# Sentiment Analysis: from Binary to Multi-Class Classification

## A Pattern-Based Approach for Multi-Class Sentiment Analysis in Twitter

Mondher Bouazizi
Graduate School of Science and Technology
Keio University
Yokohama, Japan
Email: bouazizi@ohtsuki.ics.keio.jp

Tomoaki Ohtsuki
Department of Information and Computer Science
Faculty of Science and Technology, Keio University
Yokohama, Japan
Email: ohtsuki@ics.keio.ac.jp

***Abstract*—Most of the state of the art works and researches on the automatic sentiment analysis and opinion mining of texts collected from social networks and microblogging websites are oriented towards the classification of texts into positive and negative. In this paper, we propose a pattern-based approach that goes deeper in the classification of texts collected from Twitter (i.e., tweets). We classify the tweets into 7 different classes; however the approach can be run to classify into more classes. Experiments show that our approach reaches an accuracy of classification equal to 56.9% and a precision level of sentimental tweets (other than neutral and sarcastic) equal to 72.58%. Nevertheless, the approach proves to be very accurate in binary classification (i.e., classification into "positive" and "negative") and ternary classification (i.e., classification into "positive", "negative" and "neutral"): in the former case, we reach an accuracy of 87.5% for the same dataset used after removing neutral tweets, and in the latter case, we reached an accuracy of classification of 83.0%.**

## I. Introduction

Twitter became one of the biggest web destinations: a very popular platform for people to express their thoughts about products [1] [2] or movies [3], share their daily experience and communicate their opinion about real-time and upcoming events, such as sports or political elections [4], etc.

This ecosystem presents a very rich, source of data to mine. However, due to the limitation in terms of characters (i.e. 140 characters per tweet), mining such data present lower performance than that when mining longer texts. In addition, classification into multiple classes remains a challenging task: binary classification of a text usually relies on the sentiment polarity of its components (i.e., whether they are positive or negative). However, when positive and negative classes are divided into subclasses, the accuracy tends to decrease remarkably.

In this paper, we propose an approach that relies on writing patterns, and special unigrams to classify tweets into 7 different classes, and demonstrate how the proposed approach presents good performances (i.e., classification accuracy and precision). The main contributions present in this paper are as follows:

1) We propose a set of pattern-based features, along with other features to classify tweets.
2) We classify tweets into 7 different sentiment classes.

The remainder of this paper is structured as follows: In Section II we present our motivations and describe some of the related work. In Section III we formally define the aim of our work and describe in details the proposed method. In Section IV we detail our experiments and the results obtained. Section V concludes this paper and proposes possible directions for future work.

## II. Motivations and Related Work

### A. Motivations

Social networks and microblogging websites such as Twitter have been the subject to many studies in the recent few years. Automatic sentiment analysis and opinion mining present a hot topic of study. Social networks present a huge source of data representing the opinions of a significant, yet totally random, proportion of users and customers who are using a product of a service. However, due to the informal language used, the presence of non-textual content and the use of slang words and abbreviations, classification of data extracted from such microblogging websites is rather a challenging task. Ghag et al. [5] defines *"Hidden Sentiment Identification"* which is the identification of the real feeling rather than the sentiment polarity, *"Handling Polysemy"* which is the existence of multiple meanings that might have different sentiment polarity for the same word, and *"Mapping Slangs"* which is the identification of the meaning and the polarity of slang words, among others as the most challenging tasks facing the sentiment analysis of short microblog texts.

On a related context, the state of the art proposed approaches are mostly focusing on the binary and ternary sentiment classification. In other words, they classify texts either into *"positive"* and *"negative"*, or into *"positive"*, *"negative"* and *"neutral"*. However, to study the opinion of a user, it would be more interesting to go deeper in the classification, and detect the sentiment hidden behind his post. Following two examples of tweets which are negative, however, reflect two completely different aspects:

- *"Damn damn.. no iPhone support for windows XP x64. There are some workarounds, but I can't figure this out."*

978-1-4799-6664-6/16/$31.00 ©2016 IEEE

- *"Noooooooooo! My iPhone glass cracked :("*

In the first example, the user is expressing his fury towards the absence of support of his phone on an operating system. However, in the second he is expressing some feeling of sadness because of a physical problem his phone faced. The first example shows some important information regarding the satisfaction of the user, therefore, it might be more important to study. However, in general, both information can be used, yet, they have to be distinguished from each other.

