.8%.

4 Analysis of Errors

We found some common types of errors when analyzing the misclassified samples. The most frequent errors were:

- Shortened messages,
- Misspelling,
- Humor, irony and sarcasm,
- Human tagging errors.

4.1 Shortened Messages

In the context of Twitter, it is common to see shortened messages like: '*mi celular!!! La pantalla* (*my cellphone !!! the display*)' that correspond to elliptical phrases. Messages like these do not have any sentiment interpretation for other persons; they are understandable basically by the opinion holder himself. The human annotator assigns here usually a negative opinion because he supposes that the display does not work anymore. However, this supposition is based on beliefs of the annotator, and not on the real situation. The SVM classifier assigned here the neutral value, because there is insufficient information for other type of decision.

4.2 Misspelling

Orthographic errors are common in tweets such as '*No me gushta le iPhone*' '*Ya tng iPhone de nuevo* (*I don't like the iPhone; I have an iPhone again*)'. Misspelled words and shortness of the messages make difficult for a classifier to determine the right class. The human annotator marked the first one as negative, and the second one as positive, but the SVM classifier assigned neutral class in both cases.

4.3 Humor, Irony and Sarcasm

Treatment of humor and its subtypes like irony or sarcasm is an interesting but extremely difficult problem. It is very complex because while interpreting humor we often rely on the world knowledge and the (very broad) context, as well as on much more difficult to represent subtle cultural patterns.

Also we should take into account that humor usually is not expressed directly. For example, let us consider the tweet '*Mi novio es como mi iPhone. No tengo.* (*My boyfriend is like my iPhone. I don't have one*)'. It was automatically classified as positive, but the human annotator marked it as neutral. In fact, there is no enough information to guess correctly. These phenomena are difficult to determine without reviewing the context [1].

4.4 Human Tagging Errors

Sometimes, human annotator cannot make decision because of the complexity of the context of a tweet. For example, '*Hablar vale más que un iPhone...Yo tengo tu amor*

(*Talking is worthier than having an iPhone, I have your love*)'. The human annotator marked it as negative, while the classifier marked it as neutral. If we analyze deeper the context, then two facts hold. While it is true that in the cell phone context, not having an iPhone is a negative sentiment, but in the human relationships context, usually it is positive to have a partner.

5 Spanish Emotion Lexicon

For automatic analysis of emotions expressed in tweets, specialized lexical resources are necessary. One of these resources is Spanish Emotion Lexicon. It is developed by I. Díaz-Rangel, G. Sidorov, and S. Suárez-Guerra. They submitted a journal paper where they explain detailed methodology of the creation of the dictionary [15]. Here we present only the general idea of the Lexicon and announce its availability for academic usage.

Spanish Emotion Lexicon contains 2,036 words that are associated with the measure of Probability Factor of Affective use (PFA) with respect to at least one basic emotion: joy, anger, fear, sadness, surprise, and disgust.

We selected the words from English SentiWordNet [14] and translated them automatically into Spanish. Then we manually checked 3,591 obtained words using Maria Moliner dictionary and leave only words that had at least one meaning related with the basic emotions.

Then we asked 19 annotators to evaluate how probable is the association of the word with one of the emotions, i.e., how easily a context (with the word) related with the emotion can be imagined. No semantic analysis was performed. We selected the scale: null, low, medium, high.

After this we used the weighted Cohen's kappa [16] for calculation of agreement between annotators (pairwise) and leave only 10 annotators with the best agreement scores. In this manner, we try to improve the objectivity of the values and eliminate "bad" annotators. The values of kappa were improved about 15%.

At the next step we represent the number of evaluations as percentages, as can be seen in Table 9. For example, for the word *abundancia (abundance)*, 50% of annotators chose "medium" and 50% chose "high".

Table 9. Example of average evaluation for "joy"

Word	Null[%]	Low[%]	Medium[%]	High[%]
abundancia (abundance)	0	0	50	50
aceptable (acceptable)	0	20	80	0
acallar (to silence)	50	40	10	0

Now we are ready to calculate a new measure for each word that we called Probability Factor of Affective use (PFA). It is based on the percentages of Table 9. Note that PFA is 1 if 100% of annotators relate it to the "high" value of the association with the emotion, and it is 0 if 100% of annotators relate it to the "null" value. So, intuitively it has very clear meaning: the higher the value of the PFA is, the more probable the association of the word with the emotion is. We present the exact formula in our submitted paper [15]. For example, for the words in Table 9, *abundancia (abundance)* has PFA=0.83, *aceptable (acceptable)* has PFA=0.594, *acallar (to silence)* has PFA=0.198.

