Leveraging Social Media Linguistic Features for Bilingual Microblog Sentiment Classification

Konstantinos Tsamis
Computer Engineering & Informatics Department
University of Patras
Rio, Greece
ktsamis@ceid.upatras.gr

Andreas Komninos
Computer Engineering & Informatics Department
University of Patras
Rio, Greece
akomninos@ceid.upatras.gr

John Garofalakis
Computer Engineering & Informatics Department
University of Patras
Rio, Greece
garofala@ceid.upatras.gr

***Abstract*—Social media and microblogs have become an integral part of everyday life. People use microblogs to communicate with each other, express their opinion about a wide range of topics and inform themselves about issues they are interested in. The increasing volume of information generated in microblogs has led to the need of automatically determining the sentiment expressed in microblog comments. Researchers have worked in systematically analyzing microblog comments in order to identify the sentiment expressed in them. Most work in sentiment analysis of microblog comments has been focused on comments written in the English language, whereas fewer efforts have been made in predicting the sentiment of Greek microblog comments.**

In this paper, we propose a lexicon-based sentiment analysis algorithm for the sentiment classification of both Greek and English microblog comments. The proposed method uses a unified approach for determining the sentiment of comments written in both languages and incorporates techniques that exploit the distinctive features of the language used in microblogs in order to accurately predict the sentiment expressed in microblog comments. Our approach achieves promising results for the sentiment classification of microblog comments into positive, negative or neutral.

***Index Terms*—social networks, emotion recognition, sentiment analysis, natural language processing**

I. Introduction

The rapid growth of the Web 2.0 technologies has led to the widespread use of microblogs and social media by users who desire to express their opinion about a vast variety of topics. User comments range from discussions about current political and economic issues to product reviews and opinions about locations or points of interest. Microblogs differ from traditional blogs due to the limitations that are imposed upon the maximum character length of a comment. For example, Twitter has a limit of 280 characters while Foursquare has a limit of 200 characters.

The large volume of opinionated text that is available online in the form of microblog comments, provides users easy access to information about a subject or an entity that they are interested in. An increasing number of users read product reviews in order to decide what product to buy, consult users opinions about locations and points of interests prior to visiting them or try to inform themselves about currents affairs by reading relevant comments in microblogs. On the other hand, companies try to use the feedback that comes from user comments in order to improve their products or services and fine-tune their marketing strategies.

However, the abundance of information that is available online makes difficult for users to form an opinion about a subject that interests them. Users often don't have the required time to go through a sufficient number of comments in order to form a comprehensive opinion. Furthermore, frequently comments contain conflicting opinions about a subject.

These issues have led to the development of various methods and techniques for the systematic analysis of online user comments and reviews with the aim of extracting useful information that can help users to form a comprehensive opinion without having to go through the whole volume of available comments. A common approach in this field of research involves the use of Sentiment Analysis for determining the sentiment that is expressed in user comments towards a subject or an entity.

Sentiment analysis is defined as the computational treatment of opinion, sentiment and subjectivity in text [1]. The main purpose of sentiment analysis is to determine the sentiment polarity of user opinions expressed in a piece of text. Sentiment can be classified into positive, negative or neutral. Some sentiment analysis algorithms aim to determine also the strength of the sentiment expressed in the text.

Sentiment analysis can be categorized into the three following categories according to its scope of application; document-level, sentence-level and feature-level. Document-level sentiment analysis deals with discovering the overall sentiment that is expressed in the whole text as opposed to sentence-level sentiment analysis that aims to determine the sentiment expressed in each sentence of the text. Feature-level sentiment analysis refers to determining the sentiment expressed on different features of entities (e.g the screen or the camera of a smartphone).

Sentiment analysis of microblog and social media comments poses different challenges compared to sentiment analysis of traditional long form text. The language used in microblogs is generally informal and, due to the character length limitations, users try to enclose as much information as possible in a short piece of text. This leads to frequent grammar and spelling mistakes as well as the use of abbreviations. Moreover, microblog users tend to use special forms of language for their exchanges such as the frequent use of emoticons and punctuation signs. These sort of challenges have led researchers into developing sentiment analysis methods focused on the microblog domain, that try to use the innate characteristics of microblog language in the process of identifying the sentiment expressed in microblog comments.

II. Related Work

A. Sentiment detection techniques

Most existing work in sentiment analysis of microblog comments can be categorized into two main approaches: supervised learning methods and unsupervised learning methods [1]. Supervised methods are based in the training of a classifier for the sentiment classification task. For the training of the classifier established machine learning algorithms such as Support Vector Machines, Naïve Bayes, Multinomial Naïve Bayes and Maximum Entropy are used [2]. The features employed include word unigrams [3]–[5], bigrams [3] or n-grams [6]–[9], character n-grams [9], [10], part-of-speech features [5], [7]–[9], lexicon features [5], [7]–[9] as well as microblog-specific features [5]–[7], [9].

The majority of unsupervised learning approaches to the problem of sentiment analysis in microblogs use sentiment lexicons, lexical resources that contain words or phrases annotated for the sentiment they express, in order to determine if the text contains terms that express sentiment. The overall sentiment expressed by a piece of text is a function of the sentiment expressed by the terms contained in the text.

