A topic enhanced approach to detecting multiple standpoints in web texts

Junjie Lin [a,b], Qingchao Kong [a,b,*], Wenji Mao [a,b], Lei Wang [a,b]

[a] State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[b] University of Chinese Academy of Sciences, Beijing 101408, China

A R T I C L E   I N F O

A B S T R A C T

Internet has become the most popular platform for people to exchange opinions and express stances. The stances implied in web texts indicate people's fundamental beliefs and viewpoints. Understanding the stances people take is beneficial and critical for many security and business related applications, such as policy design, emergency response and marketing management. Most previous work on stance detection focuses on identifying the supportive or unsupportive attitudes towards a specific target. However, another important type of stance detection, i.e. multiple standpoint detection, has been largely ignored. Multiple standpoint detection aims to identify the distinct standpoints people hold among multiple parties, which reflects people's intrinsic values and judgments. When expressing standpoints, people tend to discuss diverse topics, and the word usage in the topics of different standpoints often varies a lot. As topics can provide latent information for identifying various standpoints, in this paper, we propose a topic-based approach to detecting multiple standpoints in Web texts, by enhancing generative classification model as well as feature representation of texts. In addition, we develop an adaptive process to determine parameter values in our approach automatically. Experimental studies on several real-world datasets verify the effectiveness of our proposed approach in detecting multiple standpoints.

© 2019 Elsevier Inc. All rights reserved.

1. Introduction

People are inclined to exchange opinions and express their stances on the Internet. The different stances people take indicate their fundamental beliefs and viewpoints, and provide valuable information for many application domains. For example, stance detection is critical for governments and public sectors to analyze and understand public attitudes for policy design [8], decision making and emergency response. It is also beneficial for many business related applications to mine people's inclinations towards certain products and make marketing campaigns more effective.

Most traditional work on stance detection solely focuses on determining the supportive or unsupportive attitudes towards one certain entity or topic [1–3,5,7,9–11,13–16,19,22,23,28,32–34,36,42–44,46,48], without considering the real viewpoint people hold among multiple standpoints. As this distinct standpoint reveals people's intrinsic values and judgments,

---

it is vital to mine people's standpoints instead of their attitudes. For standpoint detection, the existing work usually adopt probabilistic graphical models to mine standpoint-related expressions and identify standpoints in texts [21,25,38,39].

However, these previous research only focuses on the classification of two-sided standpoints (e.g. Palestine or Israel). In contrast, multiple standpoint detection as a more general research question which aims to identify different standpoints people hold among multiple parties or targets has been largely ignored in previous research. For instance, during the 2016 American presidential election, people may stand on the side of Hillary Clinton, Bernie Sanders, Donald Trump, Ted Cruz or other candidates.

Mining multiple standpoints in Web texts can help understand what people think and believe, and facilitate a more in-depth analysis of people's intentions and motivations behind their behaviors. However, in multiple standpoint detection, the computational methods developed for traditional stance detection are not directly applicable. Even for the case of two standpoints, it cannot be guaranteed that a person manifesting unsupportive attitude towards a party takes the side of the other party.

In this paper, we address the problem of multiple standpoint detection. As people with distinct standpoints often talk about different topics, the latent topics in Web texts can provide distinguishable information for standpoint detection. Previous research has used topic-based approach for standpoint detection [21,25,38,39]. However, these methods only focus on two-sided standpoint detection, and are mostly in an unsupervised manner. Motivated by this, we propose a topic enhanced approach to multiple standpoint detection. Different from the previous topic-based methods for standpoint detection [21,25,38,39], we propose a supervised method for multiple standpoint detection, which adopts the latent topic information of each standpoint to enhance a generative classification model and the feature representation of texts respectively. We also develop an adaptive process to determine parameter values in our approach for different input texts.

Our work has made several contributions. We first address the problem of multiple standpoint detection via a topic enhanced approach. For a more fine-grained identification of multiple standpoints, we develop two topic enhanced methods by encoding standpoint-related topic information into a generative classification model and text feature representation respectively. We finally validate our proposed approach using three real datasets.

The rest of the paper is organized as follows. Section 2 introduces the related work on traditional stance detection and standpoint detection. Section 3 describes our proposed topic enhanced approach to multiple standpoint detection. In Section 4, we conduct experimental studies and analyze the experimental results. Section 5 concludes our work and discusses some future research directions.

2. Related work

There are two lines of work related to multiple standpoint detection, traditional stance detection and the two-sided standpoint detection.

2.1. Traditional stance detection

Stance detection, a related research field to sentiment analysis, has attracted much attention in recent years. While sentiment analysis focuses on identifying the positive, negative or neutral emotions expressed in texts [26,27,29], stance detection aims to classify the stances of texts towards a certain discussion topic or entity as supportive, unsupportive or none [1–3,5,7,9–11,13–16,19,22,23,28,32–34,36,42–44,46,48]. The stances people take can provide more in-depth information for user behavior analysis than sentiments, especially in social events. Besides, as people may use positive or negative language to express the same stance, sentiment information is beneficial but not sufficient for stance detection [23].

Sentiment analysis can be performed at word level, sentence level or document level. Existing methods mainly leverage the lexical information in texts for sentiment analysis, and can be categorized as lexicon based methods [17,40] and machine learning based methods [12,24]. Lexicon based methods determine sentiments by the polarity and intensity of sentiment words in sentiment lexicons, while machine learning based methods usually extract words, syntactic and semantic information as features, and train sentiment classifiers on the corpus with sentiment labels. In particular, the sentiments in texts are also used as additional information for stance detection [35,36].

