Aggregating User-Centric and Post-Centric Sentiments from Social Media for Topical Stance Prediction

Jenq-Haur Wang
Department of Computer Science and
Information Engineering,
National Taipei University of Technology
Taipei, Taiwan
jhwang@ntut.edu.tw

Kuan-Ting Chen
Joint Credit Information Center
Taipei, Taiwan

rabbitcgt@gmail.com

Abstract

Conventional opinion polls were usually conducted via questionnaires or phone interviews, which are time-consuming and error-prone. With the advances in social networking platforms, it's easier for us to automatically collect and aggregate the overall topical stance for a specific topic. In this paper, we propose to predict topical stances by aggregating user-centric and post-centric sentiments from social media. Firstly, related posts of a given topic are collected from social media and clustered by word embeddings, where major keywords are extracted as the expanded concepts. Then, machine learning methods are used to train sentiment lexicon with word embeddings. Finally, the sentiment scores from user-centric and post-centric views are aggregated as the total stance on the topic. In the experiments on data from online forums, the proposed approach can obtain the best performance with a mean absolute error (MAE) of 0.52% for stance prediction of 2016 Taiwan Presidential Election. This shows the effectiveness of our proposed approach in topical stance aggregation and prediction. Further investigation is needed to evaluate the performance of the proposed method in larger scales.

Keywords: Topical stance detection, Sentiment analysis, Word embeddings, Document clustering

1 Introduction

People usually express their opinions in social occasions with friends and to the public. To know what the general public think about a specific topic, it usually takes much human efforts in designing questionnaires, collecting feedbacks and analyzing them. It's time-consuming and error prone. Depending on the participation of people, there could be not too many effective responses. With the advances of social networking platforms, it's very easy to post articles and reply with comments. For example, Twitter, Facebook, and Instagram are among the most popular social networking sites with different functions. This facilitates users to make online discussions in an immediate way. Given huge amount of social opinions, it would be useful if we can automatically collect and aggregate the general stances from them.

There are some challenges to the problem. Firstly, given the very diverse contents in social media, it would be difficult to obtain the most relevant contents from huge amount of data. Secondly, people might express their opinions in different ways. It would be difficult to extract what they really think about specific topics from very short texts in social media.

Content in a short text is usually limited in scope. Without explaining the ideas and referencing related documents, we might only obtain fragmented terms or named entities just from the sole content of a single post. It might even contain emotional feelings or noises that cannot help us clarify the main idea.

On the other hand, users have different types of activities in addition to posting. For example, most social networking platforms provide mechanisms for making friends, following people or topics that you are interested in, and expressing agreement or disagreement, replying, or commenting on others' posts. These social relations, both explicit and implicit, provide useful clues for understanding what people really think, in addition to what they explicitly mention in post contents. This makes it possible to analyze user opinions by extracting social relations and discovering the major concepts.

In this paper, we propose to aggregate user-centric and post-centric stances for topical stance prediction. Firstly, given the simple topical keyword, we expand the concepts by clustering topic-related posts and comments by their word embeddings, and extract major keywords from each group using word segmentation and named entity recognition methods. Then, given word embeddings, sentiment classification is done by machine learning methods including Naïve Bayes (NB) and Extreme Learning Machines (ELMs) (Huang, 2015). Finally, we aggregate topical stances using both post-centric and user-centric sentiments. In post-centric views, the more positive feedbacks a post gets, the more positive it is regarding the topic. In user-centric views, the more positive comments a user gives, the more positive the user is regarding the topic. By aggregating both post-centric and user-centric sentiments, we are able to analyze the influences of user posts from broader aspects.

In the experiments, we collected data from the most popular online discussion forum in Taiwan called PTT. For sentiment analysis on short texts, we found inconsistent sentiment between user ratings and post contents. After adjustment, ELMs are more stable in sentiment classification performance than Naïve Bayes classifiers. By aggregating stances on three groups of candidates in the 2016 Taiwan Presidential Election to predict the election result, the best performance can be obtained for ELMs with the MAE of 0.52%. This shows the potential of our proposed approach in stance prediction. Further investigation is needed for different types of social media in larger scales.

