Named Entity Sentiment Classifications using Peripheral Words and Dependencies in Online Discussions

Tomohiro Ando
Graduate School of Engineering
Tokyo University of Agriculture and Technology
Koganei, Tokyo Japan
Email: ando@katfuji.lab.tuat.ac.jp

Katsuhide Fujita
Institute of Engineering
Tokyo University of Agriculture and Technology
Koganei, Tokyo Japan
Email: katfuji@cc.tuat.ac.jp

***Abstract*—The sentiment classification method of posts and sentences have been proposed in several online discussion forums. However, large-town online meetings require new methods to determine the sentiment polarity of each keyword because several topics are discussed simultaneously. We propose a sentimental classification method for each named entity in various online discussion forums. We employ machine learning for the web discussion corpus and sentiment lexicon that we have developed. We define three features that focus on the peripheral words of the named entities and on the modification structures. Our experimental results exhibit that the features of the peripheral and modification structure improve the f1-score as compared with the baseline of the f1-score.**

I. Introduction

Online discussions exhibit several advantages, e.g., meetings can be conducted regardless of the users' locations, and the histories of conversations can be recorded. Opportunities for online discussions are expected to increase. For example, COLLAGREE, an online discussion forum, conducted a town meeting in Nagoya and developed a support system for the participants and the facilitators [1]. In online discussion forums, stance classification, which defines a participant's stance toward a specific topic in the text, is considered to be an important task. However, the number of topics tends to increase as the discussion proceeds. Several topics are observed to emerge simultaneously; however, the stance of each participant must be classified for each topic.

This study focuses on the sentiment analysis of the named entities of each post in online discussions and aims to automatically classify the entities into three types (positive/negative/neutral). Generally, sentiment analysis is used to determine the attitude of a writer or other subjects toward some topic, the overall contextual polarity, or the emotional reactions to a particular document or interaction. The existing approaches to sentiment analysis can be classified into three main categories: knowledge-based techniques, statistical methods, and hybrid approaches[2]. The techniques are based on the presence of unambiguous affect words, statistical information from machine learning, and hybrid approaches that combine both machine learning and elements from knowledge representation such as ontologies and semantic networks. However, the main targets of sentiment analysis are the sentences or documents (posts). Understanding the stances of users related to some issues or themes is important in case of an online discussion forum. However, the sentiment classification of a named entity in a post is difficult because the stances on the named entity may be completely different based on the situations and users.

In this study, we propose a sentiment classification method of each named entity in online discussion forums. We employed machine learning to develop our web discussion corpus and sentiment lexicon. We defined three features that focused on the peripheral words of the named entities and on the modification structures. Our experimental results exhibit that the features of the peripheral and modification structure improved the f1-score as compared with the baseline.

The remainder of this study is organized as follows. First, we discuss other studies related to machine learning for sentimental classification. Further, we describe the proposed method for the automatic expansion of the sentiment lexicon and the machine learning method for using the lexicon. Additionally, we discuss the experimental results. Finally, we conclude this study and make recommendations to conduct future research.

II. Related Works

Sentiment classification is used for various web services. Further, considering the characteristic of each web service is important. Pang et al. proposed naive Bayes, maximum entropy, and support vector machines (SVMs) to perform the positive-negative classification of movie reviews and compared the performances of the classifiers [3]. Further, machine learning was used to perform the sentiment classifications of SNS posts, reviews of items, microblogs, and tweets. Soroush et al. developed a method for learning a tweet by acquiring the contextual information of the tweet and comparing it to the contextual information of the preceding and the following tweets [4].

Recently, several deep learning approaches have been proposed. Chen et al. (2016) used LSTM and introduced the attention model based on the user and product information to perform document-level sentiment classification [5]. Jiachen et al. proposed an RNN-based model that incorporated the target-specific information into stance classification [6]. Recurrent

TABLE I: Definition of Each Sentimental Polarity

Label	Definision
Positive	Words expressing approval, support, encouragement, and so on are assigned '+' meaning
Negative	Words expressing denial, opposition, discontent, disbelief, and so on are assigned '−' meaning
Neutral	Mere information that does not express a certain opinion

neural networks and the attention model are effective in case of NLP.