Among traditional work on stance detection, machine learning techniques play an important role. Most work applies supervised classification models and takes lexical, syntactic and semantic information as features for stance detection [1,14,23,32]. The applied classifiers include Naïve Bayes model, Support Vector Machines (SVM), Logistic Regression etc., and commonly used features include n-grams, syntactic dependencies and statistics information of texts (such as the number of words and punctuations). For example, the early work by Abbott et al. [1] uses word n-grams, generalized syntactic dependencies and discourse markers (which are the initial 1∼3 n-grams of sentences) etc. as features and trains Naïve Bayes and JRip classifiers to determine the stances of discussion posts. Sentiment features and arguing expressions have also been used for stance detection [32]. Besides, Hasan and Ng [14] incorporate frame-semantic features with traditional linguistic features to classify the stances of discussion posts, and find that they help improve classification performance. Recently, Mohammad et al. [23] show that Support Vector Machines with word and character n-grams features outperforms all other methods in Task 6A of SemEval2016 (supervised stance detection of tweets). For machine learning based methods, the performance of stance detection mainly relies on high-quality training data as well as effective features which are distinguishable for different stances.

In addition to linguistic information, the relations between discussion posts have also been used to improve the performance of stance detection [2,10,11,15,19,33,33,42]. To this end, graphical models and machine learning techniques are commonly applied. For the posts in debate forums, Walker et al. [42] construct a graph based on explicit agreement and disagreement relations between posts, and apply MaxCut algorithm to group the posts belonging to the same stance. Anand et al. [2] first recognize implicit rebuttal links between posts and then expand the feature set of stance classification with contextual features derived from parent posts. Hasan and Ng [15] train a Conditional Random Field model to label the stance of each discussion post in a post sequence. Besides, Probabilistic Soft Logic and Hinge-loss Markov Random Fields have also been applied to take into account the relations between authors and posts for stance classification [10,33,33]. As people's stances towards a certain topic could be indicated by their stances towards other related topics, some work models the interrelations between people's stances towards different topics, and identifies the stances under various topics collectively [11,19]. Although the relations between posts are helpful for stance detection, they are often hard to acquire or recognize, thus their application is limited to some extent.

In recent years, some work employs deep learning models for stance detection [3,7,9,31,36,44,46]. Among these methods, Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) are the most widely used deep neural network architectures. For example, Wei et al. [44] train a CNN with softmax output layer to classify the stances of tweets, which ranks first in Task 6B of SemEval2016 (weakly-supervised stance detection of tweets). Chen and Ku [7] take the comments and "likers" of Facebook posts as additional inputs and train a CNN to jointly acquire the embeddings of posts, users and comments for stance classification. Augenstein and Rocktaschel [3] encode tweets and stance targets via a bi-conditional RNN. The encoding of tweets is trained upon specific stance targets and thus more effective in stance detection. Recently, many variants of attention-based RNN are proposed to distinguish the importance of different words for stance detection [9,36]. To take full advantage of both CNN and RNN, Zhou et al. [46] propose a bi-directional GRU-CNN structure to mine more fine-grained features for classifying stances. These deep learning models usually need large quantities of labeled data to achieve good performance, and the parameter tuning is often time-consuming and ad hoc.

Another line of related work applies sentiment analysis to acquire the positive or negative sentiments towards a specific entity [18], and uses them for entity-level stance detection [13,28]. For example, Hammer et al. [13] propose a word-based and a dependency-based method to calculate the distances between sentiment words and stance target words so as to analyze the sentiments towards specific entities. For stance detection about certain topics, sentiment information is not sufficient because people may express the same stance towards a topic by using negative or positive language [23]. Other work on traditional stance detection considers issues such as the intensity of stance expressions [43], the joint model of stance and reasons or arguments [5,16] and rumor stance classification [6,22,41,47,48] etc.

2.2. Standpoint detection

Traditional work on stance detection focuses on identifying people's external attitudes towards different topics or entities instead of mining the internal standpoints people hold among multiple parties. The recent work on standpoint detection [21,25,38,39] (also called viewpoint discovery) extends traditional stance detection and aims to determine the standpoints of texts from two candidate parties. Most of these work model different word distributions under various latent topics in an unsupervised manner to identify standpoints [25,38,39]. The standpoint-related, topic-related, sentiment-related information as well as other supplementary information are jointly modeled to detect the standpoints in texts. Trabelsi and Zaiane [39] propose a Joint Topic Viewpoint probabilistic model to mine the underlying arguing expressions for viewpoint identification. Qiu and Jiang [25] further capture viewpoint specific topic preference, user identity and user interactions in their proposed latent variable model. Similarly, Thonet et al. [38] propose a topic model which unifies viewpoint, topic and opinion discovery. Their model distinguishes topic words and viewpoint specific opinion words so as to better discriminate different viewpoints. These unsupervised models leverage no explicit standpoint information in the training process, which leads to the degradation of their performances on standpoint identification.

For supervised standpoint detection, the only closely related work is done by Lin et al. [21]. Based on the consideration that some sentences in a document are not related to author's stances, they propose a Latent Sentence Perspective Model (LSPM) to recognize standpoint-related sentences and leverage the word distributions of different standpoints for identification. However, their model is merely based on literal words and neglects the latent topic information in texts.