2   Related Work

Sentiment classification is one of the major techniques for social media analysis and opinion mining. Documents are classified by overall sentiment instead of topic. For example, Pang et al. (2002) first utilized machine learning techniques in learning classifiers for positive and negative movie reviews. They found features as important factors in social media sentiment classification. Conventional bag-of-words models do not distinguish between word orders. Word n-gram models such as bigrams simply consider consecutive words as a unit for representing documents. It's only limited in the local context of words. Nowadays, word embedding models such as Word2Vec (Mikolov et al., 2013) or GloVe

(Pennington et al., 2014) have been used as a more suitable representation of documents, especially for short texts in social media. They utilize neural networks to learn the semantics of words in different contexts. Furthermore, different deep learning methods have been used to automatically learn the features in sentiment classification. For example, Recurrent Neural Networks (RNNs) and their variants such as Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs) are often used to capture long-term dependency in sequential data. They have been successfully applied in sentiment classification of tweets (Wang et al., 2018). Convolutional Neural Networks (CNNs) were originally used in image recognition. With suitable representation of word embeddings in documents, CNNs were also found effective in sentiment classification of tweets (Severyn and Moschitti, 2015).

Based on sentiment classification of a single review or post, it's useful to further determine the *stance* that indicates whether the author is in favor of or against a specific target entity. For example, Mohammad et al. (2017) created the first stance dataset in Twitter, and proposed a stance detection system using Support Vector Machine (SVM) classifiers with character and word n-grams and word embedding features. But the target entity needs to be specified before determining the stance, and it's only based on the tweet content. It's closely related to aspect-based sentiment analysis tasks in SemEval 2014 (Pontiki et al., 2014) and SemEval 2015 (Pontiki et al., 2015).

In addition to the typical stance detection of texts, social media data are often used in determining the polarization in political opinions (Conover et al., 2011) and predicting voting intentions or outcomes in elections (Tumasjan et al., 2010). Instead of detecting the stance of a single user on a specific target, it's useful to derive the stance from the general public on a topic, which we called *topical stance*. For example, in SemEval 2016 topical stance detection contest, MITRE (Zarrella and Marsh, 2016) used LSTM with Word2Vec word embeddings. DeepStance (Vijayaraghavan et al., 2016) used CNN models, while Du et al. (2017) used attention models. Dey et al. (2018) developed a two-phase solution to topical stance detection for Twitter including subjectivity detection and sentiment classification using LSTM with attention. Samih and Darwish (2021) proposed user-level stance detection using

only a few tweets for users by fine-tuning contextualized embedding. As mentioned in the literature (ALDaye and Magdy, 2021), there are usually two levels of stance detection: statement-level, which is simply based on text content, and user-level, which is to predict the stance of a user on the target. Also, there could be three different types of stance detection according to targets: target-specific stance, multi-related target stance, and claim-based stance. In addition to stance detection, research on stance prediction is usually concerned with detecting stances before the event. Most previous studies investigated the micro-level prediction, which estimates the individual user's viewpoint toward a target. For example, Dong et al. (2017) considers joint modeling of content and social interactions for user stance prediction. Darwish et al. (2017) used content and user interactions to calculate user similarity for stance prediction. In this paper, we propose a macro-level approach to stance prediction by aggregating topical stances from post-centric and user-centric views in social media, based on sentiment classification results on very short texts using word embedding features and machine learning methods.

3 The Proposed Method

There are three major modules in the proposed approach: concept expansion, opinion analysis, and stance aggregation. The overall architecture of the proposed approach is illustrated in Fig. 1:
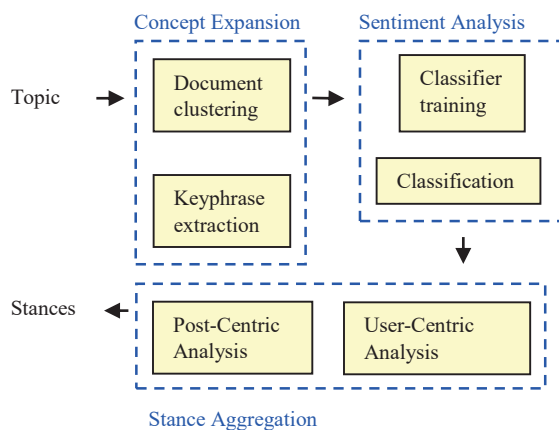

Figure 1. The system architecture of the proposed approach.

