Using Arbiter and Combiner Tree to Classify Contexts of Data

Tawunrat Chalothorn and Jeremy Ellman

***Abstract*—This paper reports on the use of ensemble learning to classify as either positive or negative the sentiment of Tweets. Tweets were chosen as Twitter is a popular tool and a public, human annotated dataset was made available as part of the SemEval 2013 competition. We report on a classification approach that contrasts single machine learning algorithms with a combination of algorithms in an ensemble learning approach. The single machine learning algorithms used were support vector machine (SVM) and Naïve Bayes (NB), while the methods of ensemble learning include the arbiter tree and the combiner tree. Our system achieved an F-score using Tweets and SMS with the arbiter tree at 83.57% and 93.55%, respectively, which was better than base classifiers; meanwhile, the results from the combiner tree achieved lower scores than base classifiers.**

***Index Terms*—Tweets, contexts, positive, negative, natural language processing, ensemble learning, sentiment analysis.**

I. Introduction

The research area of natural language processing (NLP) comprises various tasks; one of which is sentiment analysis. The main goal of sentiment analysis is to identify the polarity of natural language text. Sentiment analysis can be referred to as opinion mining; studying opinions, appraisals and emotions towards entities, events and their attributes. Sentiment analysis is a popular research area in NLP that aims to identify opinions or attitudes in terms of polarity. Currently, Twitter is a popular microblogging tool where users are increasing by the minute. Twitter allows users to post messages of up to 140 characters each time. These are called 'Tweets', which are often used to convey opinions about different topics. Consequently, various researchers are interested in classifying Tweets using sentiment analysis.

This paper introduces the original process of using the arbiter tree [1] and combiner tree [2], to classify the contexts of Tweet datasets and uses SMS datasets to evaluate the system. Arbiter tree [1] and combiner tree [2] have been chosen because they have not yet been used in sentiment analysis to classify Tweets or SMS datasets. The basic idea is to divide the training data into subsets, apply the learning algorithm to each and merge the resulting inducers. The main task is to find a solution to combining the appropriate learning model in order to achieve better results. Our main contribution is to propose and experiment with a combination of two machine learning algorithms, based on the use of the arbiter tree [1]. The remainder of this paper is constructed as follows: the details of related works are mentioned in Section II. The corpus used is discussed in Section III; the methodology with data pre-processing and details of classifier are presented in Section IV; Section V discusses the details of the experiment and results. Finally, a conclusion and recommendations for future work are provided in Section VI.

Manuscript received December 12, 2014; revised June 12, 2015.

The authors are with University of Northumbria at Newcastle, Department of Computer Science and Digital Technologies, Pandon Building, Camden Street Newcastle Upon Tyne, NE2 1XE, United Kingdom (e-mail: tawunrat.chalothorn@northumbria.ac.uk).

II. Related Works

The microblogging tool Twitter is well-known and increasingly popular. The site allows users to post messages, or 'Tweets', of up to 140 characters each time. These are available for immediate download over the Internet. Tweets are extremely interesting to the marketing sector, since their rapid public interaction can indicate either customer success or presage public relations disasters far more quickly than web pages or traditional media. Consequently, the content of Tweets and identifying their sentiment polarity as positive or negative is currently an active research topic. There are various researches that use Tweets with machine leaning algorithms; for example, [3] classify Twitter using Naïve Bayes (NB) [4], [5], Maximum Entropy Modelling [6], [7] and Support Vector Machine (SVM) [8], [9]. In the experiment, emoticons have been used as noisy labels in training data to identify the label as positive or negative. Emoticons can be referred to printable characters of emotion, such as :-) for smile and :-(for sad. SVM [8], [9] with unigram obtained high accuracy at 82.90%. [3] note that using negation and part-of-speech tagging did not help improve accuracy.

Ref. [10] divided Tweets into three groups using emoticons for classification. If Tweets contain positive emoticons, they will be classified as positive, and vice versa. Other Tweets that do not have positive/negative emoticons will be classified as neutral. However, those that contain both positive and negative emotions are ignored in their study. Their task focused on analyzing the contents of social media using n-gram graphs. The results revealed that n-grams yielded high accuracy when tested with C4.5 [11], but low accuracy with NB Multinomial (NBM) [12].