Sentiment lexicons can be created manually or automatically by using a list of seed words with predefined sentiment polarity and propagating their sentiment labels by exploiting semantic relations among words using lexical resources, such as WordNet [11], or by using mutual information based approaches [12]. The work in [13] compared several lexical resources for the task of sentiment analysis in microblogs, with the most promising results accomplished for the SentiWordNet lexicon [14] and the MPQA Subjectivity Lexicon [15].

Most state-of-the-art lexicon-based methods try to leverage the information that emerges from the use of colloquial language such as emoticons, punctuation signs, hashtags and slang in order to predict the sentiment polarity of the text [16]–[18]. Furthermore, a common practice in lexicon-based methods is the use of linguistic processing such as modifiers detection [16], [17], negation handling [13], [16], [17], [19] and part-of-speech tagging [19].

One problem with microblog texts (such as user reviews or tweets) is that since these texts are typically short, it may be hard to classify them for sentiment since they do not contain enough "sentiment-weighted" words to yield an accurate result. To circumvent the problem, researchers have proposed the use of special lexicons that leverage the unique linguistic characteristics of microblog texts. A lexicon based approach for the sentiment polarity classification and sentiment strength detection of short informal online text that incorporates the use of microblog-specific language, modifiers and negation handling was proposed by [16]. A similar method was proposed in [20] for the ternary classification of social media comments into positive, negative or objective comments. Except from traditional sentiment polarity classification, work has been done in clustering microblog comments based on the emotion they express compared to a set of eight primary emotions [18].

B. Sentiment detection in Greek microblog texts

There is little work done in sentiment analysis that focuses on the sentiment classification of Greek text compared to the volume of work available for the English language. A supervised learning method was proposed by [21] for the sentiment classification of Greek hotel reviews using a unigram language model. Another unsupervised approach presented in [22] used lexicon features and word embedding-based features to train an SVM classifier for the task of predicting the sentiment expressed in Greek and English online reviews. A similar approach was proposed by [23] for the sentiment classification of Greek tweets. Finally, in [24], an approach evaluating several machine-learning methods for the sentiment polarity classification of Greek news article user comments was presented. The authors augmented the original feature space (TF-IDF vectors of tokenized comments) by automatically translating texts to English and adding the resulting tokens into the feature space. The approach yielded an F-score upwards of 80% using Gradient Boosted machines, showing that unsupervised learning methods can be successful in categorising user-contributed texts.

The work in [25] focused on studying the impact of political Greek tweets on the Greek general election by using an unsupervised approach for identifying the sentiment expressed in Greek tweets. A lightweight lexicon-based algorithm was presented by [26] for the sentiment classification of Greek microblog venue reviews. A lexicon-based method was proposed by [27] for rating the sentiment of Greek tweets and hashtags against the set of six basic emotions; anger, disgust, fear, happiness, sadness, surprise.

Previous works, especially relating to Greek texts, have a number of shortcomings. First, the process for obtaining a labelled dataset to use as a baseline comparison is subject to bias, since it typically employs very few manual coders (e.g., just two in [22], [27]). Other works such as [21], [22] use the overall rating of a venue as an indicator of sentiment polarity, which may not always be reasonable, e.g. a user giving a bad score to a venue (e.g. "1 star"), but commenting positively on one or two features of the venue that they actually liked (e.g. "Breakfast was good"). Further, where sentiment detection is done on microblog texts (e.g. Twitter), some approaches (e.g. [23] are artificially easing the task by aggregating documents under topics (Twitter hashtags) and performing sentiment detection on the entire hashtag, rather than individual documents. Finally, as far as we are aware, there is no work that combines the use of techniques such as colloquial language detection, modifiers detection and negation handling in a lexicon-based approach for the sentiment classification of Greek text. The main motivation for our research was to investigate the use of state-of-the-art techniques for sentiment analysis of comments written in the Greek language in an unsupervised manner while also applying the same techniques for classifying English comments.

In this paper, we present a lexicon-based method for the sentiment classification of microblog comments written in both the Greek and English language. Our method incorporates the use of the colloquial language characteristics of microblog text such as emoticons and punctuation signs in order to accurately predict the sentiment expressed in a comment. Negation handling and modifier detection are implemented for capturing the effect that these linguistic features have in the sentiment expressed in the text. Spellchecking and Greeklish handling is supported for handling the informal nature of microblog comments. To compare this approach's performance, we also employ machine-learning sentiment detection algorithms. Our work is done on a small dataset of user venue reviews, however, this dataset was annotated by 148 users, with each review annotated by an average of 19 users.

III. The SENTI Algorithm

The sentiment analysis algorithm we developed (Senti) proposes a unified approach to discovering the sentiment that is expressed in microblogging comments written both in the Greek and the English language. For that purpose, we use two different sentiment lexicons which contain terms evaluated by human judges for their sentiment polarity, subjectivity and part of speech; the Greek Sentiment Lexicon [28] and the MPQA Subjectivity Lexicon [15]. The Greek Sentiment Lexicon is the only freely available Greek lexical resource for the sentiment analysis task and the MPQA Subjectivity Lexicon was chosen as one of the commonly used English sentiment lexicons that provide also a subjectivity rating.