There are three types of sentiment classifications based on the levels of data units: document level, sentence level, and phrase/aspect level.

Document Level: This type of classification analyzes a document that includes several sentences, e.g., product reviews.

Sentence Level: This type of classification focuses on the classification of a sentence. For example, consider a document with the sentences: "The weather is so nice today. However, it will be worse tomorrow." The classifier considers the first sentence to be positive and the second one to be negative.

1) Phrase/Aspect Level: This type of classification analyzes the specified phrases or aspects in a sentence. Usually, the phrases or aspects exhibit contextual sentiment polarities[7].

III. Lexicon Expansion

The existing sentiment lexicons contain strong sentiment words; however, participants often use weak sentiment words. To include weak sentiment words, we conducted an automatic expansion of the sentiment lexicon based on an existing study[8] and applied the expansion to large Japanese datasets.

We used the Japanese Sentiment Polarity Dictionary as a seed sentiment lexicon[9], [10]. It contains 18,520 words, including declinable/indeclinable words. We used word2vec to train a skip-gram model[11] based on the Japanese Wikipedia dataset. Using 200-dimensional word-embedding feature vectors and labels (positive/negative/neutral) from the dictionary, we trained a linear SVM.

After training the SVM, we predicted the sentimental polarities of the remaining words in Wikipedia. The words in the dataset that appeared for more than 10 times were the predicted targets. An SVM classifier assigned labels using this word-embedding.

IV. Named Entity Sentiment Classification

We extracted the named entities using the posts in a web discussion and then classified them by machine learning into 3 types: positive, negative, and neutral. Table I presents a definition of the labels.

A. Pre-processing

First, we prepare a word set and a sentiment set to obtain features from the posts. Each sentence in a post is divided into parts of speech. The postpositions and auxiliary verbs (in Japanese) are discarded. If an auxiliary verb implies negation, the front declinable word is observed to contain the information. A post that passes through this process is referred to as a "word set" in this study. Referring to the sentiment lexicon, we converted each word into a word set, which was assigned a label of positive (1), negative (-1), or neutral (0). If a word contain the information of negation as mentioned above, the polarity was inverted. A post that passes through this process is referred to as a "sentiment set" in this study. An example of the conversion of a post into word and the sentiment sets is depicted in Fig. 1.

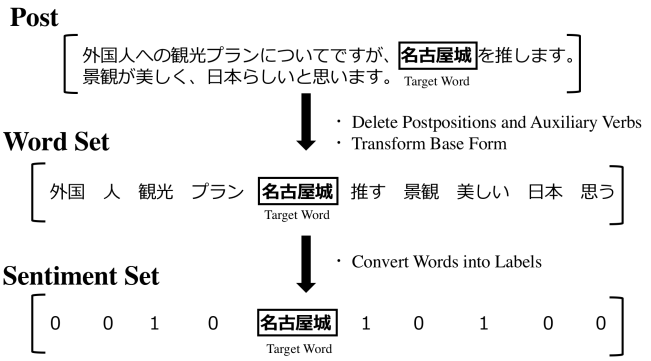

Fig. 1: Word Set and Sentiment Set.

B. Features for Named Entity Sentiment Classification

Named entities and peripheral words are expected to depict co-occurrence relations. Hence, the sentiment polarity of a named entity tends to be dependent on the peripheral word. Here, we define three features.

Feature α: Word N-gram: For each named entity in the word set, we acquire N-gram peripheral words (N includes itself). Algorithm 1 illustrates the pseudocode of word N-gram, and Fig. 2 depicts an example of the case $N = 4$ for which we obtain 4 sets (人，観光，プラン), (観光，プラン，推す), (プラン，推す，景観), and (推す，景観，美しい). Further, we vectorize the number of occurrences of each word. In this example, word N-gram is [外国：0，人：1，観光：2，プラン：3，推す：3，景観：2，美しい：1，日本：0，思う：0].

