BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 03/05 pp181–194
Volume 2, Number 3, September 2019
DOI: 10.26599/BDMA.2019.9020002

Multi-Class Sentiment Analysis on Twitter: Classification Performance and Challenges

Mondher Bouazizi* and Tomoaki Ohtsuki

Abstract: Sentiment analysis refers to the automatic collection, aggregation, and classification of data collected online into different emotion classes. While most of the work related to sentiment analysis of texts focuses on the binary and ternary classification of these data, the task of multi-class classification has received less attention. Multi-class classification has always been a challenging task given the complexity of natural languages and the difficulty of understanding and mathematically "quantifying" how humans express their feelings. In this paper, we study the task of multi-class classification of online posts of Twitter users, and show how far it is possible to go with the classification, and the limitations and difficulties of this task. The proposed approach of multi-class classification achieves an accuracy of 60.2% for 7 different sentiment classes which, compared to an accuracy of 81.3% for binary classification, emphasizes the effect of having multiple classes on the classification performance. Nonetheless, we propose a novel model to represent the different sentiments and show how this model helps to understand how sentiments are related. The model is then used to analyze the challenges that multi-class classification presents and to highlight possible future enhancements to multi-class classification accuracy.

Key words: Twitter; sentiment analysis; machine learning

1 Introduction

Over the recent years, increasing attention has been paid to the analysis of data collected from social networks and microblogging websites. This is because people tend to discuss all sorts of topics using these services; topics that might include not only their daily affairs and plans, but also some services or products they are using. That being the case, companies and organizations nowadays are trying to analyze posts and discussions of users to extract all possible useful information regarding whether or not they are interested in a given topic, the level of satisfaction of users towards products and services[1,2], or even their intentions and expectations regarding upcoming elections, sports events, etc.[3] One type of information that has been a hot topic of research in the last few years surrounds the identification of attitudes or opinions expressed by users in their posts towards a specific topic. This process is called "sentiment analysis".

Twitter, a popular microblogging website, offers for users a service allowing them to post and interact with short messages. It has some unique properties that make it interesting for companies, such as its openness, the length limitation on messages posted, and the wide use of hashtags. While most social networks require a connection between two users before they can access each other's posts, Twitter allows users to follow one another even if no mutual relation has been established, which makes it easy to collect information from Twitter. Furthermore, posts are limited to 140 characters, which means that messages are brief and usually include just one main piece of information. Due to the wide use of hashtags, companies can easily trace "tweets" (i.e.,

• Mondher Bouazizi and Tomoaki Ohtsuki are with the Department of Information and Computer Science, Keio University, Yokohama 223-8542, Japan. E-mail: bouazizi@ohtsuki.ics.keio.ac.jp; ohtsuki@ics.keio.ac.jp.

∗ To whom correspondence should be addressed.

Manuscript received: 2018-11-10; accepted: 2019-02-13

messages posted by Twitter users) that deal with their own products or services.

This makes the process of automatically performing sentiment analysis on tweets an interesting task: not only can tweets dealing with a given topic be collected quite easily (due to the presence of hashtags), but also the information included in a large enough number of tweets usually represents, with a certain level of fidelity, the opinion of a random, but representative, set of people towards the given topic.

However, some challenges remain in automatic analyzing tweets. According to Ghag and Shah[4], these challenges include, but are not limited to, opinion object identification, maintaining opinion time, and hidden sentiments identification. While most of the work done on sentiment analysis deals with the detection of the sentiment polarity of tweets (i.e., whether they are positive, negative, or neutral), hidden sentiment identification refers to the identification within the tweet of actual hidden sentiments such as anger, happiness, disgust, and joy.

In this paper, we investigate this challenge and present the obstacles that render it difficult to identify the actual sentiment of a given tweet. We perform a multi-class sentiment analysis of tweets and discuss how the number of sentiment classes impacts the classification results. We propose a new model to represent sentiments, and use it to show the relationships between the different sentiments and to explain why the task of multi-class sentiment analysis is inherently difficult.

The remainder of this paper is structured as follows. In Section 2, we present our motivation for this work and discuss some previous research dealing with the multi-class sentiment analysis. In Section 3, we describe the data sets we used for this work, and present the procedure of extraction of features from tweets. In Section 4, we present our different experiments and the obtained results. In Section 5, we introduce our model for representing sentiments and the relation between them, discuss the classification results, and analyze the effect of the number of classes on the classification. Finally, Section 6 concludes this work.

2 Motivations and Related Work

2.1 Motivations

The binary classification into positive and negative of posts collected from online web-sites, social networks, or microblogging services is an interesting approach that allows companies to estimate the level of satisfaction of users, or their expectations towards an upcoming service. However, determining whether a tweet is positive or negative might not always be sufficient.

Take the following two tweets:

- “Noooooooooo! My iPhone glass cracked :(”;
- “Damn damn.. no iPhone support for windows XP x64. There are some workarounds, but I can’t figure this out.”

The difference between these tweets, in terms of sentiment and even interpretations of what the users want, can be easily seen. Both tweets are obviously negative, but in different respects. As a matter of fact, for the company producing the product that is the subject of these tweets, the information that they can extract from each needs to be treated differently. While in the first tweet the user is expressing a sentiment of sadness because of physical damage to the product, in the second tweet the user is expressing anger and frustration due to the product’s lack of the support for a particular operating system. The company would probably be best advised to prioritize the problem raised in the second tweet; however, in general, both tweets are important in different ways, and the difference between them needs to be emphasized.

Therefore, the detection of the real sentiment within a tweet is of great importance. Gagh and Shah[4] nominated “hidden sentiments identification” as one of the most challenging tasks when performing sentiment analysis. They defined it as going beyond the identification of the polarity to the detection of the specific sentiment shown, such as hate, disgust, or anger.

While some works have tried to go beyond the binary or ternary classification of tweets, most of these have divided the positive and negative classes into subclasses that focus mainly on the intensity of the sentiment polarity (e.g., “very positive”, “positive”, “mostly positive” and “very negative”, “negative”, “mostly negative”); other works have dealt with the task of multi-class classification[5–8], but in a different context as we will describe below.

That said, the current work revolves around two main axes:

- The multi-class classification of tweets; and
- The impact of the number of classes on the classification performance.

2.2 Related work

With the growth of social network and microblogging websites, people began to openly discuss their opinions, thoughts, and even daily affairs online. This has attracted researchers to study human behaviors online, collecting and summarizing data posted by people daily. Twitter, for the reasons stated above, has attracted most of this attention. Some of the research on tweets has dealt with the form of the data, the use of slang and how these develop over the time, the use of emoticons, and the nature of tweets themselves[9, 10].