The main task of the algorithm is to represent a piece of text as a list of sentiment-weighted terms from which the text consists of, in order to determine overall sentiment expressed in the text. Sentimental terms are words that express sentiment and can be found in the sentiment lexicons. After some basic text preprocessing that involves removal of URLs from the text, text is split into tokens based on spaces and punctuation symbols. Then we search the sentiment lexicons to find out if the text contains sentimental terms.

A sentimental term is characterized by its sentiment polarity (positive, negative or neutral), degree of subjectivity (strongly subjective, weakly subjective, objective) and part of speech. Each sentimental term is assigned an initial sentiment score that measures the strength of the sentiment expressed by the term and is based on the term's degree of subjectivity.

In order to determine the sentiment polarity of a comment c, the algorithm computes a vector $V(c)$ containing three different sentiment scores S_p, one for each polarity; the positive (S_{pos}), negative (S_{neg}) and neutral (S_{neu}) overall sentiment score of the comment.

$$V(c) = [S_{pos}, S_{neg}, S_{neu}] \tag{1}$$

The sentiment score S_p for each sentiment polarity p (i.e. the vector elements) is calculated using the scores of the sentimental terms of the respective polarity. Let k_p be the number of terms of a given polarity p (positive, negative or neutral), N the total number of sentimental terms in a comment, $t_1, \ldots, t_{k_p}$ the positive, negative or neutral sentimental terms and $score(t_i)$ the sentiment score of the term t_i. Each of the scores in the final vector is calculated using the following equation:

$$S_p = \frac{k_p}{N} \sum_{i=1}^{k_p} score(t_i) \tag{2}$$

The sentiment polarity of the comment is the polarity that exhibits the greatest sentiment score S_p.

In this calculation, each term t_i can be either a textual term (i.e. a word), or a special term (an emoticon, or a pattern of repeated punctuation).

A. Special term replacement

In the text pre-processing we tokenize each document and replace any special terms found with special tags, using regular expressions. Emoticons are replaced with "posemoji" or "negemoji", based on their semantics. Repeated punctuation is replaced with "multiex" (exclamation marks), "multiq" (question marks) or "multidot" (periods). We also replace negator words (e.g. "no", "not", "don't", "wasn't") with the tag "not".

B. Scoring textual terms

Taking into consideration previous research that suggests subjective terms tend to express stronger sentiment [citation], we assign an initial score of 1.5 to strongly subjective terms, while weakly subjective or objective terms are assigned an initial score of 1. Furthermore, because of the correlation found between the presence of adjectives in a sentence and the subjectivity of the sentence [29] the initial sentiment score of adjectives is increased further by 0.5.

C. Scoring microblog-specific language

Frequently microblog users use emoticons or repeated punctuation signs (*!!!!, ????, ...*) as a way of indicating their sentiments. Another frequently occurring language pattern is the use of elongated words by repeating certain letters (e.g *goooood* instead of *good*), and finally, writing in all capitals (e.g. *I HATE this place*). Our sentiment classification algorithm attempts to exploit the presence of such colloquiality in the text in order to accurately predict the sentiment polarity of a comment.

1) Scoring emoticons: During the text preprocessing phase of the algorithm, the text is matched against a manually created and annotated list of emoticons that express positive or negative sentiment. This list was derived by examining the official Unicode Emoji List [30] icon descriptions. Each occurrence of a positive or negative emoticon is being treated as a positive or a negative sentimental term respectively, with an initial sentiment score of 2.

2) Scoring repeated punctuation: The next step is to determine the presence of repeated punctuation signs in the text by using regular expressions. Repetitions of exclamation signs are considered a positive sentimental term whereas the sentiment polarity of repeated question marks or full stops is determined by the polarity of the immediately preceding textual sentimental term that has been found in the text. The initial sentiment score of repeated punctuation signs is 2.

3) Scoring enlogated and all caps words: In many cases users write words in all caps or repeat some letters in a word with the aim of emphasizing the significance of those words. Our algorithm has the ability to detect elongated words and words that are written in all caps. If a sentimental term is elongated or written in all caps its sentiment score is increased by 0.5.

D. Modifiers and negation handling

Our algorithm tries to leverage the presence of modifiers and negators in the text. Modifiers are words that have the property of modifying the strength of the sentiment expressed by a sentimental term. Intensifiers tend to intensify the strength of the expressed sentiment and diminishers, in contrast, tend to diminish the strength of the sentiment. For example, the word *very* operates as an intensifier in the phrase *the wine was very good*. On the other hand, an example of a diminisher is the word *little* in the phrase *the bar was a little crowded*. Negators are words that express negation and can alter the sentiment polarity of a sentimental term (e.g. the food was *not* expensive).

In order to capture the effect that modifiers have in the sentiment expressed in a comment we manually created lists of Greek and English modifiers. If a diminisher is found near a sentimental term (up to 3 words before or after the term) the sentiment score of the term is decreased by 0.5, whereas if an intensifier is found near a sentimental term its sentiment score is increased by 0.5.

For the negation handling module of the algorithm, we compiled a list of frequently used Greek and English negators. The presence of negators in the text is determined during text preprocessing and negators are replaced by a negator tag. If a negator is found in the neighborhood of a positive or negative sentimental term (up to 3 words before the term) the polarity of the term is reversed.