Existing work on standpoint detection [21,25,38,39] only addresses the problem of binary standpoint classification. A more general research issue, multiple stance detection, which identifies different standpoints (or viewpoints) of multiple parties, has been paid little attention in previous research. Previous literature has stated that different standpoints have distinct dominant topics, and the word usage in expressing standpoints under different topics often varies a lot [25]. Thus in this paper, we leverage topic information to enhance multiple standpoint detection. To acquire the latent topics in texts, a lot of topic models (e.g. Latent Dirichlet allocation [4], Non-negative matrix factorization [20]) have been proposed, most of which aim to mine document-topic distributions and topic-term distributions from a text collection in an unsupervised manner. As topic-term distributions of standpoint-specific topics are helpful for identifying the corresponding standpoint, we use the topic-term distributions of each single standpoint to enhance standpoint classifier and text feature representation respectively. By leveraging the latent topic information, our proposed approach can achieve better performance compared to the related methods.
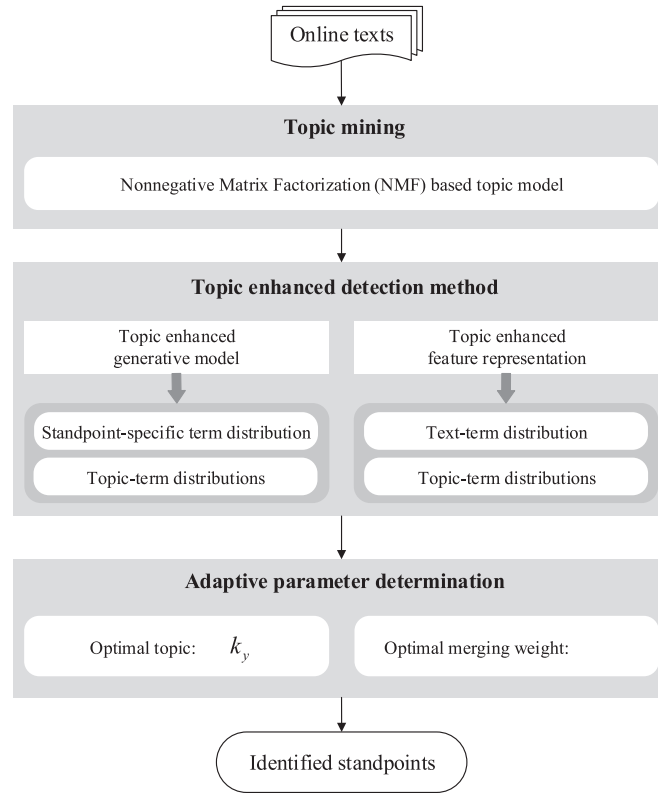
Fig. 1. Main process of the topic enhanced approach to multiple standpoint detection.

3. Proposed approach

We formularize the problem of multiple standpoint detection as follows. Given a labeled text corpus $D = \{(d_i, y_i) | y_i \in Y, \ i = 1, 2, \ldots, N\}$, the multiple standpoint detection task is to construct a standpoint classification model $f(\boldsymbol{x}; \boldsymbol{\theta})$ which outputs the standpoint label y' ($y' \in Y$) for an input text d'. Note that d_i denotes the i th text in the corpus, y_i denotes the standpoint label of d_i, Y denotes the set of standpoint labels whose size can be two or larger, N is the size of the corpus, $\boldsymbol{x}$ denotes the feature representation of texts and $\boldsymbol{\theta}$ denotes model parameters.

To solve the multiple standpoint detection problem, we propose a topic enhanced approach, which first mines the latent topic information in texts. In order to facilitate a more fine-grained treatment of standpoint identification, we then leverage the topics of each single standpoint to help identify the corresponding standpoint. Specifically, we employ standpoint-related topic-term distributions to enhance a generative standpoint classifier and feature representation of input texts. Finally, we develop an adaptive method to determine parameter values in our model for different input texts. The main process of our approach is given in Fig. 1.

3.1. Topic mining

To facilitate multiple standpoint detection, we first apply topic modeling to mine the latent topics for each standpoint. Specifically, we choose the widely used Nonnegative Matrix Factorization (NMF) based topic model to acquire the topic-term distributions and document-topic distributions in texts. The NMF based topic model factorizes the original term-document matrix V into two non-negative matrixes W and H so that $V \approx WH$, where W denotes the term-topic matrix and H denotes the topic-document matrix. By normalizing each column in matrix W and H, we acquire the term distribution of each topic and the topic distribution of each document respectively. The mined topic-term distributions are then used to help identify different standpoints in web texts.

3.2. Topic enhanced methods for multiple standpoint detection

Using the latent topic information of each standpoint, we develop a topic enhanced method for multiple standpoint detection. Specifically, this method merges the topic-term distributions of multiple standpoints into the standpoint classification model, and determines the optimal topics and merging weights for different input texts via an adaptive process.

To identify the standpoint of a text d among multiple standpoints, our standpoint classification model $f(\boldsymbol{x}; \boldsymbol{\theta})$ calculates the probabilities of text d belonging to different standpoints and chooses the one with maximum probability as text d's standpoint, i.e. $f(\boldsymbol{x}; \boldsymbol{\theta}) = \mathrm{argmax}_y P(y|\boldsymbol{x}; \boldsymbol{\theta})$, where $\boldsymbol{x}$ is the feature representation of text d.