However, most of the work has dealt with the actual content of tweets. While the majority have focused on classifying tweets depending on their sentiment polarity (positive or negative), whether the topic of the tweets is a product[1], a service[2], or democratic elections[3], more advanced works have gone deeper into the classification, and focused on assessing the level of sentiment strength (e.g., "very negative", "negative", "mostly negative", "neutral", "mostly positive", "positive", and "very positive"), or even attributing sentiment intensity scores to different texts[11–13].

Nevertheless, classification into multiple sentiment classes has been the subject of multiple recent works. Lin et al.[5, 6] proposed an approach that classifies documents into reader-emotion categories. They studied the classification of news articles into different sentiment classes representing the emotions they trigger in their readers. Their work mainly differs from other literature in focusing more on what the reader would feel while reading the article rather than what the writer was feeling while writing it. Similarly, Ye et al.[7] studied the problem of emotion detection in news articles from the reader's perspective. Given the limitation of classification into single-labeled classes, they investigated a multi-label classification. Their work falls into the same category as that of Bouazizi and Ohtsuki[14] who investigated the problem of sentiment quantification, and attributed more than one sentiment class to posts extracted from Twitter. Liang et al.[8] proposed a system that recommends emoticons to users while they are typing their texts, depending on the content of what they are writing.

In the context of multi-class classification, we proposed in a previous work[15] a scalable approach that allows the classification of tweets into different sentiment classes. While our approach can be applied to any number of sentiment classes, we restricted our study to seven. The tool we developed in Ref. [15] is used here to extract features from the tweets, and Weka[16] is used to perform the multi-class classification.

3 Multi-Class Classification: Experiment Specifications

In this section, we will show the empirical results of our experiments on two data sets. Despite the fact that these are purely empirical results, we will later use them as a starting point to identify several challenges that make the task of multi-class classification difficult and, in some cases, almost impossible.

3.1 Problem statement

Given a set of tweets, we study the possibility of classifying them into different sentiment classes. From each tweet, we extract different sets of features, refer to a manually annotated training set, and use machine learning to perform the classification.

Other than the classification itself, which has been detailed in our previous work[15], we study the impact of the number of sentiment classes on the classification performance (i.e., accuracy, precision, and recall). We analyze the results of the different experiments and conclude with the limitations that make multi-class classification a difficult task.

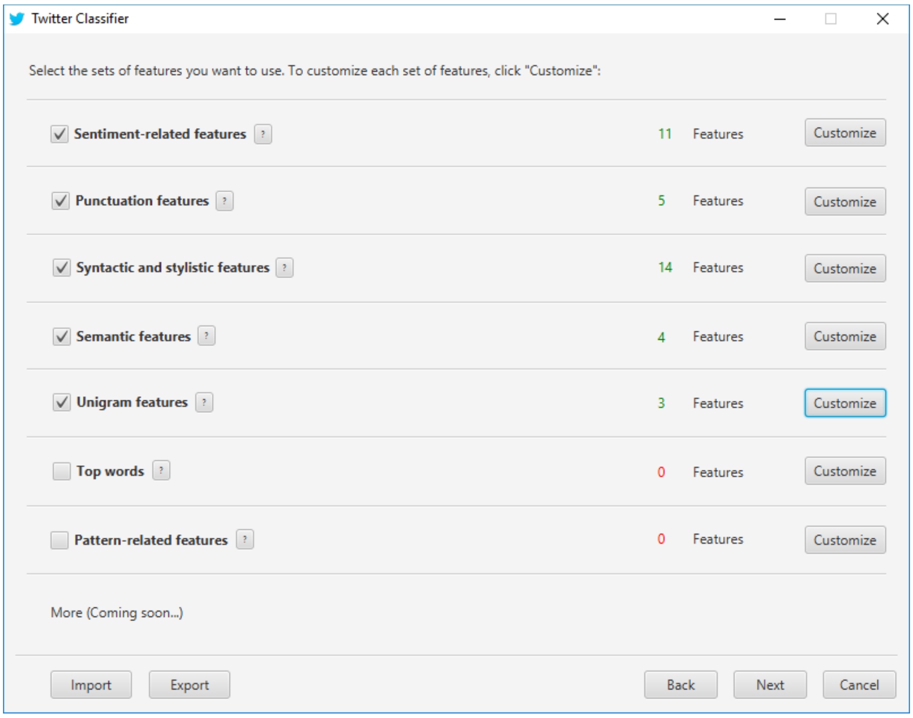
3.2 Data sets used

For our experiments, we used two data sets composed of posts extracted from Twitter that had been manually annotated into 7 different sentiment classes. The 7 different sentiments present 3 pairs of opposite sentiments (i.e., [Love vs Hate], [Happiness vs Sadness], and [Fun vs Anger]) in addition to the sentiment class [Neutral]. The structure of the data sets is given in Table 1.

We used the data sets either entirely or in part depending on the requirements of each experiment, so will explicitly mention the parts of the data set used in each case.

3.3 Features extraction

To extract the desired features from the different tweets, we used SENTA[15]. SENTA is a tool we have built that helps users to extract several types of features through a user-friendly graphical interface. In Fig. 1, we show the main window of SENTA, through which the user



Twitter Classifier

Select the sets of features you want to use. To customize each set of features, click "Customize":

☑ Sentiment-related features ? 11 Features Customize

☑ Punctuation features ? 5 Features Customize

☑ Syntactic and stylistic features ? 14 Features Customize

☑ Semantic features ? 4 Features Customize

☑ Unigram features ? 3 Features Customize

□ Top words ? 0 Features Customize

□ Pattern-related features ? 0 Features Customize

More (Coming soon...)

Fig. 1 Main interface of SENTA.

Table 1 Structure of the dataset used.

Class	Training set	Test set
Fun	3000	2643
Happiness	3000	2963
Love	3000	1945
Neutral	3000	4989
Sadness	3000	4528
Anger	3000	1558
Hate	3000	1115
Total	21 000	19 740

selects the basic sets of features he wants to extract and customize.

While SENTA offers the possibility to extract a multitude of features, we did not use all of them in this work: in this sub-section, we briefly introduce the features we did use. The detailed significance of each feature is given in Ref. [15].

Sentiment features. Sentiment features rely on the sentiment polarities of different components of the tweet. Followings are the sentiment features we extracted:

- The number of both positive and negative words;
- The number of both highly emotional positive and highly emotional negative words;
- The ratio of emotional words;
- The number of both positive and negative emoticons; and
- The number of both positive and negative slang words.