As a preprocessing step, given the topical keyword, topic-relevant posts and comments are collected and represented by word embeddings. First, concept expansion is done by clustering the topic-relevant posts and comments, and extracting the keyphrases in each cluster. Then, machine learning methods are utilized to train the classifiers for sentiment classification. Sentiment orientation is then used to calculate the corresponding stances from both post-centric and user-centric points of view. Finally, the stances are aggregated by their linear combination. In the following subsections, we will explain the details.

3.1 Data Representation

Social media contents might be very diverse and noisy. To facilitate more efficient analysis, we routinely crawled all data from the target source media and extracted the corresponding structures from the post-centric and user-centric views and stored in a search engine called Apache Solr for efficient search and analysis. The two different views are described as follows.

In post-centric views, each post consists of the major content and responses from others including replies (or comments), ratings (such as like/dislike), and sharing (such as forwarding, or retweeting, depending on the social platform). These various responses constitute how people think about this post. Generally, the more positive feedbacks a post gets, the more positive it is regarding the topic.

In user-centric views, each user might post an article, and respond to other users' posts, including replies, ratings, and sharing. From these posts and responses, we might be able to observe what he or she thinks about a topic. The more positive comments a user gives, the more positive the user is regarding the topic.

Since we focus on the analysis of text contents and users, we need to correctly identify person names and the concepts of different entities. In this step, we utilize word segmentation and word embedding for the representation of documents.

Word Segmentation: The feature units of documents usually include segmented words or word *n*-grams. In the case of Chinese documents, the definition of words depends on the result of word segmentation since there's no space characters between Chinese characters in a sentence. Usually there are two major problems in word segmentation: ambiguity and unknown words. To resolve the issues, lexicon-based and machine learning methods are often used. The size and quality of the lexicon determines the accuracy of the words segmented.

In this paper, we utilize a popular open source tool called Ansj [1] for word segmentation. It's a word-based generative model based on a bi-gram model which is a first-order Markov chain. That is, each character is assumed to be dependent on it previous character. The word candidate that generates the maximum union probability will be selected. Since the first-order Markov chain model might not be able to achieve high recalls for unknown words, a Hidden Markov Model (HMM)-based method (Zhang et al., 2003) is used for identifying out-of-vocabulary words. From our observation, this model can generate better segmentation results for person names.

Word Embedding: After feature units are identified by word segmentation technique, we need to find an appropriate representation for documents. Conventional bag-of-words model is not efficient due to the following reasons. Firstly, it's high dimensional and very sparse. Secondly, word orders are completely ignored, which generates ambiguous semantic meanings. In order to better capture semantics in documents, we utilize word embedding models such as Word2Vec. Through the training of contexts from large amounts of documents, we can better predict the contexts of a word or predict a word from its context. Also, it's fixed dimensional which make the machine learning algorithms easier to calculate. Specifically, we represent a document d_j by its component words $w_1, ..., w_n$ after word segmentation as follows.

$$V(d_j) = \frac{\sum_{i=1}^{n} V(w_i)}{n} \qquad (1)$$

where $V(w_i)$ is the vector representation of each word w_i.

3.2 Concept Expansion

People might describe the same idea in different terms. Given a single term, the semantics are usually limited. For example, people searching for information about "presidential election" might be interested in the candidates, their names, and election results. To understand what people think about a topic, we need to collect their opinions on all related concepts. In this paper, we utilize document clustering and keyword extraction for concept expansion. Firstly, initial topic was used to

collect related documents and grouped into clusters. Then, keywords are extracted from each cluster and the top-frequent keywords are kept as the major concepts. In order to improve the informativeness of the concepts extracted, we repeat the same process by using these keywords to collect related documents for augmenting the keywords until it converges to the number of concepts we need.