Algorithm 1 Definition of Feature α: Word N-gram

```
Set Target-index = 0
for i = −N + 1, −N + 2, ..., 0 do
  for j = i, i + 1, ..., i + N − 1 do
    if j ≠ null and j ≠ 0 then
      Counter[w_j](the number of occurrences of w_j)⇐
      Counter[w_j] + 1
    end if
  end for
end for
α ⇐ vectorization Counter
```

Feature β: Polarity N-gram: Polarity N-gram performs similarly to word N-gram for the sentiment set. Thus, the number of occurrences of each sentiment polarity is collected. Alg. 2 exhibits the pseudocode of the polarity N-gram. The number of occurrences is vectorized in the order of [positive, negative, neutral] and can be standardized as follows:

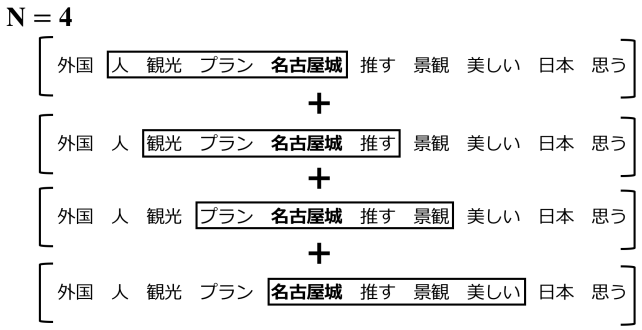

Fig. 2: Search for Peripheral Words.

$$z = \frac{X - \mu}{\sigma} \tag{1}$$

In Eq. 1, X indicates the number of occurrences of each sentiment, μ indicates the average of the number of occurrences, σ indicates the standard deviation, and z indicates an adjuster for the standard normal distribution.

Algorithm 2 Definition of Feature β: Polarity N-gram

```
Set Target-index = 0
positive, negative, neutral ⇐ 0
for i = −N + 1, −N + 2, ..., 0 do
  for j = i, i + 1, ..., i + N − 1 do
    if j ≠ null and j ≠ 0 then
      if p_j = 1 then
        positive ⇐ positive + 1
      else if p_j = −1 then
        negative ⇐ negative + 1
      else if p_j = 0 then
        neutral ⇐ neutral + 1
      end if
    end if
  end for
end for
β ⇐vectorization and standardization positive, negative, and neutral
```

1) Feature γ: Dependency Polarity: Feature β is effective to understand the peripheral polarity. However, peripheral words were not observed to originate from the sentiment polarity at times. Therefore, we used dependency analysis to obtain words that exhibited strong co-occurrence relations with the named entities. By parsing the dependency of each sentence in the posts, the words that exhibited relations with the named entities were detected. Thus, the dependency words were converted, labeled, and vectorized in the same way as performed by the polarity N-gram.

C. Random Forest Classification

We trained the random forest classifier to combine the features of α, β, and γ. The classifier is an ensemble learning method to perform classification, regression, and other tasks that operate by constructing a multitude of decision trees during the training time and by outputting the class that exhibits the mode of the classes (classification) or mean prediction (regression) of the individual trees[12], [13]. Random decision forests are observed to compensate for the decision trees' habit of overfitting to their training set[14].

V. Experiments

A. Settings of Experiments

We compared the performance of our proposed method with the baseline and confirmed the features of our proposed method that exhibited significant effects.

Annotations and Datasets: The named entities are the targets of sentiment classification. Because the online discussion dataset that was used in this study was assumed to be a town meeting, several named entities that were peculiar to the regions were observed. Hence, we focused on those words that were classified as proper nouns by mecab-ipadic-NEologd (a Japanese parser that covers neologisms).