Punctuation features. With the exception of exclamation marks, punctuation does not usually reveal any sentiments explicitly; nonetheless, the excessive use of some forms of punctuation (question marks, exclamation marks, etc.) is a good indicator of the presence of a strong sentiment. Therefore, the following features are extracted:

- The number of full stops;
- The number of exclamation marks;
- The number of question marks;
- The total number of words; and
- The number of quotation marks.

Syntactic and stylistic features. These are features related to the use of words and expressions in the tweet. The following features are extracted:

- The number of particles;
- The number of interjections;
- The number of pronouns;
- The use of negation; and
- The number and use of uncommon words.

Semantic features. Semantic features are features that focus on the meanings in language or the logic inside of sentences. The following semantic features are extracted:

- The use of opinion words;

- The use of highly sentimental words;
- The use of words expressing uncertainty; and
- The use of the passive form of speech.

Unigram features. These are features collected with reference to a prebuilt dictionary containing words that are highly correlated with the different sentiment classes. In each tweet, we check whether any of the words in the dictionary are present; if so, the feature corresponding to the sentiment of that word is incremented by 1. In other words, these features count the existence of words related to each sentiment in the tweet. Therefore, 6 features are extracted (for the 6 sentiments other than Neutral). The prebuilt dictionary is the same as that used in Ref. [15].

Pattern features. Patterns are used as a complementary set of features to detect what unigrams cannot detect. In most of the cases, sentimental words are sufficient indication of the sentiment present in a sentence, whereas in other cases a person can employ some specific longer expressions to express a sentiment. Therefore, the main contribution of pattern features is to detect these longer expressions. Pattern features are extracted from the training set. They are exclusive to each sentiment polarity (i.e., if a pattern exists in two sentiments of opposite polarities, it is excluded from the lists of patterns of both sentiments). A resemblance function has also been defined to measure how close a given tweet is a pattern. As mentioned above, the procedure of the extraction of pattern features, as well as the other sets of features, is detailed in Ref. [15]. The selection of features as well as the optimization of the parameters related to them is therefore outside of the scope of this paper.

However, we will discuss pattern and unigram features in more details in a later section when we introduce our model for representing the sentiment space.

3.4 Experiment specifications

As mentioned above, our data sets contain tweets fitting into 7 sentiment classes. The sentiments taken into account are divided into 3 pairs of opposite sentiments and an additional single sentiment: [Fun vs Anger], [Love vs Hate], [Happiness vs Sadness], and [Neutral]. For convenience, in what follows, each sentiment class will be referred to by its name or by its abbreviation:

- Fun (F);
- Anger (A);
- Happiness (Hp);
- Sadness (S);
- Love (L);
- Hate (Ht); and
- Neutral (N).

We used the Random Forest classifier[17] in our experiments, and applied 4 Key Performance Indicators (KPIs) for evaluating the classification: Accuracy, Precision, Recall, and F-measure:

- **Accuracy** refers to the overall correctness of classification, measuring the ratio of correctly classified instances over the total number of instances.
- **Precision** refers to the fraction of the tweets correctly classified, for a given sentiment, over the total number of tweets classified as belonging to that sentiment.
- **Recall** refers to the fraction of tweets correctly classified, for a given sentiment, over the total number of tweets actually belonging to that sentiment. In other words, for a single sentiment, this KPI is equivalent to its Accuracy.
- **F-measure** is defined as follows:

$$\text{F-measure} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{1}$$

4 Experimental Results

To evaluate the impact of the number of classes on the classification performance, we measure the KPIs mentioned above for different numbers of sentiments.

4.1 Two sentiment classes

In our first experiment, we run the binary classification of the different pairs of sentiments, each pair apart. To recall, the sentiments are chosen so that they fit into several pairs of approximately opposite sentiments. The term *approximately* is used here to highlight the fact that, even though we treat them as pairs of opposite sentiments, this assumption is not very accurate: this is discussed in details below.

That being said, in this first round of experiments, we divide our data set into sub-sets, each contains only the tweets of a pair of sentiments. Additionally, the term “vs” used in the following in the format [*A* vs *B*], where *A* and *B* are two sentiments, means that the sentiment *A* is checked against the sentiment *B*. In other words, the classifier is trying to classify the tweets into one of the two classes *A* and *B*. The classification Accuracy, Precision, Recall, and F-measure of the binary classification of pairs of sentiments are given in Table 2.

The binary classification of the different pairs of

Table 2 Accuracy, Precision, Recall, and F-measure of the binary classification.

Class	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)
Fun	80.1	88.4	80.1	84.0
Anger	82.2	70.9	82.2	76.1
Fun vs Anger	80.9	81.9	80.9	81.1
Happiness	81.9	74.3	81.9	77.9
Sadness	81.5	87.3	81.5	84.3
Happiness vs Sadness	81.6	82.2	81.6	81.8
Love	93.8	98.9	93.8	96.3
Hate	98.1	90.1	98.1	93.9
Love vs Hate	95.4	95.7	95.4	95.4

sentiments presents good Accuracy, Precision, and Recall. All the classification tasks acheived an Accuracy higher than 80%, with the pair [Love vs Hate] having the highest (95.4%). The average Accuracy of classification is 86.0%.

4.2 Three sentiment classes

After adding the class Neutral as a third class to the same sets we used in the previous sub-section, the Accuracy of classification dropped remarkably, as shown in Table 3.

While the pair [Love vs Hate] maintained a high Accuracy, Precision, and Recall levels, the two other pairs were highly impacted by the introduction of the third class. In particular, the class Fun showed a decrease of Accuracy and Precision from 80.1% and 88.4% to 50.0% and 63.2%, respectively. This decrease will be addressed later, but, in brief, we suspect this to be due to the low number of sentimental words collected for unigram features for this sentiment, and its proximity to the class Neutral. The overall average Accuracy with Neutral added is 72.5%.

Table 3 Accuracy, Precision, Recall, and F-measure of the ternary classification.

Class	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)
Fun (F)	50.0	63.2	50.0	55.8
Neutral (N)	74.5	73.6	74.5	74.1
Anger (A)	70.9	54.0	70.9	61.3
(F) vs (N) vs (A)	66.9	67.3	66.9	66.7
Happiness (Hp)	68.2	64.0	68.2	66.0
Neutral (N)	69.3	62.5	69.3	65.8
Sadness (S)	59.2	70.7	59.2	64.4
(Hp) vs (N) vs (S)	65.4	65.8	65.4	65.3
Love (L)	82.0	75.4	82.0	78.6
Neutral (N)	84.8	92.2	84.8	88.4
Hate (Ht)	93.0	77.2	93.0	84.3
(L) vs (N) vs (Ht)	85.3	86.1	85.3	85.5

4.3 Four sentiment classes

For this set of experiments, we discarded the class Neutral and tried the different possible combinations of pairs of sentiments. For convenience, we kept only the overall classification performance for each experiment. The results are given in Table 4.