III. Corpus

The datasets used in our experiment are taken from SemEval 2013 [13]. The data were gathered from Twitter; a well-known and increasingly popular microblogging site. Twitter allows its users to post messages, or 'Tweets', of up to 140 characters each time, which are available for immediate download over the Internet. Tweets are extremely interesting in marketing terms, since their rapid public

interaction can either indicate customer success or presage public relations disasters far more quickly than web pages or traditional media. Consequently, the content of tweets and identifying their sentiment polarity as positive or negative is a current active research topic.

The datasets comprise training data, testing data and gold standard. Gold standard refers to the testing data labelled with the correct polarity. However, these datasets were annotated using five Mechanical Turk workers; also known as Turkers [13]. For each sentence, they will use the start and end point of their opinion for the phrase or word, and state whether it is negative, neutral or positive. Then, the words that appear three times from five votes will be assigned the label. In addition to Tweets, SMS messages are used to evaluate the system. SMS messages are also obtained from the organizer of SemEval 2013 [13]. Only the datasets labelled as positive and negative will be used in this research.

Furthermore, three sentiment lexicons were used in this experiment. They are Bing Liu Lexicon (HL) (6780 words), collected over many years by [14]. They began to accumulate lexicons in 2004, during the course of their work on online customer product reviews [14]. MPQA Subjective Lexicon (MPQA) (8221 words) was created by [15] using a set of approximately 400 documents. AFINN Lexicon (AFINN) (2477 words) was created from Twitter between 2009-2011 by [16] for use in the United Nation Climate Conference (COP15).

IV. Methodologies

A. Data Pre-processing

For the process of data pre-processing, emoticons were labelled by matching those collected manually from the dataset against a well-known group of emoticons. Subsequently, negative contractions were expanded and converted to full form (e.g. don't -> do not). Moreover, the features of Tweets were removed or replaced by words, such as Twitter usernames, URLs and hashtags.

A Twitter username is a unique name displayed in the user's profile and may be used for both authentication and identification. This is demonstrated by prefacing the username with an @symbol. When a Tweet is directed towards a specific individual or entity, this can be displayed by including @username in the Tweet. For example, a Tweet directed at 'som' would include the text @som. Before URLs are posted to Twitter, they are shortened automatically to use the t.co domain whose modified URLs contain a maximum of 22 characters. However, both features have been removed from the datasets. Hashtags are used to represent keywords and topics in Twitter by using # followed by words or phrases; for example, #newcastleuk. This feature has been replaced with the following word after the # symbol. For example, #newcastleuk was replaced with newcastleuk.

Frequently, repeated letters are used to provide emphasis in Tweets. These were reduced and replaced using a simple regular expression by two of the same characters. For example, happppppppy will be replaced with happy, and coolllllll will be replaced with cool. Next, special characters were removed, such as [,{,?,and !. Slang and contracted words were converted to their full form; for example, 'fyi' became 'for your information'. Finally, Natural Language Toolkit (NLTK) [17] stopwords were removed from the datasets, such as 'a', 'the', etc. The metric and comparison of these features can be found in [18]. The flowchart of data processing are shown in Fig. 1.

B. Arbiter Trees

Arbiter tree [1] is a method that uses training data classified by using base classifiers with selection rules. Selection rules are used to compare the prediction of base classifiers for choosing the training dataset for the arbiter. Then, the final prediction is decided based on the base classifiers and arbiter by using arbitration rules with the aim of learning from incorrect classification [1].

C. Combiner Tree

The Combiner tree [2] method has similar qualities to the arbiter tree but it will be trained directly by the training output from the base classifiers that passed the composition rules. Next, the final prediction will be classified by the combiner. There are two versions of composition rules: the first uses the combination of results from the base classifier; while the second uses the same as the first with the addition of training data attributes. The aim of the combiner tree is to learn from correct classification [2].

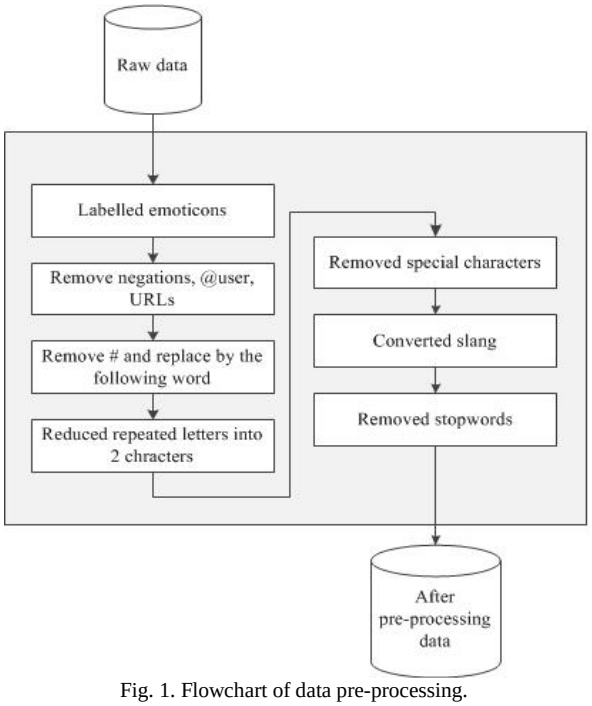

Fig. 1. Flowchart of data pre-processing.