Document Clustering: To obtain all related concepts, we first start with the topic word t. By using search engines such as Google, we get the search result pages $P(t)$. Then, we use the same word embedding models to represent each document p_i in its vector form $V(p_i)$, from which K-means clustering algorithm is used to separate them into K groups. These correspond to the different groups of documents for different concepts. The selection of K depends on how many possible concepts might be related to this topic. In the example of presidential election, the number of clusters K might correspond to the different groups of candidates in the election.

Keyword Extraction: After documents are grouped by their embeddings, the next issue is how to identify the corresponding concepts for each group. Firstly, we apply the same word segmentation technique Ansj on all documents in each cluster to identify the corresponding keywords. Then, we need to discover the named entities since they are often the most important candidates for the major concepts. In this paper, we use Stanford Named Entity Recognizer (Finkel et al., 2005) which employed conditional random fields (CRFs) to recognize the named entities in probabilistic ways.

3.3 Sentiment Analysis

To understand the opinion orientation of each post, we first train the sentiment lexicon from our training data. Then, we use ELMs (Huang, 2015) to classify the sentiment into positive, neutral, and negative, and compare with a simple baseline Naïve Bayes classifier.

The structure of ELMs is a neural network with single hidden layer. The major difference of ELMs from common neural networks is its lack of back

---

[1] https://github.com/NLPchina/ansj_seg

propagation phase to reduce training errors. Thus, it's much faster than conventional neural networks. The architecture of ELMs is shown in Figure 2.
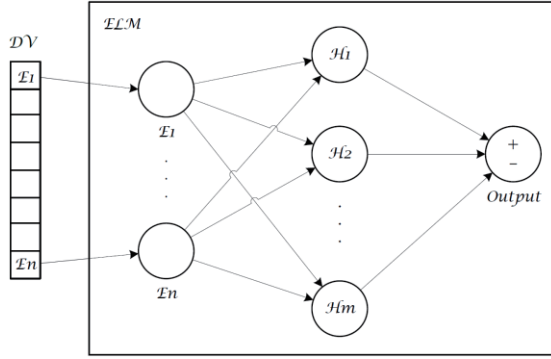

Figure 2. The architecture of Extreme Learning Machines (ELMs).

As shown in Figure 2, ELMs need numeric data as input just like common neural networks, we apply the word embedding models such as Word2Vec on each document. The number of neurons in input layer corresponds to the dimension of word embeddings. The number of neurons in output layer is one, which simply classifies each document as positive or negative. In this paper, the number of neurons in the hidden layer is set as 200.

User ratings might not reflect the actual sentiment orientation of users, for example, in the case of sarcasm. From our observation, people are more proactive in negative ratings, and they might give negative replies or comments with a positive rating. This is possible in some cases where people respond to some people or events instead of the document itself. It was also indicated in related work (Heath, 1996). In order to fix this phenomenon, we adjust the user ratings by combining with sentiment classification of replies or comments as follows.

$$Class(d_j) = \begin{cases} 1 & if\ r(d_j) > 0\ and\ Sent(d_j) > 0 \\ 0 & if\ r(d_j) = 0 \\ -1 & othrwise \end{cases} \quad (2)$$

Where $r(d_j)$ is the user rating such as like or dislike, and $Sent(d_j)$ is the sentiment orientation of the document.

3.4 Stance Aggregation

Given user input topic t and the number of concepts K, we obtain related concepts $Q_1,..., Q_K$. For each concept Q_i, we obtain the set of all the related documents D_i and the set of all the related users U_i.

Then, we conduct analyses in two different views as follows.

Post-Centric Stance: For a given concept Q_i, we have the set of all related documents D_i. For each document d_j in D_i, instead of using the sentiment orientation defined previously, we first calculate the aggregate score from all the comments.
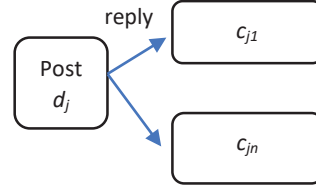

Figure 3. The idea of post-centric stance.

The idea is that: the orientation of a post is determined by the orientation of its comments.

$$S_{post}(d_j) = \sum_{p_k \in Comm(d_j)} class(p_k) \quad (3)$$

Where $Comm(d_j)$ denotes all the comments p_k for post d_j, and $class(p_k)$ is defined as in Eq.(2). The higher the score, the more positive people judge on this post.