We developed a web discussion corpus with sentiment polarity. For annotation, we developed a web application. Five undergraduate students from the Tokyo University of Agriculture and Technology annotated the dataset with the following labels: positive, negative, neutral, and N/A. A parser extracted inappropriate words at times that exhibited no importance; therefore, the annotators excluded such words. A label that was selected by the majority of the participants was defined to be the correct label. We picked the correct labels in the following order of priority: N/A, positive/negative, and neutral.

Table II presents the details of the online discussion dataset and the results of the annotation. To perform the experiments, 9,036 words (without N/A) were used.

Our proposed method was divided into three patterns: features α, α + β, and α + β + γ. To evaluate the performance of the expanded lexicon, we conducted experiments with and without the expanded lexicon for each pattern. The baseline that was used in this case was the bag-of-words. The experiments were divided into two cases as follows:

I. 3 value classifications: positive, negative, or neutral
II. 2 binary classifications: positive or other and negative or other

Each experiment was evaluated using an 8-fold cross-validation. The precision, recall, and F-1-measure were used for the evaluation metrics. The features α and β depended on the parameter N, which is a range that is used to search for the peripheral words. In the experiments, we defined $N = 20$. The hyper-parameters in the random forest were set as follows: D(the number of max depths in decision trees)= ∞, B(the number of decision trees)= 200, F(the number of selections of features)= $\sqrt{FNUM}$(FNUM indicates the number of features), $Node$(the number of max leaf nodes)= ∞.

Table II presents the web discussion corpus with the sentiment polarity as an unbalanced dataset. Most of the target words were neutral words. Only a few words exhibited negative polarity. Therefore, we conducted undersampling and oversampling before training the random forest classifiers. In experiment I, we defined the number of positives as the standard, while in experiment II, we defined twice the

TABLE II: Dataset and Results of the Annotation

discussion theme	participants	posts	targets	Positive	Negative	Neutral	N/A
Discussion about Nagoya	827	1351	8120	1238	349	5664	869
Environment in Nagoya	20	261	1153	46	43	743	321
Disaster in Nagoya	21	332	1411	44	86	823	458

TABLE III: Sentiment classification results in experiment I

	Positive			Negative			Neutral		
Method	precision	recall	f1	precision	recall	f1	precision	recall	f1
BoW(Baseline)	0.275	0.658	0.388	0.190	0.385	0.252	0.901	0.608	0.726
α	0.294	0.718	0.417	0.279	0.423	0.334	0.907	0.635	0.747
α+β	0.301	0.718	0.424	**0.304**	0.448	**0.356**	0.907	0.646	0.754
α+β+γ	**0.307**	**0.725**	**0.432**	0.291	**0.450**	0.350	**0.910**	0.649	0.758
α+β (+ Lexicon Expansion)	0.304	0.705	0.424	**0.304**	0.411	0.348	0.904	**0.663**	**0.765**
α+β+γ (+ Lexicon Expansion)	0.298	0.711	0.420	**0.304**	0.391	0.339	0.904	0.658	0.762

TABLE IV: Sentiment classification results in experiment II

	Positive			Negative		
Method	precision	recall	f1	precision	recall	f1
BoW(Baseline)	0.316	0.542	0.399	0.156	0.508	0.239
α	0.356	0.602	0.447	0.215	0.532	0.306
α+β	0.382	**0.614**	0.471	0.219	**0.578**	**0.317**
α+β+γ	**0.390**	**0.614**	**0.477**	0.217	0.545	0.310
α+β (+ Lexicon Expansion)	0.372	0.596	0.458	**0.221**	0.539	0.310
α+β+γ (+ Lexicon Expansion)	0.376	0.592	0.460	0.215	0.513	0.302

number of positives/negatives as that defined by the standard. Each label made the uniform standard using undersampling and oversampling. Undersampling was executed by random selection, whereas oversampling was executed by the SMOTE algorithm.