D. Support Vector Machine

SVM [8], [9] is a binary linear classification model with the learning algorithm for classification and regression analysis of data, and recognizing the pattern. The purpose of SVM is to separate datasets into classes and discover the decision boundary (hyper-plane). To find the hyper-plane, the maximum distance between classes (margin) will be used with the closest data points on the margin (support vector). The equation of SVM can present as:

$$\vec{w} = \sum_j \alpha_j c_j \vec{d}_j, \ \alpha_j \geq 0 \quad (1)$$

where vector $\vec{w}$ represented as hyperplane. c_j is a polarity (negative and positive) of the data d_j which $c_j \in \{-1, 1\}$.

α_j are obtained by solving he dual optimisation problem. Those $\vec{d}_j$ such that α_j is greater than zero are called, support vectors, since they are the only document vectors contributing to $\vec{w}$. Classification of test instances consists simple of determining which side of $\vec{w}$ hyperplane they fall on. Our research used the default setting of SVMLight for the SVM classifier model. SVMLight is an implementation of SVM in C.

E. Naïve Bayes

The NB algorithm [5] is a classification algorithm based on Bayes' theorem that underlies the naïve assumption that attributes within the same case are independent given the class label [19]. This is also known as the state-of-art Bayes rules [20]. NB [5] constructs the model by adjusting the distribution of the number for each feature. For example, in text classification, NB regards the documents as a bag-of-words, from which it extracts features. NB [5] model follows the assumption that attributes within the same case are independent given the class label [21]. Tang, *et al.* [22] considered that Naïve Bayes assigns a context X_i(represented by a vector X_i^*) to the class C_j that maximizes $P(C_j|X_i^*)$ by applying Bayes's rule, as in (2).

$$P(C_j|X_i^*) = \frac{P(C_j)P(X_i^*|C_j)}{P(X_i^*)} \quad (2)$$

where $P(X_i^*)$ is a randomly selected context X. The representation of vector is X_j^*. $P(C)$ is the random select context that is assigned to class C.

To classify the term $P(X_i^*|C_j)$, features in X_i^* were assumed as f_j from $j = 1\ to\ m$ as in (3).

$$P(C_j|X_i^*) = \frac{P(C_j)\prod_{j=1}^{m} P(f_j|C_j)}{P(X_i^*)} \quad (3)$$

In this research, the NB algorithm was used from the NLTK, which is a widely-used machine learning algorithm, open source, developed using Python and comprising the WordNet interface.

V. Experiment and Results

In our experiment, the idea from [1] has been adapted using the arbitter tree algorithm, as only two classifiers are used with one training data. In order to build the training data, all selection rules from [1] were adapted and used in this experiment. The processes for creating training data are detailed below:

1) Base training data were trained into base classifiers, which are SVM [8], [9] and NB [5]. The base training data were yielded from the combination of the sentiment lexicons noted in section III. They were combined by removing the words that duplicate, overlap and contradict in sentiment [23]-[26].
2) After obtaining the results from the base classifiers, they were united and passed into selection rules. There are three versions of selection rules:

a) Selection rule 1 is the different results from classifiers 1 and 2.
b) Selection rule 2 is the union of the results from selection rule 1 and the results from classifiers 1 and 2, which are the same prediction but incorrect.
c) Selection rule 3 is the union of selection rules 1 and 2 and the results of classifiers 1 and 2, which are the same prediction and correct.

3) As in the arbiter tree algorithm, [1] did not state clearly how to use the selection rules; therefore, the data from selection rules 1 , 2 and 3 have been trained with base classifiers that assume to be the arbiter for creating the final training data. The flowchart of these processes is presented in Fig. 2.