To accumulate all the scores into the overall post stance for the concept Q_i, we define the *post-centric stance* as follows:

$$Stance_{post}(Q_i) = \frac{|\{d_j \in D_i | S_{post}(d_j) > 0\}|}{|\{d_j \in D_i | S_{post}(d_j) != 0\}|} \quad (4)$$

where D_i is the set of all documents related to concept Q_i.

User-Centric Stance: For a given concept Q_i, we also have the set of all related users U_i who posted or comments on posts in related concepts. For each user u_j in U_i, we consider the aggregate score from all the posts generated by him or her.
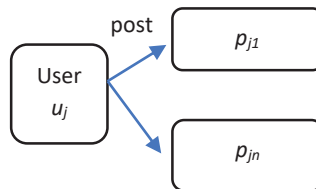

Figure 4. The idea of user-centric stance.

The idea is that: the stance of a user is determined by the orientation of his/her posts.

$$S_{user}(u_j) = \sum_{p_k \in Posts(u_j)} class(p_k) \quad (5)$$

Where *Posts(u_j)* denotes all the posts p_k by user u_j, and *class(p_k)* is defined as in Eq.(2). The higher the score, the more positive people judge on this user.

To accumulate all the scores into the overall user stance for the concept Q_i, we define the *user-centric stance* as follows:

$$Stance_{user}(Q_i) = \frac{|\{u_j \in U_i | S_{user}(u_j) > 0\}|}{|\{u_j \in U_i | S_{user}(u_j) != 0\}|} \quad (6)$$

where U_i is the set of all users related to concept Q_i.

Aggregate Stance: For each given concept Q_i, the data size might be different in terms of related posts and users. To give a more balanced aggregation, we can further consider the weights for posts and users as follows.

$$w_{post}(Q_i) = \frac{|D_i|}{\sum_{j=1}^{K} |D_j|} \quad (7)$$

Where D_i is the set of all documents related to concept Q_i.

$$w_{user}(Q_i) = \frac{|U_i|}{\sum_{j=1}^{K} |U_j|} \quad (8)$$

Where U_i is the set of all users related to concept Q_i. Thus, the weighted post-centric and user-centric stances for a given concept Q_i can be defined as follows:

$$WS_{post}(Q_i) = w_{post}(Q_i) * Stance_{post}(Q_i) \quad (9)$$

$$WS_{user}(Q_i) = w_{user}(Q_i) * Stance_{user}(Q_i) \quad (10)$$

Since we calculate the post-centric and user-centric stances individually, to further allow for the relative importance between the two views, we finally assign a weight α for linear combination for the total stance as follows.

$$Stance_{total}(Q_i) = \alpha * WS_{user}(Q_i) + (1 - \alpha) * WS_{post}(Q_i) \quad (11)$$

The idea is that: the higher the total stance for a concept, the more positive people give feedbacks to this concept.

4 Experiments

In our experiments, we designed our customized crawler in Telnet to collect data from the most popular online discussion forum called PTT. During Feb. 2015 an Jun. 2016, a total of 881,322 documents in Chinese was collected in the discussion board of "Gossip". The number of users participated in these posts is 60,018.

4.1 The Effects of Concept Expansion

To verify the effects of concept expansion, we selected a number of topics. The results of concept expansion are as follows:

Topic	Initial Concepts	Added Concepts
Presidential election (總統大選)	Chu Li-luan (朱立倫)、Tsai Ing-wen (蔡英文)、Soong Chu-yu (宋楚瑜)、Chen Chien-Jen (陳建仁)	Tsai-Chen ticket (英仁配)、Soong-Hsu ticket (宋瑩配)、Chu-Wang ticket (朱玄配)
Ma-Xi meeting (馬習會)	Ma Ying-jeou (馬英九)、Tsai Ing-wen (蔡英文)、Xi Jinping (習近平)、Ma-Xi meeting (馬習會)	Chu Li-luan (朱立倫)、Zhang Zhijun (張志軍)、Hsia Li-yan (夏立言)

Table 1: Example results of concept expansion.