E. Greeklish handling and Spellcheck

A common occurrence in comments written by Greek users in microblogs is the use of Greeklish, that is, Greek text written in Latin alphabet. To handle this special case of text, we automatically created a Greeklish Sentiment Lexicon by replacing the words included in the Greek Sentiment Lexicon with their corresponding Greeklish words using a mapping of the Greek alphabet characters to the Latin characters that are frequently used to replace them in Greeklish text. The algorithm uses the Greeklish Sentiment Lexicon to detect Greeklish sentimental terms in the text.

One problem with the above approach for constructing the Greeklish lexicon, is that in Greeklish, Greek characters can be mapped in multiple ways to Latin ones [31], [32]. For example, the word ώρα (hour) is commonly transliterated as *wra* (a visually similar representation of the letter ω and also as *ora* (a phonetically accurate representation of the word).

Due to the informal nature of microblog comments, users often make spelling mistakes or use different Greeklish mappings than in our lexicon. Our algorithm implements a spellchecking module for Greek words and words written in Greeklish. If a word cannot be found in one of the sentiment lexicons, we compute the Levenhstein string distance between the word and the terms included in the lexicon that have the same character length or differ in character length by 1. The word is replaced by a term found in the lexicon if the Levenshtein distance between them is equal to 1.

F. Overview of Senti differences with other approaches

The main differences between Senti and other state-of-the-art works are the use of lexical resources for both the Greek and English language with the aim of creating a unified framework for the sentiment classification of bilingual microblog comments; A new scoring scheme for measuring sentiment strength expressed by textual and microblog-specific sentimental terms, and; A novel function for computing the overall sentiment scores of a comment in order to determine its sentiment polarity.

IV. Experimental Evaluation

A. Dataset and Experimental Setup

To evaluate the performance of our algorithm in the sentiment classification of microblog comments we conducted an experiment in which we asked human evaluators to rate a number of microblog comments regarding their sentiment polarity. The purpose of the experiment was to find out to what extent users agree with the sentiment classification of our algorithm. The experiment was conducted online without supervision.

The dataset we used for this experiment consisted of 3180 real comments about points of interest in the city of Patras, Greece that were obtained from Foursquare using the Foursquare API and were classified using our sentiment classification algorithm. Foursquare is a social media platform that provides users search results and information about points of interest and encourages them to write short messages in order to share their opinion about a point of interest with other users. Comments in Foursquare are limited to 200 characters in length.

Our algorithm incorporates the use of emoticons, repeated punctuation signs and negation handling to aid the task of sentiment classification. In order to assess the effectiveness of these modules in the sentiment classification task we identified comments that contain emoticons, repeated punctuation signs or negation with the aim of providing users a sufficient number of these type of comments for each sentiment polarity.

From the pool of 3180 comments, we subsequently randomly selected a total of 180 comments based on their sentiment polarity, language and the presence of emoticons, repeated punctuation signs or negation. From this process we obtained 30 comments from each of the six possible combinations of language and sentiment polarity, using stratification to include an appropriate proportion of comments that contained emoticons, repeated punctuation signs or negation, in each of the six categories. In Table I and Table II we present the characteristics of the initial and the final pool of comments respectively.

We opted to limit the number of comments in our pool to 180 because, as described in the next section, each participant in our experiment would be asked to provide feedback on a small subset (24 comments), to avoid participant fatigue. A smaller pool of choices provided greater likelihood that multiple ratings would be obtained for each of the comments, thus making a more objective comparison.

1) Task interface: We developed a simple web application in which participants first answered a demographics and background questionnaire. The web app was a responsive website, therefore participants could take part in the experiment from a mobile device or desktop computer. The URL of the website was advertised through our local university and participants were invited to take part in the task. Upon completion of the demographics and background sections, the application randomly selected 4 comments from each of the six different categories, making sure that a representative sample of comments that contain emoticons, repeated punctuation signs or negation were included for each category.

Each user was thus presented with a total of 24 comments from the pool of 180, and these comments were presented in random order, one by one. Users were instructed to read the comment and rate the sentiment they felt it expresses in a 5-point Likert scale, ranging from strongly positive to strongly negative. Further from the participant responses, we also collected the time it took them to rate each comment.

Table I
CHARACTERISTICS OF THE INITIAL POOL OF COMMENTS

Language	Polarity	Total	Emoticons	Punctuation	Negation
Greek	Positive	1393	2.6%	23.81%	1.62%
	Negative	334	0.25%	3.09%	63.19%
	Neutral	310	0	4.12%	0.15%
English	Positive	640	2.8%	17.15%	0.7%
	Negative	88	0.17%	1.49%	0.52%
	Neutral	415	0	3.24%	0.17%

Table II
CHARACTERISTICS OF THE FINAL POOL OF COMMENTS

Language	Polarity	Total	Emoticons	Punctuation	Negation
Greek	Positive	30	3.33%	24.44%	2.22%
	Negative	30	5.56%	3.33%	3.33%
	Neutral	30	0	4.44%	3.33%
English	Positive	30	3.33%	17.88%	8.89%
	Negative	30	2.22%	2.22%	6.67%
	Neutral	30	0	3.33%	2.22%