Again, the overall Accuracy, Precision, Recall, and F-measure are lower than those of the ternary classification. While the pair [Love vs Hate] acheives the highest Accuracy, the classes Happiness and Fun present low Accuracy and Recall. These two classes were confused with each other, the reason for which can easily be seen from the nature of the two classes themselves: they are quite similar to each other, with most of the sentimental words used to express happiness also used to express fun and enjoyment. The overall average Accuracy is 69.9%.

4.4 Five sentiment classes

Keeping the same combinations we used in the 4-class classification, we added the class Neutral and re-ran the classification. The results are given in Table 5.

The same observations made in the previous sub-section are present again: the sentiment Fun was rather confused with the classes Happiness and Neutral. The introduction of the new class decreased the overall average Accuracy to 61.8%.

4.5 Six sentiment classes

For this experiment, we used the entire data set, except

Table 4 Accuracy, Precision, Recall, and F-measure of the 4-class classification.

Class	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)
(F)-(A)-(Hp)-(S)	60.4	60.7	60.4	60.2
(F)-(A)-(L)-(Ht)	74.9	75.9	74.9	74.5
(Hp)-(S)-(L)-(Ht)	74.5	75.2	74.5	74.7

Table 5 Accuracy, Precision, Recall, and F-measure of the 5-class classification.

Class	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)
(F)-(A)-(Hp)-(S)-(N)	54.4	55.4	54.4	54.1
(F)-(A)-(L)-(Ht)-(N)	66.9	66.9	66.9	66.3
(Hp)-(S)-(L)-(Ht)-(N)	64.1	64.6	64.1	63.8

for the tweets of the class Neutral. The performance of the classification is given in Table 6.

The class Fun still presents the lowest Accuracy and Recall, with most of its tweets misclassified. The tweets of the class Happiness present the second lowest Accuracy and Recall. The pair of sentiments [Love vs Hate] presents the highest Accuracy and Recall due to the fact that these sentiments are easily distinguishable from each other, and also from the rest of the sentiments.

The overall average Accuracy is 60.4%, which presents no major difference from that of the classification into 5 sentiments.

4.6 Seven sentiment classes

Finally, we ran the classification using the entire data set. The performance of classification into sentiment classes is given in Table 7.

The same trend seems to hold, with the overall Accuracy of 60.2% slightly lower compared to that of the previous experiment. Again, the classes Love and Hate present the highest Accuracy.

Table 6 Accuracy, Precision, Recall, and F-measure for the 6-class classification of tweets of 6 classes.

Class	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)
Fun	39.1	56.8	39.1	46.3
Anger	59.3	52.4	59.3	55.6
Happiness	57.6	54.6	57.6	56.0
Sadness	63.9	68.6	63.9	66.1
Love	71.1	55.5	71.1	62.3
Hate	86.8	73.2	86.8	79.4
Overall	60.4	60.5	60.4	60.0

Table 7 Accuracy, Precision, Recall, and F-measure for the classification of tweets of 7 classes.

Class	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)
Fun	40.7	60.5	40.7	48.7
Anger	62.2	63.0	62.2	62.6
Happiness	54.3	58.6	54.3	56.4
Sadness	52.1	65.3	52.1	58.0
Love	75.2	62.9	75.2	68.5
Hate	90.9	80.4	90.9	85.4
Neutral	67.8	52.3	67.8	59.0
Overall	60.2	60.8	60.2	59.7

5 Analysis and Discussion of the Results

5.1 Observations

Because it is the most important indicator of good classification, we focus mainly on the level of Accuracy. For each different number of sentiment classes, the level of accuracy for the different sentiments is shown, alongside the overall Accuracy, in Fig. 2.

Obviously, classification Accuracy decreases with an increase in the number of sentiments. However, the decrease rate slows. Starting from 5 sentiment classes, Accuracy starts to be almost unchanging. While this is true for the current dataset, we cannot generalize this behavior, nor determine whether it will maintain the same rate if we continue to add more sentiment classes. We suggest that the addition of an extra pair of

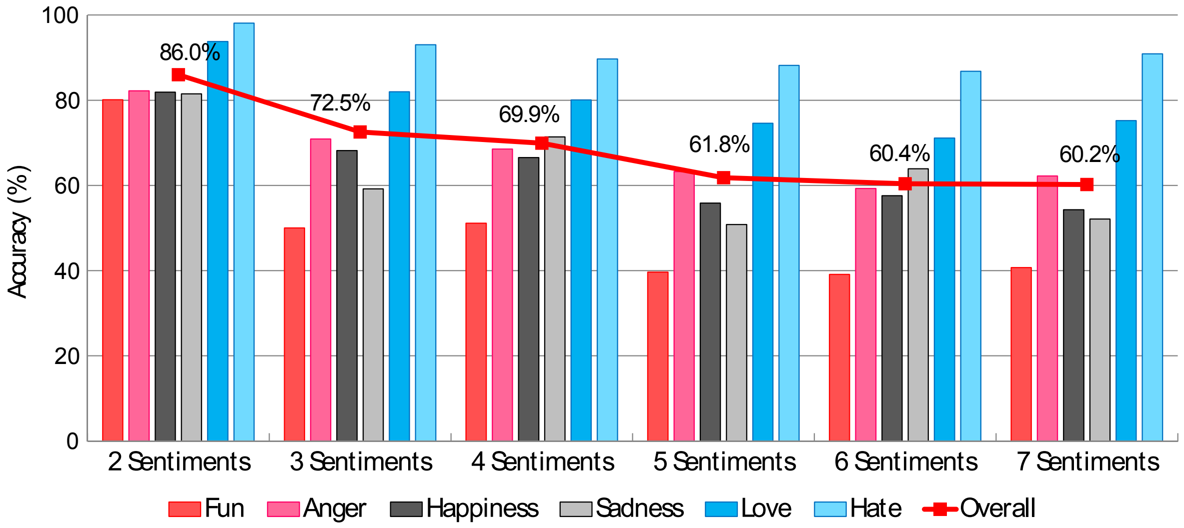


Fig. 2 Overall classification Accuracy and individual sentiment classification Accuracy for different numbers of sentiment classes.

sentiments (e.g., [Enthusiasm vs Boredom]) would help to clarify this point.

On a side note, the slight improvement in Accuracy of some sentiment classes (e.g., Fun and Anger) in the 7-class classification over that in the 6-class classification does not mean that adding a seventh class makes it easier to detect these sentiments; rather it is mainly due to how the classifier works. In other words, the classifier's rules are built so that the overall Accuracy is the highest. This can make the rules defined for 6 sentiment classes different from those of 7 sentiment classes, which results in this slight enhancement of some sentiments over others. Despite this, we believe that the overall trend still reflects the behavior of classification Accuracy as a function of the number of sentiments.