### B. Related Work

Twitter data mining has been a hot topic of research in the last few years. Nature of the data mined varies widely depending on the aim and the final result expected. Consequently, the techniques used to process data and extract the needed information are different.

Akcora et al. [6] proposed a method to determine the changes in public opinion over the time, and identify the news that led to breakpoints in public opinion. In a related context, Sriram et al. [7] proposed a method to classify tweets depending on their natures into a set of classes including private messages, opinions and event, etc.

However, most of the work has been focusing on the content of the tweets and how to extract opinions of users towards specific topics or objects. The work of Pang et al. [8] presented the pioneer work for the use of machine learning to classify texts based on their sentiment polarity. In their work, the authors used unigrams, bigrams and adjectives in different ways to classify a set of movie reviews into positive or negative. Other works iterated more on the idea, and new types of features have been used for the classification, depending on the aim and application: Boia et al. [9] and Manuel et al. [10] proposed two approaches that, respectively, rely on emoticons to detect the polarity of tweets and on slang words to assign a sentiment score to online texts. These two works proved how non-textual components can be used to detect the polarity of a text.

More recent works went deeper, and new models have been built: Gao et al. [11] proposed a recent approach that focus in the repartition or the frequency of sentiment classes in the set they analyze. Moving from classification to quantification, the authors concluded that using a quantification-specific algorithm presents a better frequency estimation than using regular classification-oriented algorithms.

Few works have been conducted on the multi-class sentiment analysis. Most of them focused on assessing the sentiment strength into different sentiment strength levels (e.g., *"very negative"*, *"negative"*, *"neutral"*, *"positive"* and *"very positive"*) or simply give numeric sentiment scores to the texts [12] [13]. Nevertheless, other works were conducted to classify texts into different sentiment classes: Lin et al. [14] [15] proposed an approach that classifies documents into reader-emotion categories. They relied on what they qualify of similarity features and word emotion features along with other basic features. The approach, although it shows some potential, is oriented towards the reader rather than the writer. Therefore, the sentiment classes proposed are different from what a writer might intend to show. Similarly, Ye et al. [16] studied the problem of emotion detection of news articles from reader's perspective, and tried various multi-label classification methods and different strategies for features selection to conclude which are to be adopted to solve the problem. Liang et al. [17] proposed an emoticon recommendation system that recommends emoticons for posted texts to help to author decide which emoticon to insert to show what he intends.

TABLE I: Structure of the Dataset Used

| Class | Training set | Test set |
|---|---|---|
| Happiness | 3000 | 225 |
| Love | 3000 | 219 |
| Sadness | 3000 | 223 |
| Anger | 3000 | 201 |
| Hate | 3000 | 157 |
| Sarcasm | 3000 | 199 |
| Neutral | 3000 | 176 |
| **Total** | **21 000** | **1400** |

## III. Proposed Approach

### A. Problem Statement

Given a set of tweets, we aim to classify each one of them to one of the following 7 classes: *"happiness"*, *"sadness"*, *"anger"*, *"love"*, *"hate"*, *"sarcasm"* and *"neutral"*. Therefore, from each tweet, we extract a 4 set of features, refer to a training set and use machine learning algorithms to perform the classification.

### B. Data

For the sake of this work, we manually collected and prepared 2 datasets as follow:

- **Set 1:** this set contains 21 000 tweets which have been manually classified into the 7 classes, each containing 3 000 tweets. This set is used for training. Therefore, in the rest of this work, it will be referred to as the *"training set"*.
- **Set 2:** this set contains 1400 tweets. All tweets are manually checked and classified into the 7 classes. This set will serve as a test set. Therefore, in the rest of this work, it will be referred to as the *"test set"*.

The structure of the dataset used is shown in TABLE I.