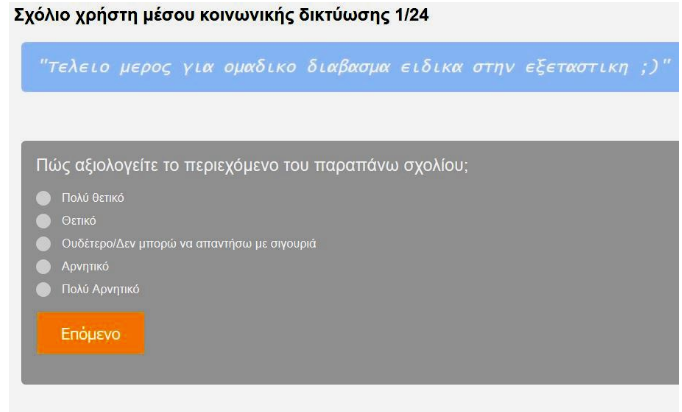

Figure 1. Experiment comment evaluation screen. Progress is shown on the page title (1/24), the review text (in blue background) and the user is asked to evaluate the sentiment polarity (gray background) in a scale of 1(very negative) to 5 (very positive).

Fig.1 shows the comment evaluation screen of the experiment.

2) Participant demographics and experience with social media comments: A total number of 148 users (120 female) participated in the experiment. Users were asked about their knowledge level of English language. 31.1% of the participants declared that they have a “high level knowledge” of English, 41.5% answered that they have “adequate knowledge”, 23.4% have “moderate knowledge” and 4.1% have “no knowledge” of English. Then we asked users about what social media and microblog apps they use in their smartphones in order to assess participants’ familiarity with microblogs. 70.3% of the participants use Facebook, 58.8% use Google Maps, 23.6% are Trip Advisor users, 8.8% are Twitter users, 4.1% are Foursquare users while 10.1% of the participants answered that they use none of the aforementioned apps. Next, users were asked if they read other users’ comments in microblogs. 79.7% of the participants answered that they read comments, result that shows the vast majority of the participants have experience in reading microblog comments. Lastly, users who declared that they read microblog comments were asked about the number of comments they usually read in order to form an opinion about a subject they are interested in. 52.5% of those users declared that they usually read 6 to 10 comments, while 24.6% read 1 to 5 comments and 22.9% read 11 to 20 comments. Overall we note that based on their exposure to user-generated content, the participant population should be able to properly evaluate content for sentiment.

B. Results

In total we collected 3,576 annotations from our participants. Because participants responded to the task unsupervised (i.e. remotely), we assumed that it is likely that some of the annotations were inappropriate (e.g. due to participant fatigue, indifference, interruption or malicious intent). Therefore we proceeded to remove some of this data, based on the time participants took to provide each response. To remove temporal outliers we use the adjusted boxplot method proposed in [33], which uses the medcouple (MC) value as a robust statistic to detect outliers. This resulted in setting an upper time threshold of 33.686s, thus we removed all responses with a longer response time. The lower threshold computed by this method is a negative value, which in our case, makes little sense. To set a lower threshold, we used the screen-reading speed reported in [34] (4.06 words per second), therefore, for each comment we calculated the minimum reasonable reading time of each comment based on its length, and excluded all responses with a response time lower than this threshold (i.e. users providing a response without actually reading the comment). For each comment, we assign a participant-based sentiment polarity according to the polarity class which received the most responses. This resulted in a final total of 3,390 responses.

Next, we present the comparison of our algorithm's classification performance based on the participant-sourced labelled set. The evaluation metrics we used to assess the performance of our algorithm (Senti) compared to the human evaluators are the Accuracy, Precision, Recall and F-score of the classification. As baseline case we used the algorithm for the sentiment classification of Foursquare comments proposed by [26]. The baseline algorithm uses a lexicon-based approach based on the Greek Sentiment Lexicon for the sentiment classification of Greek comments whereas English comments are classified by querying the Twinword Sentiment Analysis API [35]. We also evaluated the performance of our algorithm for the classification of English comments compared to SentiStrength [16].

Furthermore, we run our algorithm again on the same dataset this time without using the modules responsible for handling the occurrence of emoticons, repeated punctuation signs and negation in order to evaluate the impact that these modules have in the sentiment classification task. For the sake of brevity, we will refer to these modules as the micro modules of our algorithm.

In Table III we present the average evaluation metrics values for the sentiment classification, as well as the values for each of the three sentiment polarity classes. The average Accuracy of our algorithm is 77.03% and the average F-Score is 66.05%. Comparing the F-Score values of the Positive, Negative and Neutral classes we can deduce that our algorithm performs better in the classification of positive microblog comments.

Table III
EVALUATION METRICS OF SENTIMENT CLASSIFICATION

	Accuracy	Precision	Recall	F-score
Average	77,03%	65,55%	66,56%	66,05%
Positive Class	78,33%	83,33%	63,29%	71,94%
Negative Class	78,88%	55,00%	75,00%	63,46%
Neutral Class	73,88%	58,33%	61,40%	59,82%

Fig. 2 shows the average Accuracy and F-score of our algorithm in contrast with the average values for the algorithm without the micro modules (Senti*) and the average values of

Figure 2. Classification performance of the Senti algorithm compared to Senti* and the baseline case.

the baseline algorithm. We can see that our algorithm outperforms the baseline algorithm by a margin of approximately 3% in Accuracy and 6.6% in F-score. The performance of our algorithm drops considerably when the micro modules are not used but our algorithm still achieves a higher F-score than the baseline case.