As shown in Table 1, we can see more relevant concepts can be extracted. For example, for 2016 presidential election, the candidates and the running mates can also be discovered. In the case of Ma-Xi meeting, the major participants from both sides including the Minister of the Mainland Affairs Council Hsia Li-yan and Taiwan Affairs Office Director Zhang Zhijun. Also, the KMT chairman Chu Li-luan met Xi the year before in the 2015 Xi-Chu meeting (朱習會). From these examples, we can see more related concepts are helpful to the representation of documents.

4.2 The Effects of Sentiment Analysis

After concept expansion, we need to conduct sentiment analysis for text documents. We selected a number of concepts to test the performance. The ground truth is taken from user ratings such as likes or dislikes in each comment. The results of sentiment analysis using Naïve Bayes are as follows:

Concept	Recall	Precision	F-score	Accuracy
Tsai Ing-wen (蔡英文)	0.717	0.982	0.829	0.728
Chu Li-luan (朱立倫)	0.860	0.464	0.603	0.696

Soong Chu-yu (宋楚瑜)	0.780	0.921	0.845	0.799
Average	0.786	0.789	0.759	0.741

Table 2: Example results of sentiment analysis using Naïve Bayes.

As shown in Table 2, we can see a good average accuracy of 0.741 and F-score of 0.759 for Naïve Bayes. However, since data is imbalanced, the precision value of some concepts are as low as 0.464. This is not stable. Next, we show the results of sentiment analysis using ELMs.

Concept	Recall	Precision	F-score	Accuracy
Tsai Ing-wen (蔡英文)	0.744	0.811	0.776	0.686
Chu Li-luan (朱立倫)	0.629	0.651	0.640	0.636
Soong Chu-yu (宋楚瑜)	0.667	0.792	0.724	0.643
Average	0.680	0.751	0.713	0.655

Table 3: Example results of sentiment analysis using ELMs.

As shown in Table 3, we can see an average accuracy of 0.655 and F-score of 0.713 for ELMs. Comparing to Naïve Bayes, we can see lower accuracies, but more stable precision values and F-measures across different concepts. Given more training data, NB is able to learn the probabilistic distributions. ELM cannot reduce the error with back propagation, which gives much lower recalls. The precision values are only slightly affected. We will analyze the reasons as follows.

There are several possible reasons for the mismatch between user ratings and post content sentiments.

The first possible case of incorrect classification is a "false positive". There are many cases when the post content explains the support of one new candidate, but the opinions are against the current officers. That's why we see positive user ratings (for the new candidate), but negative content sentiments (against the current officers). If we conduct sentiment analysis on the contents, they are correctly classified as negative, which is different from the ground truth of positive.

The second example case of misclassification is when a government agency post content criticizing candidate Tsai. Users gave negative ratings against this government post, but positive content in favor

of the candidate. That's another type of mismatch for "false negatives".

To show the effects of these misclassification, we selected a part of the posts from the same concepts and manually adjust the labels of two types of misclassified instances.

Method	Recall	Precision	F-score	Accuracy
NB	0.961	0.695	0.807	0.680
NB-adj	0.722	0.851	0.781	**0.782**
ELM	0.972	0.738	0.839	0.728
ELM-adj	0.570	0.910	0.701	0.646

Table 4: Performance comparison of NB and ELM before and after adjustment of Type-1 errors.

As shown in Table 4, we can observe the performance improvement for NB in terms of accuracy. Specifically, since false positives are greatly reduced for both NB and ELM, precision values are greatly improved. At the same time, false negatives are increased much more for ELM, which gives lower recall. The best performance can be seen for NB after adjusting Type-1 errors.

Method	Recall	Precision	F-score	Accuracy
NB	0.174	0.923	0.293	0.389
NB-adj	0.986	0.723	**0.834**	**0.716**
ELM	0.193	0.846	0.314	0.495
ELM-adj	0.982	0.596	0.742	0.589

Table 5: Performance comparison of NB and ELM before and after adjustment of Type-2 errors.