### C. Features Extraction

Under different emotional conditions, humans tend to behave differently. This includes the way they talk and express their feelings. Therefore, it might be important to rely, not only on the vocabularies used, but also on the expressions and sentence structures used under the different conditions, to quantify and model these feelings. Therefore, in the rest of this section, we rely on these assumptions to extract the following four families of features:

*1) Sentiment-based features:* Sentiment-based features are ones based on the sentiment polarity (i.e., *"positive"*/*"negative"*) of the different components of tweets. We first extract emotional scores from words using SentiStrength. SentiStrength attributes sentiments scores to words, where negative words have scores varying from -1 (almost negative) -5 (extremely negative) and positive words have scores varying from 1 (almost positive) to 5 (extremely positive). We uses SentiStrength to extract the following features:

- Total score of positive words denoted by $PW$
- Total Score of negative words denoted by $NW$ (this score is positive)
- Number of highly emotional positive words (i.e., having score equal to or more than 3) denoted by $N_{pw}$
- Number of highly emotional negative words (i.e., having score equal to or less than -3) denoted by $N_{nw}$
- Ratio of emotional words $\rho(t)$ defined as

$$\rho(t) = \frac{PW(t) - NW(t)}{PW(t) + NW(t)} \quad (1)$$

where $t$ is the tweet. In case the tweet does not contain any emotional word, $\rho$ is set to 0.

We then add four more features by counting the number of positive, negative, joking (or ironic) and neutral emoticons. Joking emoticons are emoticons used sometimes with ironical or sarcastic statements (e.g., *":P"*). Hashtags also have emotional content. In some cases, they are used to disambiguate the real intention of the twitterer conveyed in his message, particularly when he is being sarcastic. Therefore, we count also the number of positive and negative hashtags.

We then define 4 features that represent whether there is a sentiment contrast between the different components. By contrast we mean the coexistence of a negative component and a positive one within the same tweet: we extract such contrast between words, between hashtags, between words and hashtags and between words and emoticons, and use them as extra features.

*2) punctuation and syntax-based features:* In addition to sentiment-based features, we extract a second set of features we qualify of punctuation and syntax-based features. A certain use of punctuation marks, the repetition of vowels or the employment of all-capital words may show how intense the sentiment of the person is. To detect such aspects, we extract the following set of features: number of exclamation marks, number of question marks, number of dots, number of all-capital words and number of quotes.

We also add a sixth feature by checking if any of the words contains a vowel that is repeated more than twice (e.g., *"looooove"*). If such a word exists, the feature is set to *"true"*, otherwise, it is set to *"false"*.

*3) Unigram-based features:* Since proposed by Pang et al. [8], unigrams and *n*-grams in general, have been used as basic features for sentiment analysis using machine learning. In the different approaches, unigrams are collected from the training datasets, and either the count or the presence of these unigrams are used as features for the classification. In our work, we make use of WordNet [18] to collect unigrams related to each sentiment class. We start with a set of seed words few in number for each class, and used WordNet to collect their synonyms and hyponyms down to a certain depth.

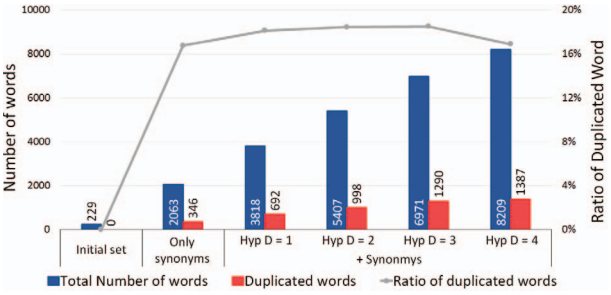


Fig. 1: Number of unigrams collected from WordNet using the seed words proposed

The choice of synonyms and hyponyms is based on the fact that these words are highly correlated with the initial seed word, and usually describe the same object, if not a more precise one. While synonyms refer usually to equivalent terms, hypernyms and hyponyms show the relationship between the more general term and its more specific instances. A hypernym, or a superordinate, is a broader term than a hyponym, whereas a hyponym is a word or an expression which is more specific than its hypernym. For example, for the word *"feeling"*, two of its direct hypernyms are *"perception"* and *"idea"*. The words *"happiness"*, *"anger"* and *"fear"* are some of its hyponyms. Hypernyms might lose some of the specificities of the initial word, therefore, in our study, we collect only synonyms and hyponyms of the seed words. On the other hand, hyponyms also might lose the original meaning of the word, and collide with some of other classes. Therefore, the depth down to which we collect the hyponyms is set to a certain value $D_{hypo}$, which is a parameter to optimize.