In addition, the pair of sentiment classes [Love vs Hate] seems to be the least prone to have their Accuracies decrease regardless of the number of classes, whereas sentiments such as Fun and Happiness seem to be easily confused with each other and with other sentiments, such that many of these tweets are misclassified.

5.2 Analysis

5.2.1 Sentiment space representation

At a first glance, we could imagine sentiments as defined in this work as pairs of opposite sentiments, as we initially intended. Accordingly, we could define a space with $n/2$ different dimensions, where each dimension has two ends representing the opposite sentiments. Figure 3 shows this possible representation of the sentiments for 3 pairs of sentiments (the seventh is the sentiment Neutral). Obviously, the farther a point from the origin, the stronger the sentiment is. A short text (such as a tweet), in this space, could be represented as a point, or a vector starting from the origin whose projection on each of the dimensions shows how strong it is. In the same figure, the point T_1 represents a text showing the sentiments [Happiness, Love, Fun], while the point T_2 represents a text showing only the sentiment Hate, and the point T_3 represents a Neutral text.

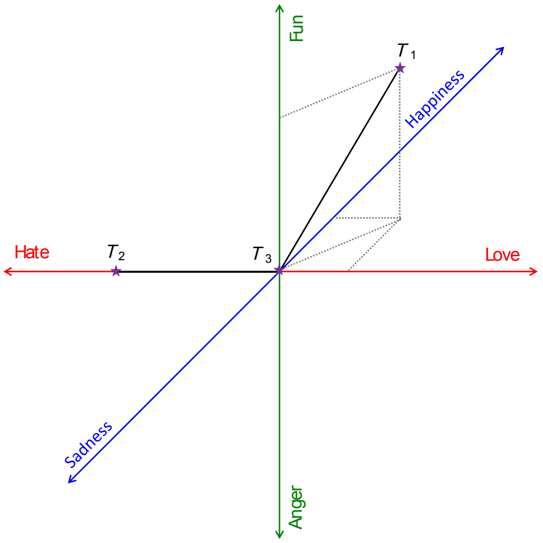


Fig. 3 Representation of the sentiment space.

However, in practice, and based on our observations on the data set, this representation has several flaws. One flaw is that it suggests that the dimensions are orthogonal. This is not always true, because some sentiments are highly correlated and are not sufficiently independent from each other to be considered orthogonal, as we discuss below. Also, the class Neutral in this representation is restricted to an infinitesimal region near the origin.

A more reasonable and practical way to represent the sentiments in a given space is to have each sentiment represented by a cloud centered on a specific point. This is more natural as it suggests the texts are by default neutral, unless they are in or near the given region of a particular cloud (which represents a sentiment). In addition, the dimensions in this space could represent any information, and does not need to be sentiment related. In Fig. 4, we show an example of this representation in a 2-dimensional space. Some sentiments are obviously close to each other such as sentiments S_2 and S_3, and therefore share a common area in the space.

However, in such a representation, it is not clear

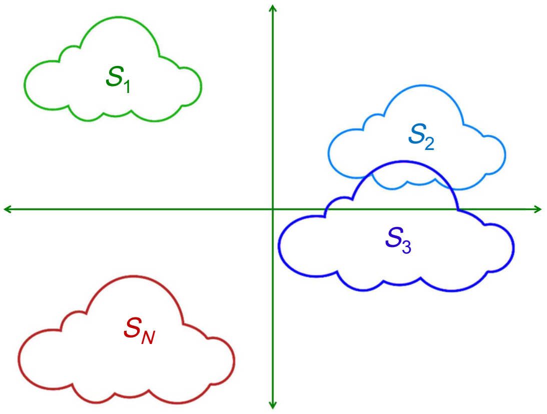


Fig. 4 Representation of the sentiment space in a 2-dimensional space.

how a given text could be presented in such a space. In addition, the cloud representation does not give an accurate description of where the sentiment is at its strongest. For these reasons, the representation is slightly modified in the current work as follows: a cloud is denser near the center and fades away as we get farther from it. In other words, a text located at the edges of the cloud shows less of the sentiment.

More importantly, this representation could allow us to define what we call the distance between two sentiments. Unlike the case of multi-dimensional representation, sentiments here can be correlated, and it is possible to define metrics to measure the distance between any two sentiments, such as the distance between the centers of the two corresponding clouds. In this work, we will refer to a cloud corresponding to a given sentiment S_i as Ω_i.

Given two different sentiments, S_i and S_j, they each could share some resemblance, through similar patterns or expressions, or a set of words that can be used for either of them. The word "fun" in the expression "@user I'm having soo much fun here!", for example, shows sentiments of Fun and Happiness.

5.2.2 Distance between two sentiments

A simple way to define the distance between two sentiments S_i and S_j is as follows: suppose there is a set of words, expressions or patterns that are commonly used to show each of the two. We will refer to the number of words, expressions or patterns that are used to express S_i as N_i, and those that are used to express S_j as N_j. The two sentiments share n words, expressions or patterns to express them (e.g., the word "upset" could be used to show both Anger and Sadness). The distance between the two sentiments could be expressed as follows:

$$D(S_i, S_j) = 1 - 2 \times \frac{n}{N_i + N_j} \qquad (2)$$

The distance is maximal (i.e., equal to 1) when the two sentiments share nothing in common, and is minimal (i.e., equal to 0) when they are identical. This representation is efficient but does not faithfully reflect how we defined the sentiment clouds, as there is no way to tell whether or not a point is close to the center of the cloud.

Thankfully, in the particular case of words (i.e., unigrams), we could derive an even more precise and meaningful expression for the distance. To recall, unigrams are simple words that are extracted in the context of unigram Features using SENTA. SENTA extracts unigrams as follows:

Step 1. For each sentiment, the user defines a small set of words that he judges as highly correlated with the given sentiment;

Step 2. SENTA refers to WordNet to extract the hyponyms of the words defined by the user and adds them to the list;

Step 3. SENTA extracts the hyponyms of the new words and adds them to the list, keeping a single copy of each word; then

Step 4. SENTA keeps repeating Step 3 several times according to the parameters set by the user.

The final list of words for a given sentiment will have the following format:

$$U(S_i) = \{w_1, w_2, \cdots, w_{n_i}\} \qquad (3)$$

However, the words that have been added manually by the user are more trustworthy and more likely to be highly correlated with the sentiment than the ones that are extracted later on. This is because hyponyms lose part of the meaning of their hypernyms as explained in Ref. [14].