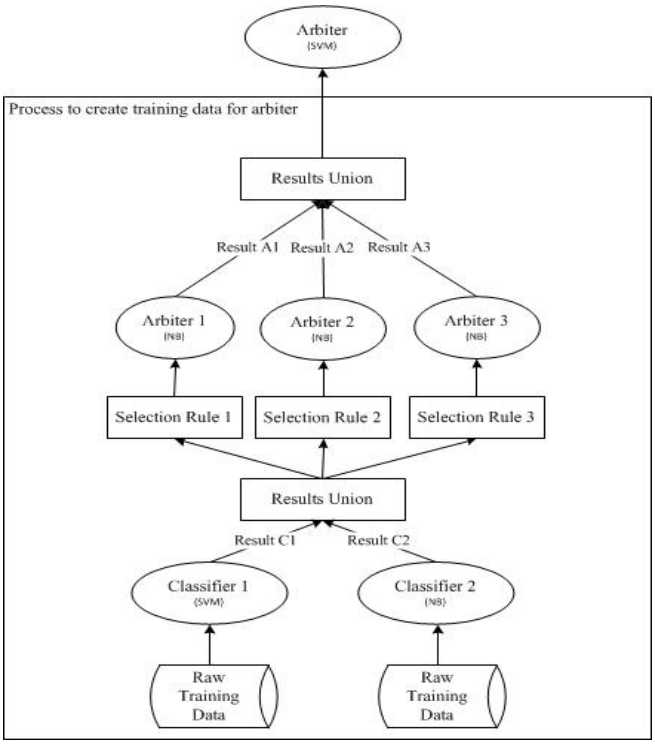

Fig. 2. Process for making training data for arbiter.

TABLE I: The Results of Tweets and SMS Dataset from Base Classifiers

	Tweet dataset Avg. F-score (%)	**SMS dataset** Avg. F-score (%)
SVM	83.55	85.49
NB	81.54	85.05

TABLE II: The Results of Tweets and SMS Dataset from Arbiter Tree

	Tweet dataset Avg. F-score (%)	**SMS dataset** Avg. F-score (%)
Arbiter rules version 1	82.31	84.87
Arbiter rules version 2	83.57	85.56

After obtaining the final training data for the arbiter, they were used in the final classification process for the final prediction results. During this process (see Fig. 3), the base classifiers were rained by using base training data, while the arbiter was trained by using arbiter training data to classify the test set. Next, their results went through the process of arbiter rules for the final prediction results. There are two versions of arbiter rules. The first uses the majority vote of prediction from the base classifier and the arbiter prediction. If the results of predictions 1 and 2 are equal, the results from prediction 2 will be used. Conversely, the arbiter results will be used. In the second version, if the results of predictions 1 and 2 are not equal, the different arbiter results will be used. If the results of prediction 1 are equal to those of the correct

arbiter, use the correct arbiter results. In contrast, the incorrect results from the arbiter tree were used. The evaluation metric was used F-score [27].

TABLE III: THE RESULTS OF TWEETS AND SMS DATASET FROM COMBINER TREE

	Tweet dataset Avg. F-score (%)	**SMS dataset** Avg. F-score (%)
Combiner rules version 1	30.25	34.59
Combiner rules version 2	32.36	34.65

The datasets of Tweets and SMS were tested in the arbiter tree [1]. Their results are presented in Table I. Following the comparison between the arbiter and base classifiers (Table II), the results of Tweets using arbiter rules version 1 did not achieved better accuracy than base classifiers at 82.31%; meanwhile, the results from arbiter rules version 2 achieved a better F-score than SVM [8, 9] and NB [5] at 83.57 %. Conversely, the results of the SMS dataset revealed that the results from arbiter rule version 2 achieved a better F-score than base classifiers at 84.57% and 85.56%, respectively.

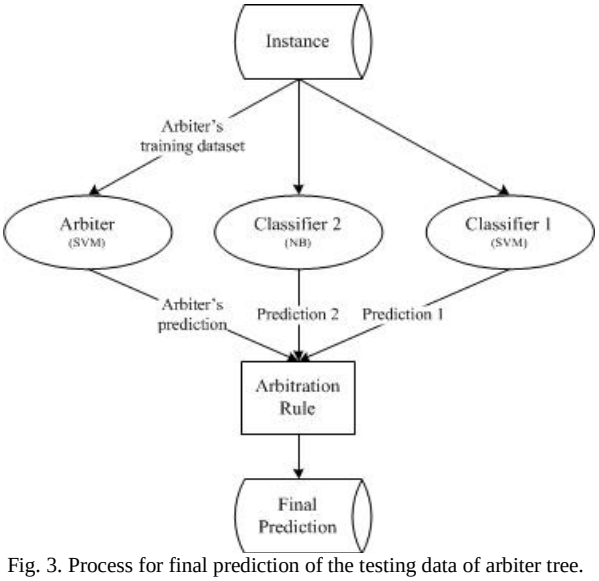

Fig. 3. Process for final prediction of the testing data of arbiter tree.