We start with an initial set of seed words for each class (except the class "neutral"). The words selected are nouns, adjectives and verbs. We then collect the synonyms and hypernyms up to different depths. Fig. 1 shows the number of words for each depth as well as the number/ratio of duplicated terms in different classes. To obtain as much terms as possible, while maintaining a low duplication ratio and keeping in mind that the deeper we go, the more we lose in the original meaning of the word, we set $D_{hypo}$ to 1. The words are associated to the classes described, and are given the absolute value of scores returned by SentiStrength (if a word has a score equal to 0 in SentiStrength, we give it a score equal to 1).

We use the resulted sets of words to extract 6 features, by counting the occurrences of the words in the tweet to classify, taking into consideration the score of the words (words are given positive score regardless of their class).

*4) Pattern-based features:* The idea of our pattern-related features is inspired from our previous work [19], in which we proposed an approach that relies on Part of Speech tags (PoS-tags) to extract sarcastic patterns. In our current work, we

TABLE II: Expressions Used to Replace the Words of EI and GFI

| PoS-tag | Expression |
|---|---|
| "CD" | [CARDINAL] |
| "FW" | [FOREIGNWORD] |
| "UH" | [INTERJECTION] |
| "LS" | [LISTMARKER] |
| "NN", "NNS", "NNP", "NNPS", | [NOUN] |
| "PRP", "PRP$" | [INTERJECTION] |
| "MD" | [MODAL] |
| "RB", "RBR", "RBS" | [ADVERB] |
| "VB", "VBD", "VBG", "VBN", "VBP", "VBZ" | [VERB] |
| "WDT", "WP", "WP$", "WRB" | [WHDETERMINER] |
| "SYM" | [SYMBOL] |

TABLE III: Part-of-Speech Tag Categories

| Class | PoS Tags |
|---|---|
| CI | "CC", "DT", "EX", "IN", "MD", "PDT", "POS", "RB", "RBR", "RBS", "RP", "TO", "WDT", "WP", "WP$", "WRB" |
| GFI | "CD", "FW", "LS", "NNP", "NNPS", "PRP", "PRP$", "SYM", "UH" |
| EI | "JJ", "JJR", "JJS", "NN", "NNS", "VB", "VBD", "VBG", "VBN", "VBP", "VBZ" |

rely on PoS-Tag of words to extract similar patterns. However, instead of dividing words into two categories, we divide them into three: a first one, referred to as *EI*, containing words which might have emotional content, a second one, referred to as *"CI"*, containing non emotional words whose content is important and a third one, referred to as *"GFI"*, containing the words whose grammatical function is important. If a word belongs to the first category, it is replaced by the corresponding expression shown in TABLE II along with its polarity (e.g., the word *"good"* would be replaced by *POS-ADJECTIVE*); if it belongs to the second, it is lemmatized and replaced by its lemma; and if it belongs to the third, it is replaced by the corresponding expression shown in TABLE II.

As mentioned above, the classification into categories is done based on the PoS-tag of the word. The list of part-of-speech tags and their category is given in TABLE III.

We generate the vector of words for each tweet as defined. For example, the following PoS-tagged tweet *"He_PRP is_VBP dummy_JJ ,_, why_WP would_VBD you_PRP think_VBP I_PRP want_VBP to_TO go_VB with_IN him_PRP !!!!_."* gives, among others, the following pattern vector [PRONOUN VERB NEG-ADJECTIVE . *why* VERB PRONOUN VERB PRONOUN POS-VERB *to* VERB *with* PRONOUN .].