B. Experimental Results

Table III exhibits the results of experiment I. Row 1 depicts the results of the baseline method, whereas rows 2-6 depict our proposed methods. First, all of our proposed methods outperformed the baseline based on all the metrics. The bag-of-words algorithm counted the number of words in every post. If several targets appeared in a post, the bag-of-words algorithm judged them to depict the same features. Hence, the baseline performed unsatisfactorily in case of the online discussion corpus. In contrast, our proposed methods improved the performance by focusing on each target word. Moreover, the feature β improved the results for all the metrics and labels. In particular, the negative's f1-score of α + β improved on only one of α by 2.2%. γ slightly improved the positive's score but performed unsatisfactorily for the negative. The expanded lexicon performed efficiently for the recall of the neutral, whereas the others implied a neutral or negative efficient.

Table IV presents the results of experiment II. As with experiment I, features α and β outperformed the baseline for each label. Comparing the experiment I with II, the binary classification outperformed the three-value classification for the positive. For the negative, the binary classification significantly improved the recall; however, it exhibited worse precision and a worse f-1 score.

C. Discussion

Error Analysis: We analyzed the cause of our method's failure to predict the appended feature γ and the expanded lexicon. Generally, analyzing the dependency was observed to work satisfactorily for well-regulated sentences. In the online discussion dataset, many informal sentences appeared, e.g., an addition (A is good. B is also.) and a split substantive (I recommend. C). Our method was not able to adequately analyze these sentences and the extracted unnecessary words.

The expanded lexicon included unnecessary words among which most exhibited no polarities. For example, "思う (think)" and "言う (say)" were recorded to be positive. Both words are used frequently in web discussions and are observed to interfere with correct training.

Parameter N Analysis: In our proposed methods, tuning the ideal parameter is a significant task. Thus, N parameters within the range of 5 to 100 by each 5 are searched. Fig. 3 depicts the transitions of the f1-scores of the positive under a similar classification as that depicted in experimentII. Both α and α + β tended to improve when the value N was increased. When $N = 95$, α + β outperformed the baseline by 10.9% because our method was able to deal with long sentences as compared to small N.

Fig. 4 depicts the f1-score of the negative words of every N value under a similar classification as that depicted in

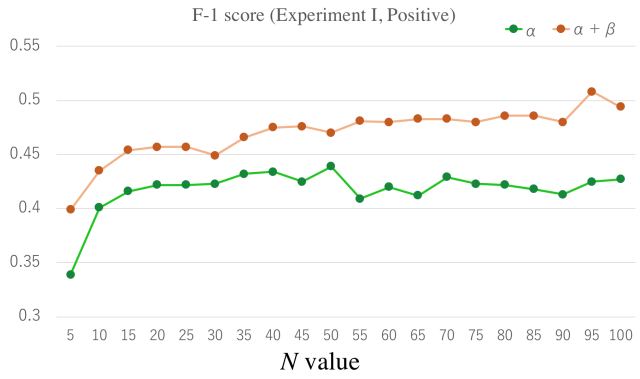

Fig. 3: f1-score of positive words under every N value

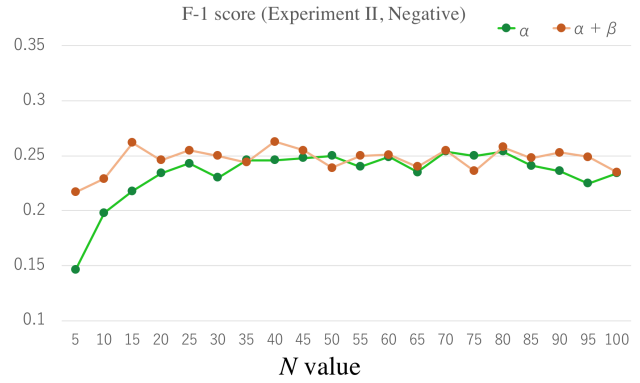

Fig. 4: f1-score of negative words under every N value

experimentII. Both α and $\alpha + \beta$ tended to converge on the values of N from $N = 20$ to 100 because the negative expressions were observed to appear in short sentences. In addition, several negative opinions without direct negative words were observed to appear in the datasets. For example, "Everyone recognizes that Nagoya is the third city, after Tokyo and Osaka. People in Nagoya tend to compare with them." Nagoya is the target that we intended to classify as negative. However, this example did not include any direct negative words and was classified to be neutral. To obtain an appropriate result, we must develop new features that consider the contextual information.