The final step of the evaluation was to examine the classification performance of our algorithm for the Greek and the English language. Fig.3 and Fig.4 show the average Accuracy and F-score of our algorithm compared to Senti* and the baseline case for the Greek and English languages respectively. Our algorithm achieved an average F-score of 63,11% for the classification of Greek comments and an average F-score of 72,27% for English comments. Our algorithm performs better than the baseline case for both languages. Compared against SentiStrength [16], we obtain the same Accuracy and a better F-score in comparison for English comments. As before, we can see the performance drop when the micro modules are not used.

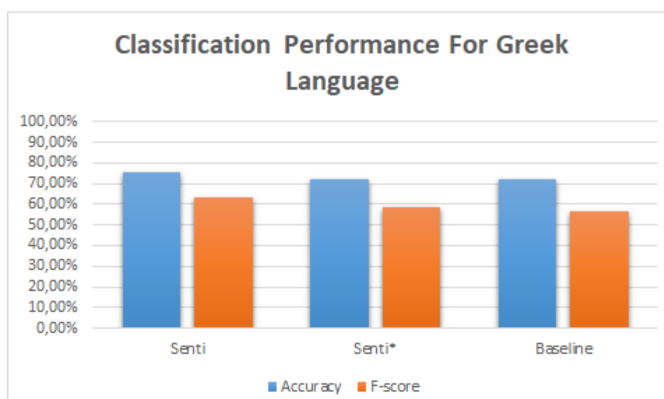

Figure 3. Classification performance of the Senti algorithm for Greek comments compared to Senti* and the baseline case.

C. Comparison against ML techniques

To also compare the performance of our algorithm against supervised machine learning techniques, we used the Rapid-

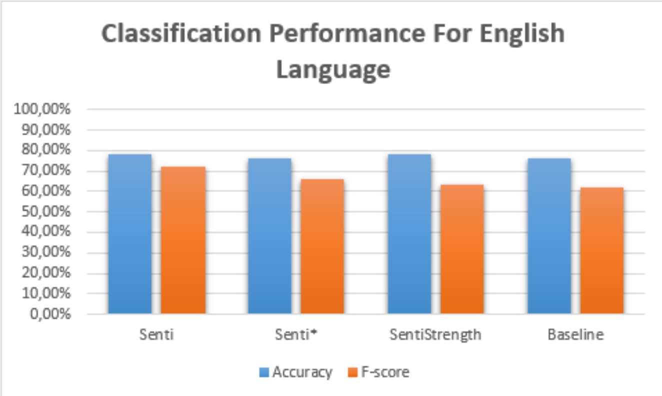

Figure 4. Classification performance of the Senti algorithm for English comments compared to Senti*, SentiStrength and the baseline case.

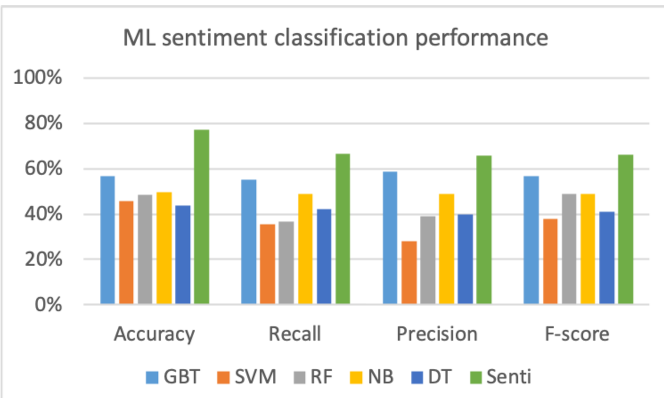

Figure 5. Performance metrics of ML algorithms vs. Senti

Miner environment to perform the relevant modelling and evaluation tasks. For the models' feature set, we use the *TF-IDF* vectors generated on the comment texts. To afford a fair comparison, we preprocess the text, applying the same special term token replacement as in our algorithm, and all tokens are converted to lower case. For this process we use 10-fold cross validation and the algorithm parameters used are the same as in [24]. As seen in Fig.5,. Senti outperforms all ML algorithms, with the 2nd best performance offered by GBT (F-score = 57%). Analyzing performance by language (Fig.6), Senti again outperforms all ML algorithms, though its performance is not too far off Naive Bayes and GBT for Greek (S:63.11%, NB: 56.74%, GBT: 55.22%).

Table IV
ML ALGORITHM PARAMETERS

Algorithm	**Parameters**
Decision Tree	Criterion: Gini Index; Maximal Depth: 60; Confidence: 0.3; Minimal Leaf Size: 12
SVM	Kernel Type: polynomial; Degree: 3; C: 1.35; Epsilon: $2x10^{-4}$
Naive Bayes	No parameters to optimize
Gradient Boosted Tree	Number of Trees: 120; Max-depth: 4; Learning rate: 0.05
Random Forest	Number of Trees: 100; Criterion: information-gain; Maximal Depth: 20