In the following, we will suppose that we keep track of the depth at which each unigram is found for the first time. So words that have been introduced by the user are considered to have been found at depth 0, whereas words that are hyponyms of the words introduced by the user are considered to have been found at depth 1, and so on.

In this context, the unigrams of a given sentiment could be seen as a cloud with several layers as shown in Fig. 5, where unigrams closer to the center of the cloud are ones extracted at an earlier stage (i.e., having a lower depth value). At the very center of the cloud are the words that are used to name the sentiment along with their direct derivations (e.g., for the class Happiness,

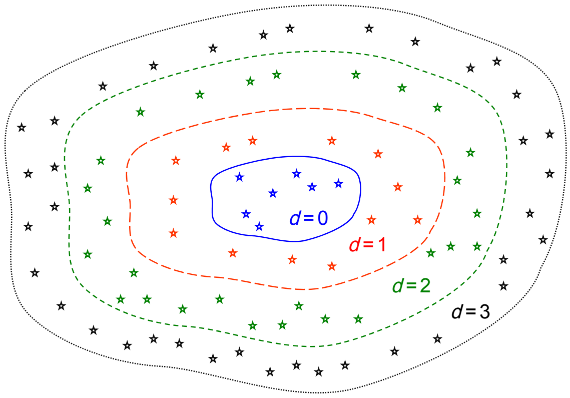


Fig. 5 Multiple layers of a single cloud of a given sentiment.

these are “happiness”, “happy”, etc.).

Following the same logic, we could also represent two sentiments in the same space as two clouds sharing some of their unigrams, as shown in Fig. 6.

With that being said, given the sentiment S_i, we will refer to the maximum depth selected by the user as $d_{\max}$, a given depth as a or b, and $N_{(i,d)}$ will equal the number of new words added to the sentiment S_i at the depth d. The seed words are those that have a depth equal to 0.

Therefore, returning to the definition of the distance between two sentiments, we express it as follows:

$$D(S_i,\, S_j) = \sum_{a=0}^{d_{\max}} \sum_{b=0}^{d_{\max}} \delta_{(a,b)} \times \left(1 - 2 \times \frac{n_{(a,b)}}{(N_{(i,a)} + N_{(j,b)})}\right) \tag{4}$$

where $n_{(a,b)}$ is the number of common unigrams of the sentiments S_i at the layer a and S_j at the layer b, and $\delta_{(a,b)}$ is a coefficient highlighting the weight of the common unigrams between two different layers (a and b) of the two clouds. Obviously δ is symmetric (i.e., $\delta_{(a,b)} = \delta_{(b,a)}$), and all of the coefficients $\delta_{(a,b)}$ should sum up to 1.

5.2.3 Correlation between different sentiments

Now that the distances between the clouds are defined, we define the question **(Q1)**: “Is it possible to identify which sentiments are more likely to co-occur or to be highly correlated?”. The short answer for this question is *“yes”*. However, below we realistically measure the distances between sentiments in our data set, and identify which sentiments are likely to co-occur within a tweet.

Another interesting output of the current representation of sentiments is that, given an expression (or a unigram in this case), we can also tell how far it is from each cloud and what sentiment it conveys. While we have limited our study in this paper to unigrams, it is always possible to extend it to longer n-grams, patterns, or even full sentences. This leads us to our next question **(Q2)**: “Given a sentence (i.e., a tweet in our case), is it possible to attribute different scores to show the distance the sentence has from the sentiment?”, which can be reformulated into **(Q2’)**: “Is it possible to attribute different scores showing the strength of each of the sentiments within the sentence?” This can be simply seen as representing the sentence by a point in the space introduced above, where the closer that point is to a cloud, the stronger the sentiment corresponding to the cloud is in the sentence. In other words, the score can be any increasing function of the inverse of the distance.

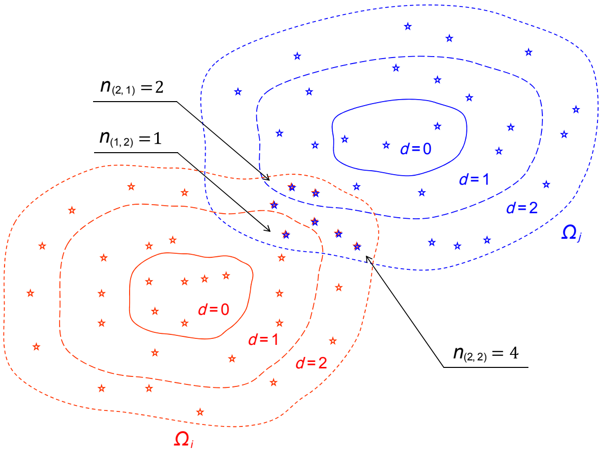


Fig. 6 Intersection between two clouds with several layers.

In the current work, we briefly introduce the concept of quantification, which we explain in more detail elsewhere. By quantification, we refer to the attribution of sentiment scores to a given text, where each score represents how strongly the sentiment is present in the text. The scores are rarely equal to 0, so we define a certain threshold T_L below which a sentiment score is considered too low, and the corresponding sentiment is thereby considered non-existent or negligible. That being said, in the current work, given a tweet T, and a set of N sentiments $S_1, S_2, \cdots, S_N$, we extract 2 different sentiment scores for each of these sentiments using the two sets of features qualified as unigram features and pattern features, as explained in Ref. [15] and which we will refer to as “unigram score” (s^u) and “pattern score” (s^p), respectively.

In the case of unigram scores s^u, they are generated simply by counting the number of unigrams generated by SENTA for each sentiment present in the tweet.

As for pattern scores, these are computed slightly differently: SENTA, as explained above, allows for extracting writing patterns from the training set (or eventually any manually annotated set, which we will be referring to as the “pattern set”) that are unique to each sentiment. These patterns could have different lengths. Given a tweet T and a pattern p extracted from the pattern set for a sentiment S_i and whose length is equal to L_j (i.e., the j-th length), we have used the following resemblance function defined in previous works[14, 15, 18].

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

Patterns of different lengths and for different sentiments are saved into different lists. We then have defined a certain number of features we qualified as "pattern features", each in the following format:

$$F_{ij} = \sum_{k=1}^{\text{knn}} \text{res}(p_k, T) \tag{5}$$

where p_k are patterns that most resemble the tweet T, and knn is a parameter referring to the number of patterns to be considered. These features are used to attribute a pattern score: suppose that we have set the minimal pattern length to $L_{\min}$ and the maximal pattern length to $L_{\max}$. We will refer to $M = L_{\max} - L_{\min}$ as the number of lengths. The pattern score s_p will be defined as follows:

$$s_p = \sum_{j=1}^{M} \Big(\beta_j \cdot \sum_{k=1}^{\text{knn}} \text{res}(p_k, T)\Big) \tag{6}$$

where β_j is a weight given to each length. Obviously, the longer the pattern is, the more important its weight should be.