To incorporate topic information, we merge topic-term distributions into the classification model $f(\boldsymbol{x}; \boldsymbol{\theta})$ and calculate the probability $P_{k_y, \lambda}(y|\boldsymbol{x}; \boldsymbol{\theta})$ that a text $\boldsymbol{x}$ belonging to standpoint y, where T_{k_y} denotes the k-th topic of standpoint y, i.e. the merging topic, and λ is the merging weight. Specifically, we first propose a Topic Enhanced Generative Model (i.e. TEGM) which merges topic-term distributions with standpoint-specific term distributions for classification. As generative models often achieve relatively poor performance when the real data distribution does not satisfy the specific assumption of models, we further propose a more general solution which refines the feature representation of texts with topic-term distributions (i.e. Topic Enhance Feature Representation, abbreviated as TEFR).

3.2.1. Topic enhanced generative model (TEGM)

We first incorporate topic information into a generative model, in which we use topic-term distributions to adjust model parameters, i.e. $P_{k_y, \lambda}(y|\boldsymbol{x}; \boldsymbol{\theta}) = P(y|\boldsymbol{x}; \boldsymbol{\theta}_{k_y, \lambda})$. This model classifies standpoints based on term features, and adopts the assumption of Naïve Bayes classifier that different terms in a text are independent of each other. Thus the probability $P(y|\boldsymbol{x}; \boldsymbol{\theta}_{k_y, \lambda})$ that a text is generated from standpoint y can be formulated as Eq. (1), where $\boldsymbol{x}$ is the feature representation of the text, $P(y)$ is the prior probability of standpoint y, V is the size of the feature set, $\boldsymbol{\theta}_{k_y, \lambda}$ is the model parameter which indicates the term distributions of multiple standpoints, and $\theta_{k_y, \lambda, i}$ denotes the i th element in $\boldsymbol{\theta}_{k_y, \lambda}$.

$$P(y|\boldsymbol{x}; \boldsymbol{\theta}_{k_y, \lambda}) \propto P(y) \cdot P(\boldsymbol{x}|y; \boldsymbol{\theta}_{k_y, \lambda}) = P(y) \cdot \prod_{i=1}^{V} \theta_{k_y, \lambda, i}^{x_i} \tag{1}$$

In order to calculate the probability $P(y|\boldsymbol{x}; \boldsymbol{\theta}_{k_y, \lambda})$ by Eq. (1), we need to estimate the prior probability (i.e. $P(y)$) and term distributions (i.e. $\boldsymbol{\theta}_{k_y, \lambda}$) of standpoint y based on the training corpus. Firstly, we follow Naïve Bayes classifier that estimates $P(y)$ as the portion of the texts belonging to standpoint y in the training corpus. Secondly, we generate a basic term distribution of each standpoint and then merge it with multiple topic-term distributions of this standpoint.

To acquire the basic term distribution $\boldsymbol{\psi}_{\boldsymbol{y}}$ of standpoint y, we use the term information of the texts belonging to this standpoint to draw a term distribution from the Dirichlet distribution, as given in Eq. (2), where $\boldsymbol{x}_i$ denotes the term feature representation of text d_i, y_i denotes the standpoint label of text d_i and $\boldsymbol{\alpha}_{\boldsymbol{\psi}}$ is a hyper-parameter which provides the prior information about term distribution. We draw the term distribution from a Dirichlet distribution as it provides the flexibility to incorporate the prior knowledge about term distributions.

$$\boldsymbol{\psi}_{\boldsymbol{y}} \sim Dirichlet\left(\boldsymbol{\alpha}_{\boldsymbol{\psi}} + \sum_{y_i = y} \boldsymbol{x}_i\right) \tag{2}$$

We then refine the basic term distribution of each standpoint with the corresponding topic-term distributions, as given in Eq. (3), where $\theta_{k_y, \lambda, i}$ denotes the i th element of $\boldsymbol{\theta}_{k_y, \lambda}$ (i.e. the refined term distribution for the k-th topic of standpoint y with merging weight λ), $\psi_{y, i}$ denotes the i th element of $\boldsymbol{\psi}_{\boldsymbol{y}}$, and $\varphi_{k_y, i}$ denotes the i th element of $\boldsymbol{\varphi}_{k_y}$ (i.e. the term distribution for the k-th topic of standpoint y which is acquired by the NMF based topic modeling).

$$\theta_{k_y, \lambda, i} = \frac{\psi_{y, i} + \lambda \cdot \varphi_{k_y, i}}{\sum_j \left(\psi_{y, j} + \lambda \cdot \varphi_{k_y, i}\right)} \tag{3}$$

By this means, we generate multiple topic-refined term distributions for a standpoint and each of the distributions corresponds to a specific topic of this standpoint. According to Eq. (1), we finally calculate the probabilities $P(y|\boldsymbol{x}; \boldsymbol{\theta}_{k_y, \lambda})$ to classify the standpoint of an input text via the adaptive process, which we will introduce in Section 3.3. The standpoint detection algorithm based on our topic enhanced generative model is given in Algorithm 1.

3.2.2. Topic enhanced feature representation method (TEFR)

The first method TEGM focuses on refining a generative standpoint classification model with latent topic information. However, generative models assume that data are generated from a certain probabilistic distribution. When the real data does not satisfy the pre-defined assumption, the performances of generative models usually are not very good. To facilitate a more general solution of standpoint classification, we then propose a topic enhanced method which refines the feature representation of texts for probability calculation, i.e. $P_{k_y, \lambda}(y|\boldsymbol{x}; \boldsymbol{\theta}) = P(y|\boldsymbol{x}_{k_y, \lambda}; \boldsymbol{\theta})$, where $\boldsymbol{x}_{k_y, \lambda}$ denotes the topic-based refined feature representation of texts. The refined feature representation can be used together with most existing classification algorithms which are able to output the probabilities of input texts belonging to different standpoints.