VI. Conclusion

This study proposed a new method to classify sentiment polarities (positive/negative/neutral). First, we expanded the sentiment lexicon using the Japanese Wikipedia dataset as seed data. We used word-embedding as a feature and trained a linear SVM classifier. After expanding the lexicon, we defined three features: word N-gram, polarity N-gram, and dependency polarity. We trained the random forest classifiers to classify the sentiment polarities. The experimental results exhibited that word N-gram and polarity N-gram were efficient to perform this task, whereas the dependency polarity and lexicon expansion were inefficient with regard to the labels.

For future research, we consider to investigate methods to improve some of the details of our methodology as follows.

Lexicon Expansion: For our method, we expanded the sentiment lexicon. However, the lexicon lacked the notion of sentiment strength. In online discussions, there are several words with weak polarity; hence, we must distinguish between weak and strong polarities. Therefore, we will assign sentiment scores of $[-1, 1]$ to each word. We will also eliminate the unnecessary words from the lexicon to obtain more crucial words.

Contextual Features: Each classifier cannot work efficiently when no polarity words are observed to exist. Hence, we will define a few contextual features without referring to the sentiment lexicon. In online discussions, participants reply to others affirming (or contradicting) their stances. We will employ these relations as features.

Tuning Parameters and Settings: Parameter N exhibited a significant influence on the prediction of the correct labels. The ideal parameters for considering a post's length were observed and adjusted in an automatic manner. We will further develop a training method for the unbalanced datasets. To rectify a shortage of datasets with positive/negative words, we will extract positive/negative expressions from other datasets that can be used to further train the classifiers.

Acknowledgment

This work was supported by JST CREST Grant Number JPMJCR15E1, Japan.

References

[1] Yuma Imi, Takayuki Ito, Takanori Ito, and Eizo Hideshima. A large-scale consensus support system called collagree based on online facilitation functions — a real-world application for nagoya next generation total city planning. *Information Processing Society of Japan*, 56(10):1996–2010, oct 2015.

[2] E. Cambria, B. Schuller, Y. Xia, and C. Havasi. New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems*, 28(2):15–21, March 2013.

[3] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[4] Soroush Vosoughi, Helen Zhou, and Deb Roy. Enhanced twitter sentiment classification using contextual information. *CoRR*, abs/1605.05195, 2016.

[5] Huimin Chen, Maosong Sun, Cunchao Tu, Yankai Lin, and Zhiyuan Liu. Neural sentiment classification with user and product attention. In *EMNLP*, 2016.

[6] Jiachen Du, Ruifeng Xu, Yulan He, and Lin Gui. Stance classification with target-specific neural attention. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, (IJCAI-17)*, pages 3988–3994, 2017.

[7] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[8] Roy Bar-Haim, Lilach Edelstein, Charles Jochim, and Noam Slonim. Improving claim stance classification with lexical knowledge expansion and context utilization, 2017.

[9] Nozomi Kobayashi, Kentaro Inui, Yuji Matsumoto, and Kenji Tateishi. Collecting evaluative expressions for opinion extraction. *Journal of Natural Language Processing*, 12(3):203–222, 2005.

[10] Yuji Matsumoto Masahiko Higashiyama, Kentaro Inui. Learning sentiment of nouns from selectional preferences of verbs and adjectives. *Proceedings of the 14th Annual Meeting of the Association for Natural Language Processing*, pages 584–587, 2008.

[11] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[12] Tin Kam Ho. Random decision forests. In *Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Volume 1*, ICDAR '95, pages 278–, Washington, DC, USA, 1995. IEEE Computer Society.

[13] Tin Kam Ho. The random subspace method for constructing decision forests. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(8):832–844, August 1998.

[14] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.