We define a pattern as an ordered sequence of words. Patterns are extracted from the training set such as their lengths satisfy:

$$L_{min} \leq Length(pattern) \leq L_{max} \tag{2}$$

where $L_{min}$ and $L_{max}$ represent respectively the minimal and maximal allowed length of patterns in $words$ and $Length(pattern)$ is the length of the pattern in $words$. The number of pattern lengths will be referred to as $N_L$ and is equal to $(L_{max} - L_{min} + 1)$. Only patterns that appear at least $N_{occ}$ times in our training set for the same class are kept; the others are discarded. We then divide the resulted patterns into $N_F$ sets where:

$$N_F = N_L \times N_C \tag{3}$$

where $N_L$ is the number of pattern lengths and $N_C$ is the number of classes (7 in our case).

TABLE IV: Pattern Features

| | | Pattern length | | | |
|---|---|---|---|---|---|
| | | $L_1$ | $L_2$ | $\cdots$ | $L_N$ |
| Sentiment Class | 1 | $F_{11}$ | $F_{12}$ | $\cdots$ | $F_{1N}$ |
| | $\vdots$ | $\vdots$ | $\vdots$ | $\ddots$ | $\vdots$ |
| | 7 | $F_{71}$ | $F_{72}$ | $\cdots$ | $F_{7N}$ |

We create $N_F$ features, as shown in TABLE IV. Each feature $F_{ij}$ of the table represents the degree of resemblance of the tweet to the patterns of sentiment class $i$ and length $j$. Therefore, given a tweet $t$, we calculate the resemblance degree $res(p,t)$ of each pattern in the training set $p$ to the tweet $t$ [19]:

$$res(p,t) = \begin{cases} 1, & \text{if the tweet vector contains the pattern as it is, in the same order,} \\ \alpha \cdot n/N, & \text{if } n \text{ words out of the } N \text{ words of the pattern appear in the tweet in the correct order,} \\ 0, & \text{if no word of the pattern appears in the tweet.} \end{cases}$$

Given the $K$ patterns that have the highest resemblance to the pattern $p$ among the patterns extracted from the class $i$ which have a length $j$, the value of the feature $F_{ij}$ is

$$F_{ij} = \beta_j * \sum_{k=1}^{K} res(p_k, t) \tag{4}$$

where $\beta_j$ is a weight given to patterns of length $L_j$ (regardless of their class). We give different weights for each length of pattern since longer patterns are more likely to have higher impact. $F_{ij}$ as defined measures the degree of resemblance of a tweet $t$ to patterns of class $i$ and length $j$.

In our previous work [19], we demonstrated that the optimal values for $N_{occ}$, $L_{min}$, $L_{max}$, $\alpha$ and $\beta_i$ are as follows:

$$\begin{cases} N_{occ} = 2, \\ L_{min} = 3, \\ L_{max} = 10, \\ \alpha = 0.03, \\ \beta_n = (n-1)/(n+1), \qquad \forall n \in \{3, \ldots, 10\}. \end{cases}$$

On the other hand the parameter $K$ has been introduced in this work since we noticed a high imbalance between the number of patterns for each class. Fig. 2 shows the classification accuracy using pattern-based features for different values

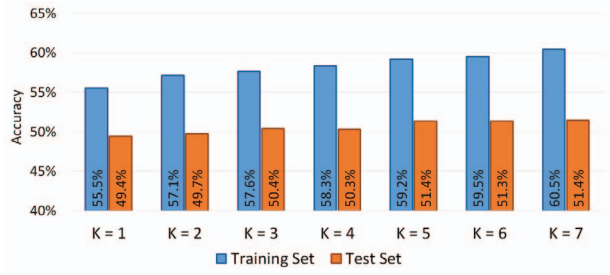


Fig. 2: Accuracy of Classification Using Pattern-Based Features for Different Value of $K$

TABLE V: Binary Classification Accuracy, Precision, Recall and and F-Measure

| | Accuracy | Precision | Recall | F-Measure |
|---|---|---|---|---|
| Positive | 90.5% | 82.4% | 90.5% | 86.3% |
| Negative | 85.2% | 92.2% | 85.2% | 88.6% |
| **Overall** | **87.5%** | **87.9%** | **87.5%** | **87.6%** |

of $K$. According to the figure, the optimal value is 5. Higher values enhance the accuracy during cross-validation, but have no big impact on that of the test set.