As the topics mined from different standpoints can provide distinguishable information for standpoint classification, the standpoint related information of a text would be more salient and more discriminative when the feature representation of this text is enhanced by its topic information. Therefore, we merge the term feature representation of a new text (i.e. a text in the testing corpus) with the topic-term distributions of multiple standpoints, as given in Eq. (4), where $\boldsymbol{x}_0$ denotes the original feature representation of texts and $\boldsymbol{\varphi}_{k_y}$ denotes the term distribution for the k-th topic of standpoint y.

$$\boldsymbol{x}_{k_y, \lambda} = \boldsymbol{x}_0 + \lambda \cdot \boldsymbol{\varphi}_{k_y} \tag{4}$$

Algorithm 1 TEGM based multiple standpoint detection.

Input: Training corpus $D = \{(d_i, y_i)|y_i \in Y, \ i = 1, 2, \ldots, N\}$, testing text d'
Output: The standpoint y' of text d'
1. Represent training corpus in the feature space, i.e. $\{(\boldsymbol{x_i}, y_i)|y_i \in Y, \ i = 1, 2, \ldots, N\}$
2. **For** each standpoint $y \in Y$
3.   Draw the basic term distribution $\boldsymbol{\psi_y}$ according to Eq. (2)
4.   Acquire multiple topic-term distributions $\boldsymbol{\varphi_{k_y}}$ by NMF based topic modeling
5.   Acquire the topic enhanced term distributions $\boldsymbol{\theta_{k_y,\lambda}}$ with parameter k_y and λ according to Eq. (3)
6. **End For**
7. Represent testing text d' in the feature space, i.e. $\boldsymbol{x'}$
8. Use the adaptive process to determine the optimal topic $\widetilde{k_y}$ of each standpoint y and the optimal merging weight $\tilde{\lambda}$ for text $\boldsymbol{x'}$
9. **For** each standpoint $y \in Y$
10.   Calculate probability $P(y|\boldsymbol{x'}; \boldsymbol{\theta_{k_y,\lambda}})$ with $\widetilde{k_y}$ and $\tilde{\lambda}$ according to Eq. (1)
11. **End For**
12. **Return** the identified standpoint $y' = \text{argmax}_y P(y|\boldsymbol{x'}; \boldsymbol{\theta_{k_y,\lambda}})$

Algorithm 2 TEFR based multiple standpoint detection.

Input: Training corpus $D = \{(d_i, y_i)|y_i \in Y, \ i = 1, 2, \ldots, N\}$, testing text d'
Output: The standpoint y' of text d'
1. Represent training corpus in the feature space, i.e. $\{(\boldsymbol{x_i}, y_i)|y_i \in Y, \ i = 1, 2, \ldots, N\}$
2. Construct a standpoint classifier based on the training corpus
3. Represent testing text d' in the feature space, i.e. $\boldsymbol{x'}$
4. **For** each standpoint $y \in Y$
5.   Acquire multiple topic-term distributions $\boldsymbol{\varphi_{k_y}}$ by NMF based topic modeling
6.   Acquire the topic enhanced feature representations $\boldsymbol{x'}_{k_y,\lambda}$ of $\boldsymbol{x'}$ with parameter k_y and λ according to Eq. (4)
7. **End For**
8. Use the adaptive process to determine the optimal topic $\widetilde{k_y}$ of each standpoint y and the optimal merging weight $\tilde{\lambda}$ for text $\boldsymbol{x'}$
9. **For** each standpoint $y \in Y$ **do**:
10.   Calculate probability $P(y|\boldsymbol{x'}_{k_y,\lambda}; \boldsymbol{\theta})$ with $\widetilde{k_y}$ and $\tilde{\lambda}$ according to Eq. (1)
11. **End For**
12. **Return**: the identified standpoint $y' = \text{argmax}_y P(y|\boldsymbol{x'}_{k_y,\lambda}; \boldsymbol{\theta})$

The calculation of the probability that the new text belongs to a specific standpoint with refined feature representation (i.e. $P(y|\boldsymbol{x_{k_y,\lambda}}; \boldsymbol{\theta})$) depends on the applied classification algorithm. Then we adopt the adaptive process to determine the optimal topics and merging weight of this new text for standpoint classification. The standpoint detection algorithm based on our topic enhanced feature representation is given in Algorithm 2.

3.3. Adaptive parameter determination

For an input text, to determine its topics in different standpoints as well as the merging weight, we develop an adaptive process which adopts the idea of Support Vector Machine that maximizes the margins between classes. Specifically, we determine the value of merging weight $\tilde{\lambda}$ and topics $\widetilde{k_y}$ via an object function which aims to maximize the gap between the largest and second largest probability, as indicated in Eqs. (5) and (6), where max ($\cdot$) and sndmax denote the largest and second largest values respectively, and $\max_{k_y} P_{k_y,\lambda}(y|\boldsymbol{x}; \boldsymbol{\theta})$ denotes the maximum probability of a text belonging to standpoint y when merging with different topics of y by a fixed weight λ.

$$\tilde{\lambda} = \underset{\lambda}{\text{argmax}} \left(\max_y \left(\max_{k_y} P_{k_y,\lambda}(y|\boldsymbol{x}; \boldsymbol{\theta}) \right) - \underset{y}{\text{sndmax}} \left(\max_{k_y} P_{k_y,\lambda}(y|\boldsymbol{x}; \boldsymbol{\theta}) \right) \right) \tag{5}$$

$$\widetilde{k_y} = \underset{k_y}{\text{argmax}} \, P_{k_y,\tilde{\lambda}}(y|\boldsymbol{x}; \boldsymbol{\theta}) \tag{6}$$