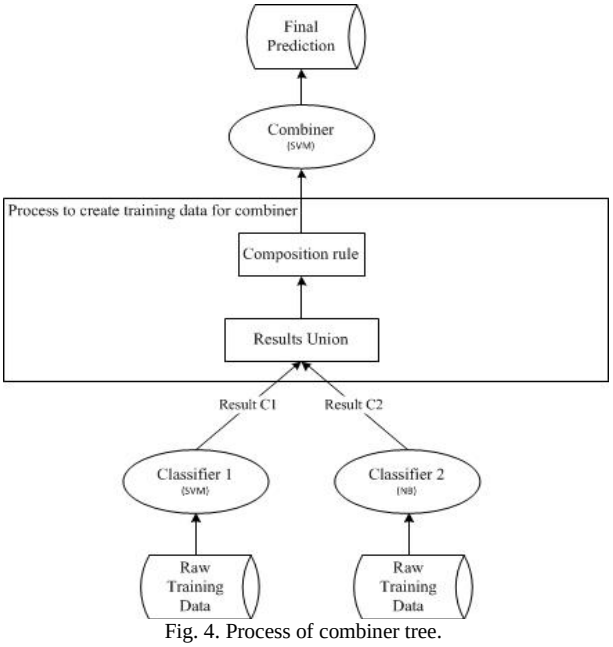

Fig. 4. Process of combiner tree.

In addition to the arbiter tree [1], the combiner tree [2] was also used in the experiment for comparison purposes. The training dataset for the combiner have to be built based on the base classifiers and composition rules, see Fig. 4. There are two versions of the composition rules: The first version uses the combination of results from the base classifiers, while the second uses a combination of the first version and the instance from training data. Next, they will be used as the training data for classify the testing data. The results of testing Tweets demonstrated a very low F-score of 30.25% and 32.36% respectively for the first and second versions. Conversely, the results from SMS revealed F-scores of 34.59% and 34.65% respectively for the first and second versions. The results from the combiner tree [2] (see Table III) achieved lower F-scores than base classifiers in both datasets.

VI. CONCLUSION AND FUTURE WORK

In this experiment, the original process of using the arbiter tree [1] and combiner tree [2] algorithms to classify Tweets and SMS datasets have been demonstrated and clearly explained. The use of ensemble learning might not always have achieved the most accuracy as the results from combiner tree [2]; however, the results of the classification of Tweets and SMS dataset using arbiter tree [1], demonstrated their ability to achieve F-scores of 83.57% and 92.55%, respectively, which is better than the scores achieved for both base classifiers.

For future work, the results from the arbiter tree [1] will be combined with the SVM [8], [9], NB [5] and SentiStrength [28] by using majority voting. The main purpose is to improve sentiment classification using a combination of machine learning algorithms and sentiment resources. SentiStrength [28] is the sentiment analysis methodology used to judge whether a sentence has a positive or negative sentiment, which is developed from comments posted on MySpace.

REFERENCES

[1] P. K. Chan and S. J. Stolfo, "Toward parallel and distributed learning by meta-learning," presented at the International Association for the Advancement of Artificial Intelligence (AAAI) Workshop in Knowledge Discovery in Databases, 1993.
[2] P. K. Chan and S. J. Stolfo, "On the accuracy of meta-learning for scalable data mining," *Journal of Intelligent Information Systems*, vol. 8, pp. 5-28, 1997.
[3] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," CS224N Natural Language Processing, Project Report, Stanford, pp. 1-12, 2009.
[4] D. D. Lewis, "Naive (Bayes) at Forty: The independence assumption in information retrieval," presented at the 10th European Conference on Machine Learning, 1998.
[5] J. Liangxiao, H. Zhang, and C. Zhihua, "A novel bayes model: Hidden Naive Bayes," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, pp. 1361-1371, 2009.
[6] E. T. Jaynes, "Information theory and statistical mechanics," *Physical Review*, vol. 106, p. 620, 1957.
[7] R. A. Baldwin, "Use of maximum entropy modeling in wildlife research," *Entropy*, vol. 11, pp. 854-866, 2009.
[8] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and Their Applications*, vol. 13, pp. 18-28, 1998.
[9] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University Press, 2000.
[10] F. Aisopos, G. Papadakis, and T. Varvarigou, "Sentiment analysis of social media content using N-Gram graphs," presented at the 3rd ACM