In the next section, we evaluate the model we built, and present the results of our experiments in the cases of binary, ternary and multi-class classification.

## IV. Experimental Results

After the extraction of features, we run different test using *"Random Forest"* [20] classifier. We use 4 Key Performance Indicators (KPIs) to evaluate the effectiveness of our approach: Accuracy, Precision, Recall and F-measure which is defined as follows:

$$F\text{-}measure = 2 \cdot \frac{precision \cdot recall}{precision + recall}. \quad (5)$$

### A. Binary Classification

We first run our experiment to detect the sentiment polarity of tweets. For this sake, we remove the tweets belonging to the classes *"neutral"* and *"sarcasm"*, and grouped the other classes into two main classes which are *"positive"* and *"negative"*. The former class contains tweets from the classes *"love"* and *"happiness"*, while the latter contains tweets from the classes *"hate"*, *"anger"* and *"sadness"*. TABLE V shows the results of classification. The accuracy obtained reaches 87.51%. Noticeably, the recall of positive tweets is the highest (i.e., 90.5%), however the precision of negative tweets is the highest (i.e., 92.2%). This means that tweets which are classified as negative are mostly negative. However, tweets which have positive polarity tend to be classified more correctly as shown in the confusion matrix presented in TABLE VI.

### B. Ternary Classification

Despite its importance, binary classification supposes that the given data are already known to be emotional. However, Twitter contains many tweets which have no emotional polarity such as news tweets, etc. Therefore, in this subsection we add neutral tweets as shown before in the description of our dataset. We then rely on the same set of features to classify the tweets. The results obtained are given in TABLE VII, and the confusion matrix of classification is given in TABLE VIII.

TABLE VI: Binary Classification Confusion Matrix

| **Class** | **Classified as** | |
|---|---|---|
| | Positive | Negative |
| Positive | **402** | 42 |
| Negative | 86 | **495** |

The obtained results show that the introduction of a third class decreases noticeably the accuracy to reach 83.0%. The new class (i.e., *"neutral"*) presents a low accuracy, but a very high precision rate. This can be explained by the fact that the amount of training data (i.e., number of tweets) for this class is lower than that for the other classes. Therefore, a tweet that meets the conditions of the class *"neutral"* can be easily detected by the classifier as *"neutral"*. However, not many of them meet the condition, and therefore, they are misclassified. Overall, the results obtained are promising.

### C. Multi-class classification

In this subsection, we use the 7 sentiment classes that we described in Section III. The classification results are given in TABLE IX, while the confusion matrix is given in TABLE X.

Despite the number of classes, the accuracy obtained is equal to 56.9%, with a precision that reaches 65.2%. More interestingly, some sentiments seem to be easier to detect than others. In particular, tweets belonging to the class *"happiness"* were classified with an accuracy equal to 83.1%. This shows that tweets belonging to this class are easily distinguished from other classes. This might be due to the fact that, contrarily to negative tweets, positive tweets belong to mainly two classes, easy to distinguish from each other. Negative tweets on the other hand are closer to each other. A typical example is given by the following tweet: *"Damn it.. I really hate when this happens. This crap doesn't want to work!!!"*. In this tweet, the user expresses both sentiments of anger and hate. However, since he explicitly uses the word *"hate"* the tweet would be classified as belonging to the class *"Hate"*, although it shows sentiments of anger more than hate.