Using both the unigram scores and the pattern scores, we can attribute scores showing the strength of the different sentiments within a tweet. However, this step falls outside of the scope of the current paper, in which our main goal is to model sentiments in way that makes it possible for a given text to have multiple sentiments, and to measure the distance between the text and a given sentiment, as well as the distance between different sentiments.

In the current work, we have used both unigram scores and pattern scores to define the distance between the different sentiments. We will use Eqs. (2) and (4) to measure the distances between sentiments using pattern scores and unigram scores, respectively.

In particular, regarding Eq. (4), it is important to mention that we have restricted our extraction of unigrams to a maximum depth $d_{\max} = 3$. Without loss of generality, we define and will be using the values of the different combinations of a and b shown in Table 8.

The distance measures between the different sentiment classes will be referred to as D_U and D_P for unigrams and patterns, respectively. For our data set, these distances are displayed in Tables 9 and 10.

As expected, and under both metrics, the class Fun has the smallest distance to the class Happiness. Especially when using the metric D_U, these two classes have by far the smallest distance. This means that these two sentiments have a lot in common, and therefore can be easily confused. In addition, using the metric D_P with reference to the class Neutral, the class Fun has a relatively small distance compared with all other sentiments.

It is also noticeable that, overall, the positive sentiments have a smaller distance from one another,

Table 8 Values of $\delta(a, b)$ for different depths.

(a, b)	$\delta(a, b)$
$(0, 0)$	1/2*
$(0, 1), (1, 0)$	1/8
$(1, 3), (2, 2), (3, 1)$	1/24
$(1, 4), (2, 3), (3, 2), (4,1)$	1/64
$(2, 4), (3, 3), (4, 2)$	1/96
$(3, 4), (4, 3), (4, 4)$	1/128

Note: *To make sure that all the coefficients sum up to 1, this coefficient is set to be equal to 65/128 instead.

Table 9 Distance between the different sentiments as measured with D_U.

	(F)	(Hp)	(L)	(N)	(A)	(S)	(Ht)
(F)	0	0.61	0.85	–	1	1	1
(Hp)	0.61	0	0.79	–	1	1	1
(L)	0.85	0.79	0	–	1	1	1
(N)	–	–	–	0	–	–	–
(A)	1	1	1	–	0	0.83	0.71
(S)	1	1	1	–	0.83	0	0.84
(Ht)	1	1	1	–	0.71	0.84	0

Table 10 Distance between the different sentiments as measured with D_P.

	(F)	(Hp)	(L)	(N)	(A)	(S)	(Ht)
(F)	0	0.95	0.94	0.98	1	1	1
(Hp)	0.95	0	0.95	0.99	1	1	1
(L)	0.94	0.95	0	0.99	1	1	1
(N)	0.98	0.99	0.99	0	0.99	0.99	0.99
(A)	1	1	1	0.99	0	0.96	0.97
(S)	1	1	1	0.99	0.96	0	0.96
(Ht)	1	1	1	0.99	0.97	0.96	0

compared to that of the negative ones. This translates into a lower Accuracy and Precision for positive sentiments than negative ones.

5.3 Discussion

From our observations and analysis, we can confirm that the task of multi-class sentiment analysis presents many challenges. To begin with, the presence of multiple classes, in general, makes it harder for a given classifier to define the borders between different classes. Moreover, in the case of text sentiment analysis, different sentiments have much in common, and the actual border between two sentiments, examplified by Happiness and Fun, is somewhat unclear. In other words, it is sometimes difficult even for humans to detect the difference. In addition, the more classes there are, the less patterns can be extracted for an individual class. Nevertheless, some sentiments can coexistent, and a certain sentence can contain more than one sentiment. Given the following tweet: *"Man, I'm having sooo much fun here. Glad my whole family came with me. It's just amazing!"*, the author explicitly presents enjoyment and happiness. This makes it hard to attribute the tweet to one sentiment class.

This leads to an important conclusion: even though many texts can be classified into one of multiple sentiment classes, it might be a more interesting task to detect all of the sentiments that exist in a tweet, and to attribute a certain score to each sentiment class, reflecting its weight.

5.4 Multi-class classification: Challenges

To recapitulate, here we list the main challenges that make multi-class sentiment analysis difficult. We illustrate with tweets from our data set that have been misclassified and explain the reasons for the misclassification.

Presence of negation. Handling negation has always been an issue when it comes to sentiment analysis. Not only is it hard to tell whether the presence of negation is a polarity switcher or not, but also, in the case of multi-class classification, switching polarity does not automatically indicate that the sentiment of the tweet is the opposite of that negated. This can be seen in the following tweet: "Well guess what?? I'm not really happy with what he said anyway!" The word "happy" is a word that is used usually to express sentiments of Happiness. On the other hand, as stated in the previous subsections, Happiness and Sadness are supposedly a pair of opposite sentiments. However, the negation in this tweet did not show the sentiment of Sadness which has been reported by the classifier, but rather the sentiment of Anger.

Context dependency. Tweets are often intended as replies to other tweets, making them highly context dependent. We read the tweet *"I remember someone saying it's gonna be fun.."* as a Neutral tweet, but some of the annotators labelled the tweet as showing sentiments of Anger. This is because they assumed the user is showing dissatisfaction towards an event that was supposed to be funny, but in actual fact was not. However, while this assumption can be made by a human, machines are not able to imagine such scenarios and extract the actual sentiment out of it.

Polysemy. Several words in English, as with other languages, have multiple meanings depending on their context. These meanings could be similar or totally unrelated. However, for multi-class sentiment analysis, even the similar meanings could indicate different emotions. An example is the word "mad", the meanings of which include angry as well as crazy. Furthermore, craziness often points to something being good or funny. *"Mad"* can also be used as an adverb meaning "very", as can be seen in the following tweet: "It was mad fun man!" This tweet was classified as showing sentiments of Anger, despite the presence of two sentimental words. However, the tweet could have easily been detected as belonging to the class Fun if the PoS-Tagger could identify the word "mad" as an adverb.

Presence of multiple sentiments. Even though tweets are short in length and limited to a certain number of characters (i.e., 140 characters per tweet), they can be poly-sentimental in the sense of containing more than just one sentiment. As a matter of fact, a large number of the tweets we have in our data set present multiple sentiments, as illustrated with these tweets:

- "I'll miss you sooo much! I can't believe you have to leave.. love you!!" This tweet shows sentiments of Sadness and Love.
- "Damn it.. This guy behind me just ruined the movie for me. I hate people talking in the cinema. Idiots!!" This tweet shows sentiments of Anger and Hate.