the Special Interest Group on Multimedia (SIGMM) International Workshop on Social Media, Scottsdale, Arizona, USA, 2011.
[11] A. R. Abdel-Dayem, "Detection of arterial lumen in sonographic images based on active contours and diffusion filters," presented at the 7th International Conference on Image Analysis and Recognition-Volume Part II, Portugal, 2010.
[12] A. McCallum and K. Nigam, "A comparison of event models for Naive Bayes text classification," in *Proc. AAAI-98 Workshop on Learning for Text Categorization*, 1998, pp. 41-48.
[13] T. Wilson, Z. Kozareva, P. Nakov, A. Ritter, S. Rosenthal, and V. Stoyanov, "SemEval-2013 task 2: Sentiment analysis in Twitter," presented at the 7th International Workshop on Semantic Evaluation (SemEval), 2013.
[14] M. Hu and B. Liu, "Mining and summarizing customer reviews," presented at the tenth ACM Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD) International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA, 2004.
[15] T. Wilson, P. Hoffmann, S. Somasundaran, J. Kessler, J. Wiebe, Y. Choi, *et al.*, "OpinionFinder: A system for subjectivity analysis," presented at the HLT/EMNLP on Interactive Demonstrations, Vancouver, British Columbia, Canada, 2005.
[16] F. A. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," presented at the 7th International Conference Mechatronic Systems and Materials (MSM 2011), Kaunas, Lithuania, 2011.
[17] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly, 2009.
[18] T. Chalothorn and J. Ellman, "TJP: Identifying the polarity of tweets from contexts," presented at the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 2014.
[19] M. Elangovan, K. I. Ramachandran, and V. Sugumaran, "Studies on Bayes classifier for condition monitoring of single point carbide tipped tool based on statistical and histogram features," *Expert Systems with Applications*, vol. 37, pp. 2059-2065, 2010.
[20] A. Cufoglu, M. Lohi, and K. Madani, "Classification accuracy performance of Naive Bayesian (NB), Bayesian Networks (BN), Lazy Learning of Bayesian Rules (LBR) and Instance-Based Learner (IB1) — Comparative study," presented at the International Conference on Computer Engineering & Systems (ICCES), 2008.
[21] L. R. Hope and K. B. Korb, "A bayesian metric for evaluating machine learning algorithms," presented at the 17th Australian Joint Conference on Advances in Artificial Intelligence, Cairns, Australia, 2004.
[22] H. Tang, S. Tan, and X. Cheng, "A survey on sentiment detection of reviews," *Expert Systems with Applications*, vol. 36, pp. 10760-10773, 2009.
[23] P. Melville, W. Gryc, and R. D. Lawrence, "Sentiment analysis of blogs by combining lexical knowledge with text classification," presented at the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 2009.
[24] B. Yuan, Y. Liu, H. Li, T. T. T. Phan, G. Kausar, C. N. Sing-Bik *et al.*, "Sentiment classification in Chinese microblogs: Lexicon-based and learning-based approaches," *International Proceedings of Economics Development and Research (IPEDR)*, vol. 68, 2013.
[25] E. Refaee and V. Rieser, "An Arabic twitter corpus for subjectivity and sentiment analysis," presented at the 9th International Conference on Language Resources and Evaluation (LREC'14), 2014.
[26] L. Wang and C. Cardie, "A piece of my mind: A sentiment analysis approach for online dispute detection," presented at the 52nd Annual Meeting of the Association for Computational Linguistics Baltimore, USA, 2014.
[27] D. M. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," *Machine Learning RTechnologies*, vol. 2, pp. 37-63, 2011.
[28] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, *SentiStrength*, University of Wolverhampton, 2010.

Tawunrat Chalothorn has a M.Sc. degree in web computing from the University of Northumbria at Newcastle and currently is a postgraduate researcher at the Department of Computer Science and Digital Technologies at the University of Northumbria at Newcastle, UK.

Jeremy Ellman has a B.Sc. degree in experimental psychology from the University of Sussex, an M.Sc. degree in computer science from Essex University, and a Ph.D. degree in computer science from the University of Sunderland. Jeremy is currently a senior lecturer at the Department of Computer Science and Digital Technologies at the University of Northumbria at Newcastle, UK.