TABLE VII: Ternary Classification Accuracy, Precision, Recall and F-Measure

| | Accuracy | Precision | Recall | F-Measure |
|---|---|---|---|---|
| Positive | 90.3% | 77.6% | 90.3% | 83.5% |
| Negative | 85.0% | 86.5% | 85.0% | 85.8% |
| Neutral | 58.2% | 90.4% | 58.2% | 70.8% |
| **Overall** | **83.0%** | **83.8%** | **83.0%** | **82.7%** |

TABLE VIII: Ternary Classification Confusion Matrix

| **Class** | **Classified as** | | |
|---|---|---|---|
| | Positive | Negative | Neutral |
| Positive | **401** | 38 | 5 |
| Negative | 81 | **494** | 6 |
| Neutral | 35 | 39 | **103** |

TABLE IX: Multi-Class Classification Accuracy, Precision, Recall and F-Measure

| | Accuracy | Precision | Recall | F-Measure |
|---|---|---|---|---|
| Happiness | 83.1% | 82.4% | 83.1% | 82.7% |
| Love | 43.4% | 59.7% | 43.4% | 50.3% |
| Neutral | 62.7% | 69.4% | 62.7% | 65.9% |
| Sadness | 48.9% | 71.7% | 48.9% | 58.1% |
| Anger | 42.3% | 77.3% | 42.3% | 54.7% |
| Hate | 59.2% | 68.4% | 59.2% | 63.5% |
| Sarcasm | 59.1% | 25.7% | 59.1% | 35.8% |
| **Overall** | **56.9%** | **65.2%** | **56.9%** | **58.8%** |

TABLE X: Multi-Class Classification Confusion Matrix

| **Class** | **Classified as** | | | | | | |
|---|---|---|---|---|---|---|---|
| | H | L | N | Sd | A | H | Sr |
| Happiness (H) | **187** | 15 | 1 | 1 | 0 | 0 | 21 |
| Love (L) | 14 | **95** | 10 | 10 | 3 | 3 | 84 |
| Neutral (N) | 4 | 1 | **111** | 8 | 2 | 3 | 48 |
| Sadness (Sd) | 0 | 4 | 6 | **109** | 6 | 15 | 83 |
| Anger (A) | 6 | 5 | 8 | 16 | **85** | 14 | 67 |
| Hate (H) | 2 | 8 | 8 | 6 | 4 | **93** | 36 |
| Sarcasm (Sr) | 14 | 31 | 16 | 2 | 10 | 8 | **117** |

On the other hand, presence of the class *"sarcasm"* was a main reason which led to the low classification accuracy. The presence of sarcastic tweets engenders the misclassification of many tweets. Although it has a relatively high classification accuracy, many tweets are misclassified as sarcastic. Therefore, arises the necessity of detecting sarcastic statements in a first stage to discard them before classification. The work presented in [19] presents good accuracy for classification of tweets into sarcastic and non-sarcastic.

However, globally, we can confirm that classifying tweets into separate sentiment classes is a challenging task: as mentioned above, many tweets present more than one sentiment. Therefore, a more interesting task would be quantifying the sentiments present in the tweet: a tweet should be attributed more than one sentiment with different scores.

## V. Conclusion

In this paper, we have proposed a new approach for sentiment analysis, where a set of tweets is to be classified into 7 different classes. The obtained results show some potential: the accuracy obtained for multi-class sentiment analysis in the data set used was 56.9%. However, we believe that a more optimized training set would present better performances.

Throughout this work, we demonstrated that multi-class sentiment analysis can achieve high accuracy level, but it remains a challenging task. A more interesting task is to quantify sentiments present in the tweet. Therefore, in a future work, we will use the results obtained for ternary classification (which achieved an accuracy equal to 83.0%) to classify tweets into *"positive"*, *"negative"* and *"neutral"*. The classified sentimental tweets (i.e., which have been classified as *"positive"* or *"negative"*) will then be given scores for the corresponding sentiment subclasses. Eventually, we will use the work [19] in an earlier stage to put aside sarcastic tweets.


## Acknowledgment

The research results have been achieved by "Cognitive Security: A New Approach to Securing Future Large Scale and Distributed Mobile Applications," the Commissioned Research of National Institute of Information and Communications Technology (NICT) , JAPAN.


## References


[1] B. O'Connor, R. Balasubramanyan, B. Routledge, and N. Smith, "From tweets to polls: Linking text sentiment to public opinion time series," in *Proc. Int. AAAI Conf. Weblogs and Social Media*, pp. 26–33, May 2010.

[2] M. A. Cabanlit and K. J. Espinosa, "Optimizing N-gram based text feature selection in sentiment analysis for commercial products in Twitter through polarity lexicons," in *Proc. 5th Int. Conf. Inform., Intelligence, Syst. and Applicat.,* pp. 94–97, July 2014.