That being the case, it is quite difficult to identify all existing sentiments present in a few words, let alone detect which one is predominant. Several tweets that have been misclassified present multiple sentiments, and the classifier had difficulty determining the predominant one.

Closeness between different sentiments. This has been discussed in the previous sub-section. Sentiments such as Happiness and Fun or Anger and Hate are largely similar, and tweets of one of each pair could easily be misclassified as being of the other. Along with context dependency, this is probably the major cause of misclassification.

Absence of sentiment indicators. As stated above, tweets are short in length, and sometimes it is hard to extract useful information from them, or even find a common pattern that makes similar sentences show the same emotion. This has led, in the case of 7-class classification, to the misclassification of many tweets as Neutral (i.e., a low Precision of the class Neutral), as well as the misclassification of tweets with sentiments of the same polarity or even of different polarities. For example, the tweet *"Dead sure it was. invite me again anytime soon!"* was annotated as being of the class Happiness but classified as being of the class Sadness.

6 Conclusion

In this paper, we studied the task of multi-class sentiment analysis. We evaluated the evolution of various KPIs as the number of sentiment classes increased. We analyzed the difficulties of, and the different challenges involved with, multi-class classification, and proposed some metrics to measure the distance between sentiments (i.e., how similar they are to one another). We concluded that, even though the task of multi-class analysis is important, it might be more interesting to perform a sentiment detection task through which all of the sentiments present within a text are extracted.

References

[1] 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. Information*, *Intelligence*, *Systems and Applications*, Chania, Greece, 2014, pp. 94–97.

[2] U. R. Hodeghatta, Sentiment analysis of Hollywood movies on Twitter, in *Proc. 2013 IEEE/ACM Int. Conf. on Advances in Social Networks Analysis and Mining*, Niagara Falls, Canada, 2013, pp. 1401–1404.

[3] J. M. Soler, F. Cuartero, and M. Roblizo, Twitter as a tool for predicting elections results, in *Proc. 2012 IEEE/ACM Int. Conf. on Advances in Social Networks Analysis and Mining*, Istanbul, Turkey, 2012, pp. 1194–1200.

[4] K. Ghag and K. Shah, Comparative analysis of the techniques for sentiment analysis, in *Proc. 2013 Int. Conf. on Advances in Technology and Engineering*, Mumbai, India, 2013, pp. 1–7.

[5] K. H. Y. Lin, C. H. Yang, and H. H. Chen, What emotions do news articles trigger in their readers? in *Proc. 30th Annu. Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, Amsterdam, Netherlands, 2007, pp. 733–734.

[6] K. H. Y. Lin, C. H. Yang, and H. H. Chen, Emotion classification of online news articles from the reader's perspective, in *Proc. 2008 IEEE/WIC/ACM Int. Conf. on Web Intelligence and Intelligent Agent Technology*, Sydney, Australia, 2008, pp. 220–226.

[7] L. Ye, R. F. Xu, and J. Xu, Emotion prediction of news articles from reader's perspective based on multi-label classification, in *Proc. 2012 Int. Conf. on Machine Learning and Cybernetics*, Xi'an, China, 2012, pp. 2019–2024.

[8] W. B. Liang, H. C. Wang, Y. A. Chu, and C. H. Wu, Emoticon recommendation in microblog using affective trajectory model, in *Proc. 2014 Asia-Pacific Signal and Information Processing Association Annu. Summit and Conf.*, Chiang Mai, Thailand, 2014, pp. 1–5.

[9] 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. on Research and Development in Information Retrieval*, Geneva, Switzerland, 2010, pp. 841–842.

[10] 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. 2013 Int. Conf. on Social Computing*, Alexandria, VA, USA, 2013, pp. 345–350.

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

[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. 2015 IEEE Int. Conf. on Electrical*, *Computer and Communication Technologies*, Coimbatore, India, 2015, pp. 1–9.

[13] R. Srivastava and M. P. S. Bhatia, Quantifying modified opinion strength: A fuzzy inference system for Sentiment Analysis, in *Proc. 2012 Int. Conf. on Advances in Computing*, *Communications and Informatics*, Mysore, India, 2013, pp. 1512–1519.

[14] M. Bouazizi and T. Ohtsuki, Sentiment analysis: From binary to multi-class classification: A pattern-based approach for multi-class sentiment analysis in Twitter, in *Proc. 2016 IEEE Int. Conf. on Communications*, Kuala Lumpur, Malaysia, 2016, pp. 1–6.

[15] M. Bouazizi and T. Ohtsuki, A pattern-based approach for multi-class sentiment analysis in twitter, *IEEE Access*, vol. 5, pp. 20617–20639, 2017.

[16] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, The WEKA data mining software: An update, *ACM SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, 2009.

[17] L. Breiman, Random forests, *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[18] D. Davidov, O. Tsur, and A. Rappoport, Semi-supervised recognition of sarcastic sentences in Twitter and Amazon, in *Proc. 14th Conf. on Computational Natural Language Learning*, Uppsala, Sweden, 2010, pp. 107–116.

Tomoaki Ohtsuki received the BEng, MEng, and PhD degrees in electrical engineering from Keio University, Yokohama, Japan, in 1990, 1992, and 1994, respectively. He is now a professor at Keio University. He has published more than 140 journal papers and 340 international conference papers. He served a chair of IEEE Communications Society, and Signal Processing & Communications and Electronics Technical Committee. He served a technical editor of *IEEE Wireless Communications* and an editor of *Physical Communications*. He is now serving an area editor of *IEEE Transactions on Vehicular Technology* and an editor of *IEEE Communications Surveys and Tutorials*. He has served general co-chair and symposium co-chair of many conferences, including IEEE GLOBECOM 2008, SPC, IEEE ICC2011, CTS, IEEE GCOM2012, SPC, and IEEE SPAWC. He gave tutorials and keynote speech at many international conferences including IEEE VTC, IEEE PIMRC, and so on. He was a vice president of Communications Society of the IEICE. He is a senior member of the IEEE and a fellow of the IEICE. He is engaged in research on wireless communications, optical communications, signal processing, and information theory.

Mondher Bouazizi received the BS degree from Carthage University in 2010. He worked as a telecommunication engineer (access network quality and optimization) for 3 years at Ooredoo Tunisia. In 2015, he enrolled as a master student at Keio University and obtained the master degree in 2017. He is currently a PhD student at Keio University. He is engaged in research on the applications of machine learning and deep learning for social network analysis, natural language processing, and data mining.