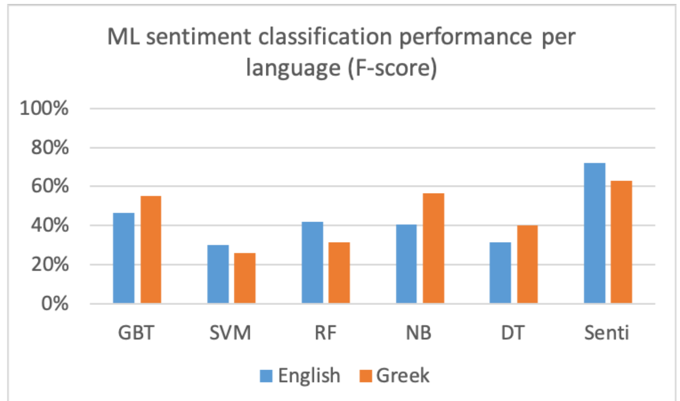

Figure 6. F-score metrics of ML algorithms vs. Senti per comment language

V. Conclusions and Future Work

In this paper, we presented a lexicon-based algorithm for the sentiment classification of user comments originating from microblogs and social media. Our method uses sentiment lexicons for the Greek and English languages in order to identify the sentiment expressed in comments written in both languages and takes advantage of the innate characteristics of the language used in microblogs. We proposed a weighting scheme for measuring the sentiment strength of sentimental terms based on the terms' degree of subjectivity and a simple function for calculating the sentiment scores of the text.

We tested our approach in a dataset of 180 user comments about points of interest obtained from Foursquare. Each comment was annotated by an average of approximately 19 human evaluators, fact that reinforces the reliability and significance of our results.

Our algorithm achieved an average Accuracy of 77.03% and an average F-score of 66.05% for the sentiment classification task. The performance of our algorithm was better than previous approaches [16], [26] for the classification of both Greek and English comments. The modules responsible for handling the occurrence of emoticons, repeated punctuation marks and negation improve considerably the performance of our algorithm. We also note that on our dataset, the algorithm significantly outperforms the ML methods. Ostensibly, these suffer from the fact that the training data available to them is limited, compared to the several thousand documents fed to these algorithms in other studies (e.g. [24]). As demonstrated in [2], it is likely that a larger labelled dataset might afford better results to this class of algorithms.

In the future, we plan to assess the different elements that constitute our method in order to consider adjustments and additions that could further improve the performance of our algorithm. A possible addition could be to expand the lists of modifiers and negators that our algorithm uses to detect the presence of these linguistic features in the text. Furthermore, we could ask human evaluators to evaluate the list of emoticons used in our method with the aim of obtaining a more reliable judgement of the sentiment expressed by emoticons.

We would like to evaluate our algorithm on larger reliably labelled datasets, which, however, require significant effort to obtain. Finally, a more robust approach for the handling of Greeklish text that takes into account the different ways of transliterating Greek words to Greeklish, could be considered.

REFERENCES

[1] Bo Pang and Lillian Lee. Opinion Mining and Sentiment Analysis. *Foundations and Trends® in Information Retrieval*, 2(1-2):1–135, 2008.

[2] Alexandros Baltas, Andreas Kanavos, and Athanasios K. Tsakalidis. An Apache Spark Implementation for Sentiment Analysis on Twitter Data. In Timos Sellis and Konstantinos Oikonomou, editors, *Algorithmic Aspects of Cloud Computing*, Lecture Notes in Computer Science, pages 15–25. Springer International Publishing, 2017.

[3] Adam Bermingham and Alan Smeaton. Classifying Sentiment in Microblogs: Is Brevity An Advantage? *CIKM '10 Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1833–1836, 2010.

[4] Dmitry Davidov, Oren Tsur, and Ari Rappoport. Enhanced Sentiment Learning Using Twitter Hashtags and Smileys. *COLING 2010, 23rd International Conference on Computational Linguistics*, 2010.

[5] Satarupa Guha, Aditya Joshi, and Vasudeva Varma. Sentibase: Sentiment Analysis in Twitter on a Budget. *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 590–594, 2015.

[6] Alexandra Balahur. Sentiment Analysis in Social Media Texts. *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 120–128, 2013.

[7] Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. Twitter Sentiment Analysis: The Good the Bad and the OMG! *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, 11:538–541, 2011.

[8] Saif M. Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. NRC Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. *Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013)*, 2013.

[9] Svetlana Kiritchenko, Xiaodan Zhu, and Saif M. Mohammad. Sentiment Analysis of Short Informal Texts. *Journal of Artificial Intelligence Research*, 50:723–762, 2014.

[10] Fotis Aisopos, George Papadakis, Konstantinos Tserpes, and Theodora Varvarigou. Content vs. Context for Sentiment Analysis: a Comparative Analysis over Microblogs. *HT '12 Proceedings of the 23rd ACM conference on Hypertext and social media*, pages 187–196, 2012.

[11] Alina Andreevskaia and Sabine Bergler. Mining WordNet for a fuzzy sentiment: Sentiment tag extraction from WordNet glosses. *In Proceedings of the European Chapter of the Association for Computational Linguistics (EACL)*, 2006.

[12] Peter Turney. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. *In Proceedings of the Association for Computational Linguistics (ACL)*, pages 417–424, 2002.

[13] Cataldo Musto, Giovanni Seemerano, and Marco Polignano. A Comparison of Lexicon-based approaches for Sentiment Analysis of microblog posts. *Proceedings of the 8th International Workshop on Information Filtering and Retrieval*, pages 59–68, 2014.