[3] U. R. Hodeghatta, "Sentiment analysis of Hollywood movies on Twitter," in *Proc. IEEE/ACM ASONAM,* pp. 1401–1404, Aug. 2013.

[4] J. M. Soler, F. Cuartero, and M. Roblizo, "Twitter as a tool for predicting elections results," in *Proc. IEEE/ACM ASONAM*, pp. 1194–1200, Aug. 2012.

[5] K. Ghag and K. Shah, "Comparative analysis of the techniques for sentiment analysis," in *Proc. Int. Conf. Advances in Technology and Eng.*, pp. 1–7, Jan. 2013.

[6] C. G. Akcora, M. A. Bayir, M. Demirbas, and H. Ferhatosmanoglu, "Identifying breakpoints in public opinion," in *Proc. First Workshop on Social Media Analytics*, pp. 62–66, July 2010.

[7] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas, "Short text classification in twitter to improve information filtering," in *Proc. 33rd Int. ACM SIGIR Conf. Research and development in information retrieval*, pp. 841–842, July 2010.

[8] B. Pang, L. Lillian, and V. Shivakumar, "Thumbs up?: Sentiment classification using machine learning techniques," in *Proc. ACL-02 Conf. Empirical Methods in Natural Language Process.*, vol. 10, pp.79–86, July 2002.

[9] M. Boia, B. Faltings, C.-C. Musat and P. Pu, "A :) is worth a thousand words: How people attach sentiment to emoticons and words in tweets," in *Proc. Int. Conf. Social Computing*, pp. 345–350, Sept. 2013.

[10] K. Manuel, K. V. Indukuri and P. R. Krishna, "Analyzing internet slang for sentiment mining," in *Proc. 2nd Vaagdevi Int. Conf. Inform. Technology for Real World Problems*, pp. 9–11 Dec. 2010.

[11] W. Gao and F. Sebastiani, "Tweet Sentiment: From Classification to Quantification," in *Proc. IEEE/ACM ASONAM*, pp. 97–104, Aug. 2015.

[12] Y.H.P.P. Priyadarshana, K.I.H. Gunathunga, K.K.A. Nipuni N.Perera, L. Ranathunga, P.M. Karunaratne, and T.M. Thanthriwatta, "Sentiment analysis: Measuring sentiment strength of call centre conversations," in *Proc. IEEE ICECCT*, pp.1–9, March 2015.

[13] R. Srivastava and M.P.S. Bhatia, "Quantifying modified opinion strength: A fuzzy inference system for Sentiment Analysis," in *Proc. Int. Conf. Advanced in Computing, Communications and Informatics*, pp.1512–1519, Aug. 2013.

[14] K.H. Lin, C. Yang and H.Chen, "What emotions do news articles trigger in their readers?," in *Proc. ACM SIGIR '07*, pp. 733–734, July 2007.

[15] K.H. Lin, ; C. Yang and H Chen, "Emotion Classification of Online News Articles from the Reader's Perspective," in *Proc. IEEE/WIC/ACM WI-IAT '08*, vol.1, pp.220–226, Dec. 2008.

[16] L. Ye, R. Xu and J. Xu, "Emotion prediction of news articles from reader's perspective based on multi-label classification," in *Proc. Int. Conf. Machine Learning and Cybernetics*, vol.5, pp. 2019–2024, July 2012.

[17] W. Liang, H. Wang, Y. Chu and C. Wu, "Emoticon recommendation in microblog using affective trajectory model," in *Proc. Annual Summit and Conf. Asia-Pacific Signal and Information Processing Association (APSIPA)*, pp.1–5, Dec. 2014.

[18] C. Fellbaun, *WordNet: an Electronic Lexical Database*, Cambridge, Massachusetts, 1998.

[19] M. Bouazizi and T. Ohtsuki, "Sarcasm detection in Twitter," to be published in *IEEE Globecom*, Dec. 2015.

[20] L. Breiman, "Random Forest," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Jan. 2001.