As this object function is non-convex, it is hard to find the global optimization solution. Thus we try to search for an approximate solution by traversal. To this end, we discretize λ and confine it in a predefined interval, i.e. $\lambda = i \cdot t, i = 1, 2, \ldots, L$, where L is the size of the interval and t is the step size in search. By this means, we first search for the optimal merging weight by Eq. (5), and then determine the optimal topics in different standpoints by Eq. (6). Besides, we set identical number of topics for all the standpoints for simplicity. We finally classify the standpoint of the input text based on the optimal topics $\widetilde{k_y}$ and merging weight $\tilde{\lambda}$, i.e. $f(\boldsymbol{x}; \boldsymbol{\theta}) = \text{argmax}_y P(y|\boldsymbol{x}; \boldsymbol{\theta}) = \text{argmax}_y P_{\widetilde{k_y},\tilde{\lambda}}(y|\boldsymbol{x}; \boldsymbol{\theta})$.

4. Experiments

4.1. Datasets

We use three datasets for our experimental study. The first dataset contains Web documents collected from the "bitter-lemons.org" website (abbreviated as the "Bittterlemons" dataset), which is also used in several related work [21,38,39]. This website constantly published four documents every week about various topics of the Palestine-Israel conflict from 2001 to 2012. Among these four documents, two of them were written by two fixed editors and the other two were written by invited guest editors. In addition to the content and author information, this website provides the standpoint of each document, either Palestine or Israel. We crawl all the 1765 documents written by 335 authors together with their standpoints. For the separation of training and testing data, we adopt the strategy of Lin et al. [21] whose experiments revealed that the most challenging use of this dataset is to train on the documents written by the editors and test on the documents written by the guest editors. In this dataset, the numbers of documents belonging to Palestine and Israel standpoints in the training corpus are both 446, and the numbers of documents belonging to Palestine and Israel standpoints in the testing corpus are 436 and 437 respectively.

Compared to the first dataset which mainly consists of long documents with the average length of 374, the second and third datasets are crawled from Twitter, whose tweets are relatively short with the average length of 26. These two Twitter datasets are about the 2016 American presidential election and the four presidential candidates, namely Hillary Clinton, Bernie Sanders, Donald Trump and Ted Cruz, are regarded as targets. The second dataset (abbreviated as the "Twitter dataset 1") is constructed by ourselves while the third dataset (abbreviated as the "Twitter dataset 2") is acquired from Sobhani et al. [30], which is relatively larger. For "Twitter dataset 1", we use the name of four candidates as the keywords in Twitter streaming API to crawl relevant tweets. The crawled tweets are published by 7907 users from March 30th to June 6th in 2016. To construct the labeled dataset, we randomly select 4000 tweets and invite two senior students majored in Artificial Intelligence and Social Media Analysis to annotate the standpoints of them. To ensure high-quality annotated data, the raters were instructed to only annotate the tweets which show clear evidence about the standpoints. The averaged kappa coefficient of the two raters is 95%. Among the data annotated by each rater, we randomly select 66% of the standpoint-related tweets for training and use the rest of the tweets for testing. According to the annotation of rater 1, the total number of training and testing tweets belonging to Hillary Clinton, Bernie Sanders, Donald Trump and Ted Cruz standpoints is 705 and 364, respectively. According to the annotation of rater 2, the numbers of training and testing tweets belonging to Hillary Clinton, Bernie Sanders, Donald Trump and Ted Cruz standpoints are 869 and 448, respectively. For the preprocessing of the datasets, we lemmatize the words and remove URLs and non-English words. For "Twitter dataset 2", we select those tweets which express the "favor" stance for any one of the targets from the original dataset in Sobhani et al. [30] and then construct the dataset of detecting standpoints regarding the above four candidates. As a result, we have 1957 and 490 tweets which support Hillary Clinton, Bernie Sanders, Donald Trump or Ted Cruz for training and testing, respectively.

4.2. Experimental setup

For the parameter setting of our topic enhanced approach, to balance the efficiency and accuracy in searching for the optimal merging weight $\tilde{\lambda}$, we set the size of the search interval (i.e. L) to 5, and set the step size (i.e. t) in search to 0.1 and 2.0 for TEGM and TEFR respectively. We determine the topic number of each standpoint by cross-validation on the training corpus. Specifically, for TEGM, we set the topic number of each standpoint in the "Bitterlemons" dataset and the Twitter datasets to 5 and 10 respectively. For TEFR, we set the topic number in the "Bitterlemons" dataset and the "Twitter dataset 1" to 4, and set the topic number in the "Twitter dataset 2" to 2. Besides, in TEGM, as we have no prior information about the term distributions of multiple standpoints, currently we set each element in the hyper-parameter α_ψ to 1 in all datasets, which is a non-informative prior. In TEFR, we choose Logistic Regression as the classification algorithm because of its good performance and high efficiency. For both TEGM and TEFR, we take word unigrams as features in the "Bitterlemons" dataset. In the two Twitter datasets, to make use of more linguistic information of short tweets, we follow the related work [23] and take 1~3 word n-grams and 2~5 character n-grams as features.