As shown in Table 5, we can observe the performance improvement for both NB and ELM. Specifically, since false negatives are greatly reduced for both NB and ELM, recall values are greatly improved. At the same time, false positives are increased, which gives lower precision. Although ELM can also be improved, the best performance can be seen for NB after adjusting Type-2 errors.

These results are simply sampled from selected topics, it could not reflect the overall performance. But we can see the advantage of adjusting Type-1 and Type-2 errors, which are very common in social media posts, especially for the discussion forum PTT. The sarcastic phenomenon on political issues among social network users have much impact on the sentiment analysis results.

4.3 The Effects of Post-centric vs. User-centric Stance Detection

In this experiment, we want to verify the effects of post-centric and user-centric stance detection. Here, we focus on the prediction of 2016 presidential election in Taiwan by the two views of stance detection using NB and ELM classifiers and manually adjusted ratings, which are denoted as Post-NB, Post-ELM, User-NB, User-ELM, respectively. Then, we have two baselines: Post-Baseline, and User-Baseline, which simply use statistics of user ratings as the baseline for post-centric and user-centric, respectively.

In this experiment, the topic "presidential election" can be expanded into the three candidates, who got the percentages of final votes: Chu-Wang (31.04%), Tsai-Chen (56.12%), and Soong-Hsu (12.84%). These are considered as the ground truth. Firstly, we compared the mean absolute error (MAE) as follows.

Method	MAE -Tsai	MAE -Chu	MAE-Soong	MAE -avg.
Post-Baseline	6.45	10.14	3.69	6.76
Post-NB	6.12	9.76	3.63	6.50
Post-ELM	1.25	2.99	4.24	**2.83**
User-Baseline	8.01	8.33	0.33	5.56
User-NB	0.38	0.81	1.19	**0.79**
User-ELM	0.58	2.01	1.43	1.34

Table 6: Performance comparison of election result prediction for both post-centric and user-centric views.

As shown in Table 6, we can see the best post-centric result is Post-ELM with a MAE of 2.83%, and the best user-centric result is User-NB with a MAE of 0.79%. For each method, we can obtain better performance for user-centric views.

4.4 The Effects of Stance Aggregation

Next, we further determine the stance aggregation using different weights for post-centric and user-centric results. We compared the better results as shown previously with the aggregated results. From our observation, better MAE values can be obtained when α is 0.7-0.9, we show the result when α = 0.7 as follows.

Method	MAE -Tsai	MAE -Chu	MAE-Soong	MAE -avg.
Aggregate-Baseline	7.54	8.88	1.34	5.92
Aggregate-NB	2.10	2.36	0.26	1.57
Aggregate-ELM	0.78	0.51	0.27	**0.52**

Table 7: Performance comparison of election result prediction when α = 0.7 in stance aggregation.

As shown in Table 7, we can observe the best performance for ELMs in predicting the percentage of votes for three candidates. Specifically, when aggregating stances using ELMs, the best MAE of 0.52% can be obtained. This shows the potential of the proposed approach to topical stance aggregation from post-centric and user-centric sentiments.

5 Discussions

From our experimental results, there are some observations:

- Firstly, from our observations on sentiment classification of PTT data, we found Type-1 and Type-2 errors that frequently occurred in posts. Users might give positive ratings with negative contents, or vice versa. After adjusting these errors, the performance of sentiment classification can be improved for both ELM and NB.

- Secondly, we consider two different views of stance detection: post-centric and user-centric. User-centric stance detection works better than post-centric, especially for Naïve Bayes.

- Finally, we validated the effects of stance aggregation by the weighted sum of both user-centric and post-centric stances. The best prediction performance with MAE of 0.52% can be obtained. It shows the potential of our proposed approach to stance prediction.

6 Conclusions

In this paper, we have proposed to aggregate post-centric and user-centric sentiments from social media for stance detection. Firstly, we performed concept expansion to obtain the related concepts

for the given topic. Secondly, we trained classifiers such as Naïve Bayes and Extreme Learning Machines for sentiment classification. Finally, we proposed a potential way of calculating the individual influences from comments for posts and posts from each user, and aggregating to obtain the total stance for the topic. From our experimental results, we can see a good performance with the best MAE of 0.52% when we aggregate stances estimated using ELMs. This shows the potential of our proposed approach in topic-specific opinion mining and stance detection. Further investigations are needed to evaluate our proposed approach in different topic domains.