[14] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. SentiWordNet 3.0: An Enhanced Lexical Resourcefor Sentiment Analysis and Opinion Mining. *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, pages 2200–2204, 2010.

[15] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. *HLT '05 Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 347–354, 2005.

[16] Mike Thelwall, Kevan Buckley, Georgios Paltoglou, Di Cai, and Arvid Kappas. Sentiment Strength Detection in Short Informal Text. *Journal of the American Society for Information Science and Technology*, 61(12):2544–2558, 2010.

[17] Georgios Paltogloy and Mike Thelwall. Twitter,MySpace,Digg: Unsupervised Sentiment Anaysis in Social Media. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(4), 2012.

[18] Katerina Tsagkalidou, Vassiliki Koutsonikola, Athena Vakali1, and Konstantinos Kafetsios. Emotional Aware Clustering on Micro-blogging Sources. *ACII'11 Proceedings of the 4th international conference on Affective computing and intelligent interaction*, Part I:387–396, 2011.

[19] Xiangfeng Dai, Irena Spasić, and Frédéric Andrès. A Framework for Automated Rating of Online Reviews against the Underlying Topics. *ACM SE '17 Proceedings of the SouthEast Conference*, pages 164–167, 2017.

[20] Hamidreza Keshavarz and Mohammad Saniee Abadeh. ALGA: Adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs. *Knowledge-Based Systems*, 122:1–16, 2017.

[21] George Markopoulos, George Mikros, Anastasia Iliadi, and Michalis Liontos. Sentiment Analysis of Hotel Reviews in Greek: A Comparison of Unigram Features. *Cultural Tourism in a Digital Era, Springer Proceedings in Business and Economics*, 2015.

[22] Maria Giatsoglou, Manolis G. Vozalis, Konstantinos Diamantaras, Athena Vakali, George Sarigiannidis, and Konstantinos Ch. Chatzisavvas. Sentiment analysis leveraging emotions and word embeddings. *Expert Systems with Applications*, 69:214–224, 2017.

[23] Adam Tsakalidis, Symeon Papadopoulos, Rania Voskaki, Kyriaki Ioannidou, Christina Boididou, Alexandra I. Cristea, Maria Liakata, and Yiannis Kompatsiaris. Building and evaluating resources for sentiment analysis in the Greek language. *Language Resources and Evaluation*, 52(24):1021–1044, 2018.

[24] Vasileios Athanasiou and Manolis Maragoudakis. A Novel, Gradient Boosting Framework for Sentiment Analysis in Languages where NLP Resources Are Not Plentiful: A Case Study for Modern Greek. *Algorithms*, 10(1), 2017. bibtex*[article-number=34].

[25] Katia Lida Kermanidis and Manolis Maragoudakis. Political sentiment analysis of tweets before and after the Greek elections of May 2012. *International Journal of Social Network Mining,*, 1(3/4):298–317, 2013.

[26] Panayiotis Kolokythas, Andreas Komninos, Lydia Marini, and John Garofalakis. A lightweight algorithm for the emotional classification of crowdsourced venue reviews. *PCI2017 Proceedings of the 21st Pan-Hellenic Conference on Informatics*, 2017.

[27] Georgios Kalamatianos, Dimitrios Mallis, Symeon Symeonidis, and Avi Arampatzis. Sentiment Analysis of Greek Tweets and Hashtags using a Sentiment Lexicon. *PCI2015 Proceedings of the 19th Pan-Hellenic Conference on Informatics*, 2015.

[28] Adam Tsakalidis, Symeon Papadopoulos, Ourania Voskaki, Kyriaki Ioannidou, and Christina Boididou. Greek sentiment lexicon. *[Online]. Available: http://www.socialsensor.eu/results/datasets/147-greek-sentiment-lexicon. [Accessed: 15- Apr- 2019]*, 2014.

[29] Vasileios Hatzivassiloglou and Janyce Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. *Proceedings of the International Conference on Computational Linguistics (COLING)*, 2000.

[30] Unicode. Full Emoji List, v12.0. *[Online]. Available: https://unicode.org/emoji/charts/full-emoji-list.html. [Accessed: 15- Apr- 2019].*

[31] Theodoros Marinis, Angeliki Papangeli, and Theodora Tseliga. "Potizo" or "Potizw"? The Influence of Morphology in the Processing of Roman-alphabeted Greek. *Selected Papers in Theoretical and Applied Linguistics: 17th International Symposium on Theoretical and Applied Linguistics*, 2:443–452, 2007.

[32] Paraskevi Tzekou, Sofia Stamou, Nikos Zotos, and Lefteris Kozanidis. Querying the Greek Web in Greeklish. *Proceedings of the ACM SIGIR Workshop on Improving Web Retrieval for non-English Queries*, 2007.

[33] M. Hubert and E. Vandervieren. An adjusted boxplot for skewed distributions. *Computational Statistics & Data Analysis*, 52(12):5186–5201, August 2008.

[34] Mary Dyson and Mark Haselgrove. The effects of reading speed and reading patterns on the understanding of text read from screen. *Journal of Research in Reading*, 23(2).

[35] Twinword. Twinword Sentiment Analysis API. *[Online]. Available: https://www.twinword.com/api/sentiment-analysis.php. [Accessed: 15- Apr- 2019].*