4.3. Comparative methods

To test the performance of our topic enhanced approach to multiple standpoint detection, we first compare it with three notable state-of-the-art methods of traditional stance detection, namely Support Machine Vector[+] (denoted as SVM[+]) [23], pkudblab [44] and Bidirectional conditional LSTM (denoted as BiConditional) [3]. SVM[+] and pkudblab are the methods which achieve the *best* performances in the stance detection task A and B of SemEval 2016. SVM[+] takes word and character n-grams as features and constructs an SVM classifier to detect different stances, while pkudblab trains a convolutional neural network for stance classification. BiConditional is based on deep bi-directional recurrent neural network, which is a widely used strong comparative method in stance detection. We further take the closely related work on supervised standpoint detection (i.e. Latent Sentence Perspective Model, denoted as LSPM [21]) for comparison.

In addition, we also include several recently proposed work on stance detection, such as AS-BiGRU-CNN [46], Two-Phase Attention Neural network (denoted as T-PAN) [9] and Hierarchical Attention Network (denoted as HAN) [36]. The AS-BiGRU-

Table 1

Performances of our approach and comparative methods.

| Methods | Bitterlemons dataset | | Twitter dataset 1 | | | | Twitter dataset 2 | |
| | | | Rater 1 | | Rater 2 | | | |
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
HAN [36]	66.46%	66.43%	76.93%	76.98%	79.19%	78.57%	50.09%	48.37%
T-PAN [9]	68.11%	66.83%	77.34%	75.96%	79.48%	76.81%	61.02%	59.31%
BiConditional [3]	84.31%	84.30%	75.55%	74.13%	81.56%	79.40%	57.88%	55.65%
pkudblab [44]	82.04%	82.00%	**84.95%**	82.53%	86.83%	85.19%	69.12%	44.07%
AS-BiGRU-CNN [46]	77.21%	76.99%	81.87%	79.62%	85.71%	84.33%	72.48%	**67.50%**
LSPM [21]	84.32%	84.28%	78.41%	76.15%	81.54%	79.67%	65.13%	44.00%
SVM$^+$ [23]	84.19%	84.19%	83.24%	80.93%	86.38%	84.80%	71.01%	58.98%
TEGM	84.81%	84.80%	82.94%	80.69%	86.70%	84.88%	65.06%	45.15%
TEFR	**85.34%**	**85.31%**	84.62%	**82.83%**	**87.50%**	**86.49%**	**74.37%**	62.64%

CNN model adopts a BiGRU based method and attention mechanism between stance targets and tweets, followed by a CNN structure. Different from AS-BiGRU-CNN, the T-PAN model adopts a two-phase framework and uses LSTM to extract semantic features. The HAN model not only considers the target and tweets as inputs, but also includes sentiments, semantic dependency relations and argument aspects. It further applies the hierarchical attention model to weight different linguistic information.

For our proposed approach, we compare the performances of TEFM and TEFR in different datasets.

4.4. Experimental results and analysis

To evaluate our proposed approach, we first compare it with the related methods introduced in Section 4.3. We then examine whether the topic enhanced topic representation (i.e. TEFR) can work well with different classification algorithms. Finally, we investigate the influence of different topic numbers on the performance of our approach. We use both Accuracy and macro-averaged F1-value as the evaluation measures.

4.4.1. Comparison with related methods

The accuracies and F1-values of our approach and the comparative methods are given in Table 1.

From Table 1, we can see that traditional machine learning methods, such as SVM$^+$ and LSPM, generally perform better than deep learning based methods, such as HAN and T-PAN, for multiple standpoint detection task. Those methods that perform better on long texts (i.e. Bitterlemons dataset) do not necessarily perform well on short texts (i.e. Twitter datasets), and vice versa. For instance, BiConditional and LSPM perform better on long texts, whereas AS-BiGRU-CNN, HAN and T-PAN perform better on short texts. Almost all these methods perform relatively worse on "Twitter dataset 2", mainly due to the fact that each tweet in this dataset contains more expressions of "favor" or "against" towards multiple targets. Our proposed methods, especially TEFR, generally perform better and more stable on different datasets compared with all the other methods.

For the two non-deep learning methods, SVM$^+$ and LSPM achieve stable and relatively good performances in both "Bitterlemons" and Twitter datasets. This indicates that lexical features play an important role in distinguishing multiple standpoints. LSPM yields the highest accuracy among the comparative methods in the "Bitterlemons" dataset, but it does not perform well in the Twitter datasets. The reason could be that LSPM merely uses automatically identified standpoint-related sentences in a document for standpoint classification. Thus it could miss some important information and be unsuitable for short and informal tweets.

For deep learning based methods, the RNN based models, including HAN, T-PAN and BiConditional, perform worse than the CNN based models, such as pkudblab. By combining RNN (i.e. BiGRU) and CNN, AS-BiGRU-CNN achieves the best performance among all the deep learning based methods. The main reason why HAN performs worse than other two RNN-based methods could be that, HAN needs sentiment, dependency and argument information as inputs, which usually could not be accurately obtained automatically.

For our topic enhanced approach, TEGM outperforms all the comparative methods in the "Bitterlemons" dataset. Specifically, TEGM improves the accuracy and F1-value by about 0.5%, indicating that topic-related lexical information contributes to identifying multiple standpoints. TEFR outperforms all the comparative methods and TEGM in most cases. This implies that the lexical features enhanced with topic information are more effective in classifying different standpoints. In summary, the experimental results verify the effectiveness of our topic enhanced approach to multiple standpoint detection.