Spanish Emotion Lexicon is available from www.cic.ipn.mx/~sidorov.

6 Conclusions

The large amount of information contained in Twitter makes it an attractive source of data for opinion mining and sentiment analysis. Performance of machine learning techniques is relatively good when classifying sentiments in tweets, both in English and in Spanish. We believe that the precision can be further improved using more sophisticated features.

In this research, we presented an analysis of various parameter settings for selected classifiers: Supported Vector Machines, Naïve Bayes and Decision Trees. We used n-grams of normalized words (additionally filtered using their POS-tags) as features and observed the results of various combinations of positive, negative, neutral, and informative sets of classes. We made our experiments in Spanish language for the topic related to cell phones, and also partially used data from tweets related to the recent Mexican presidential elections (for checking the balanced vs. unbalanced corpus).

From the analysis of the results, we found that the best configuration of parameters was: (1) using unigrams as features, (2) using less possible number of classes: positive and negative, (3) using at least 3,000 tweets as training set (incrementing this value does not improve precision significantly), (4) balancing the corpus as regards the proportional representation of all classes gives slightly worse results, and (5) Supported Vector Machines was the classifier with the best precision.

We also present in this paper Spanish Emotion Lexicon that is useful available resource for analysis of emotions in tweets and in any texts, if we do not perform detailed word sense analysis. The resource contains 2,036 words marked for six basic emotions with Probability Factor of Affective use (PFA).

In future work, we plan to explore richer linguistic analysis, for example, parsing, semantic analysis and topic modeling. Also, better preprocessing is needed in order to avoid errors mentioned above.

Acknowledgments. This work was done under partial support of the Mexican Government (SNI, COFAA-IPN, PIFI-IPN, SIP-IPN 20120418, 20121823, 20113295 and 20111146, CONACYT 50206-H and 83270), CONACYT-DST India (“Answer Validation through Textual Entailment”), Mexico City Government (ICYT PICCO10-120), and FP7-PEOPLE-2010-IRSES: Web Information Quality - Evaluation Initiative (WIQ-EI) European Commission project 269180.

References

1. Pang, B., Lee, L.: Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 2, 1–135 (2008)

2. Liu, B.: Sentiment Analysis and Subjectivity. Handbook of Natural Language Processing. Ed(s) Indurkhya, N. and Damerau, F.J., 2nd ed. (2010)
3. Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the ACL-2004 (2004)
4. Pang, B., Lee, L., and Vaithyanathan S.: Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the ACL, pp. 79–86. Association for Computational Linguistics (2002)
5. Polanya,L., Zaenen, A.: Contextual Valence Shifters. Computing Attitude and Affect in Text: Theory and Applications In Computing Attitude and Affect in Text: Theory and Applications, Vol. 20 (2006)
6. Wilson, T., Hoffmann, P., Somasundaran, S., Kessler, J., Wiebe, J., Choi, Y., Cardie, C., Riloff, E., Patwardhan, S.: Opinion Finder: a system for subjectivity analysis, Proceedings of HLT/EMNLP on Interactive Demonstrations, pp. 34–35, Vancouver, British Columbia, Canada (2005)
7. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford University, Stanford, CA. (2009)
8. Martínez Cámara, E., Martín Valdivia, M.T., Perea Ortega, J.M., Ureña López, L.A.: Técnicas de Clasificación de Opiniones Aplicadas a un Corpus en Español. Procesamiento del Lenguaje Natural, Revista nº 47, pp. 163–170 (2011)
9. Aiala, R., Wonsever, D., Jean-Luc, M.: Opinion Identification in Spanish Texts. Proceedings of the NAACL HLT (2010)
10. Padró, L., Collado, M., Reese, S., Lloberes, M., Castellón, I.: FreeLing 2.1: Five Years of Open-Source Language Processing Tools. Proceedings of 7th Language Resources and Evaluation Conference, La Valletta, Malta (2010)
11. Das, S., Chen, M.: Yahoo! For Amazon: Extracting market sentiment from stock message boards. In Proceedings of the 8th Asia Pacific Finance Association Annual Conference (2001)
12. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1(2009)
13. EAGLES. Recommendations for the morphosyntactic annotation of corpora, *Eag-tcwg-mac/r*, ILC-CNR, Pisa (1996)
14. Esuli, A., Sebastiani, F.: SentiWN: A Publicly Available Lexical Resource for Opinion Mining. In: Fifth international conference on Language Resources and Evaluation (LREC 2006), pp. 417–422 (2006)
15. Díaz-Rangel, I., Sidorov, G., Suárez-Guerra, S.: Weighted Spanish Emotion Lexicon. (submitted) (2012)
16. Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement, 20, pp. 37–46 (1960)