Acknowledgments

References

Abeer ALDaye, Walid Magdy, 2021. Stance detection on social media: State of the art and trends, *Information Processing & Management*, 58(4).

Michael Conover, Jacob Ratkiewicz, Matthew R. Francisco, Bruno Goncalves, Filippo Menczer, and Alessandro Flammini. 2011. Political polarization on Twitter. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, pp. 89–96.

Kareem Darwish, Walid Magdy, Tahar Zanouda, 2017. Improved Stance Prediction in a User Similarity Feature Space. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2017)*, pp.145–148.

Kuntal Dey, Ritvik Shrivastava, and Saroj Kaushik. 2018. Topical Stance Detection for Twitter: A Two-Phase LSTM Model Using Attention, In *Proceedings of the 40th European Conference on Information Retrieval (ECIR 2018)*, pp. 529–536.

Rui Dong, Yizhou Sun, Lu Wang, Yupeng Gu, Yuan Zhong, 2017. Weakly-Guided User Stance Prediction via Joint Modeling of Content and Social Interaction. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM 2017)*, pp.1249–1258.

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

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pp. 363-370.

Chip Heath, 1996. Do People Prefer to Pass Along Good or Bad News? Valence and Relevance as Predictors of Transmission Propensity. *Organizational Behavior and Human Decision Processes*, 68 (2): 79–94.

Guang-Bin Huang, 2015. What are extreme learning machines? Filling the gap between Frank Rosenblatt's dream and john von Neumann's puzzle. *Cognitive Computation*, 7(3): 263–278.

Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, 2013. Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations 2013 Workshop*.

Saif M. Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko, 2017. Stance and Sentiment in Tweets, *ACM Transactions on Internet Technology*, 17(3):26.

Bo Pang, Lillian Lee, Shivakumar Vaithyanathan, 2002. Thumbs up? Sentiment Classification using Machine Learning Techniques, In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pp.79-86.

Jeffrey Pennington, Richard Socher, Christopher D. Manning, 2014. GloVe: global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pp. 1532–1543.

Maria Pontiki, Dimitrios Galanis, Haris Papageogiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 Task 12: Aspect based sentiment analysis. In *Proceedings of the International Workshop on Semantic Evaluation*.

Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 Task 4: Aspect based sentiment analysis. In *Proceedings of the International Workshop on Semantic Evaluation*.

Younes Samih and Kareem Darwish, 2021. A Few Topical Tweets are Enough for Effective User Stance Detection, In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2637–2646.

Aliaksei Severyn, Alessandro Moschitti, 2015. Twitter Sentiment Analysis with Deep Convolutional Neural Networks, In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2015)*, pp.959–962.

Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner, Isabell M. Welpe, 2010. Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment, In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media (ICWSM)*, pp.178-185.

Prashanth Vijayaraghavan, Ivan Sysoev, Soroush Vosoughi and Deb Roy, 2016. DeepStance at SemEval-2016 Task 6: Detecting Stance in Tweets Using Character and Word-Level CNNs, In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*.

Jenq-Haur Wang, Ting-Wei Liu, Xiong Luo, and Long Wang, 2018. An LSTM Approach to Short Text Sentiment Classification with Word Embeddings, In *Proceedings of the 30th annual Conference on Computational Linguistics and Speech Processing (ROCLING 2018)*, pp.214-223.

Guido Zarrella and Amy Marsh, 2016. Mitre at SemEval-2016 Task 6: Transfer Learning for Stance Detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*.

Hua-Ping Zhang, Hong-Kui Yu, De-Yi Xiong, Qun Liu, 2003. HHMM-based Chinese lexical analyzer ICTCLAS, In *Proceedings of the second SIGHAN workshop on Chinese language processing (SIGHAN 2003)*, pp.184–187.

Meishan Zhang, Yue Zhang, Duy-Tin Vo, 2016. Gated Neural Networks for Targeted Sentiment Analysis, In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*, pp.3087-3093.