4.4.2. Comparison of TEFR with different classification algorithms

To examine whether the topic enhanced feature representation method TEFR works well with different classifiers, we choose four typical classification algorithms for investigation, i.e. Naïve Bayes (NB), Decision Tree (DT), Support Vector Machine (SVM) and Logistic Regression (LR). Fig. 2 gives the F1-values of these classification algorithms using pure word
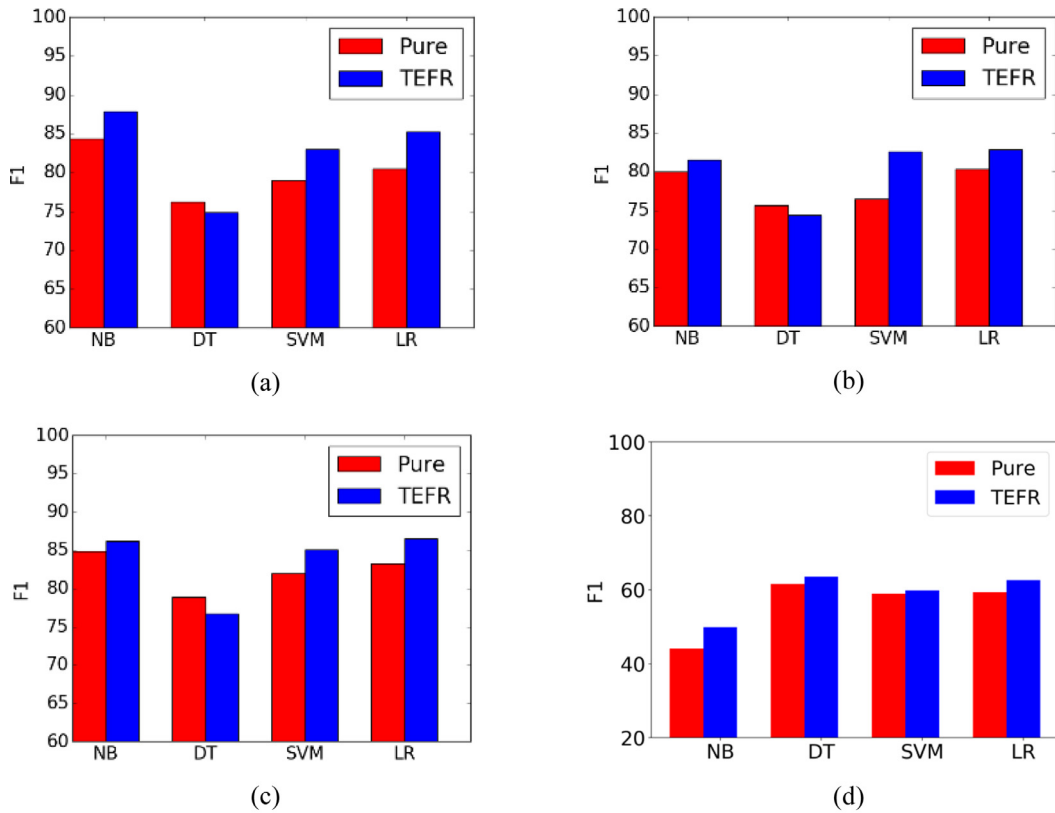
Fig. 2. F1-values of different classification algorithms using pure and topic enhanced features in "Bitterlemons" dataset (a), Twitter dataset 1 annotated by rater 1 (b) and rater 2 (c), and Twitter dataset 2 (d).

features (Pure) and those using topic enhanced features (TEFR) in the "Bitterlemons" dataset, the "Twitter dataset 1" and the "Twitter dataset 2" respectively.

We can see from Fig. 2 that over these classification algorithms, NB achieves the best performance in the "Bitterlemons" dataset, LR achieves the best performance in the "Twitter dataset 1", and DT achieves the best performance in the "Twitter dataset 2". NB, SVM and LR using topic enhanced feature representation perform better than the corresponding classification algorithms using pure word features. This indicates that TEFR can be applied to both generative and discriminative models and improve their performances. It can also be seen that DT using topic enhanced feature representation often performs worse than DT using pure word features. One reason for this is that Decision Tree classifier iteratively selects one most distinguishable feature to construct a tree structure for classification, thus the selected features greatly affect its performance. The topic-based feature enhancement could make some useful features less distinguishable, which leads to performance degradation.

4.4.3. Influence of topic number

We finally investigate how topic number influences the performance of our approach, including TEGM and TEFR. The F1-values of TEGM and TEFR with different topic numbers in all the datasets are given in Fig. 3.

We can see from Fig. 3 that in the "Bitterlemons" dataset, TEGM achieves the highest F1-value when the topic number is 9, and TEFR achieves the highest F1-value when the topic number is larger than 10. In the "Twitter dataset 1" labeled by rater 1, TEGM achieves the highest F1-value when the topic number is 7, and TEFR achieves the highest F1-value when the topic number is 3 or 4. In the "Twitter dataset 1" labeled by rater 2, TEGM achieves the highest F1-value when the topic number is 8, and TEFR achieves the highest F1-value when the topic number is larger than or equal to 4. In the "Twitter dataset 2", both TEGM and TEFR achieve the highest F1-value when the topic number is 2. Empirically, larger topic number can facilitate more fine-grained modeling of the latent topics in texts, which is beneficial for our topic enhanced methods. However, if the topic number is too large, our methods could suffer from data sparsity problem. Therefore, in many cases, when topic number increases, the F1-values of TEGM and TEFR become higher in the beginning and then drop slowly. Besides, TEGM is more sensitive to pre-defined topic number than TEFR. One possible reason for this is that the performance of the topic enhanced standpoint classifier (i.e. TEGM) can be easily influenced by topic quality, which is closely related to the predefined topic number in topic modeling. In contrast, TEFR only use topic information to enhance feature
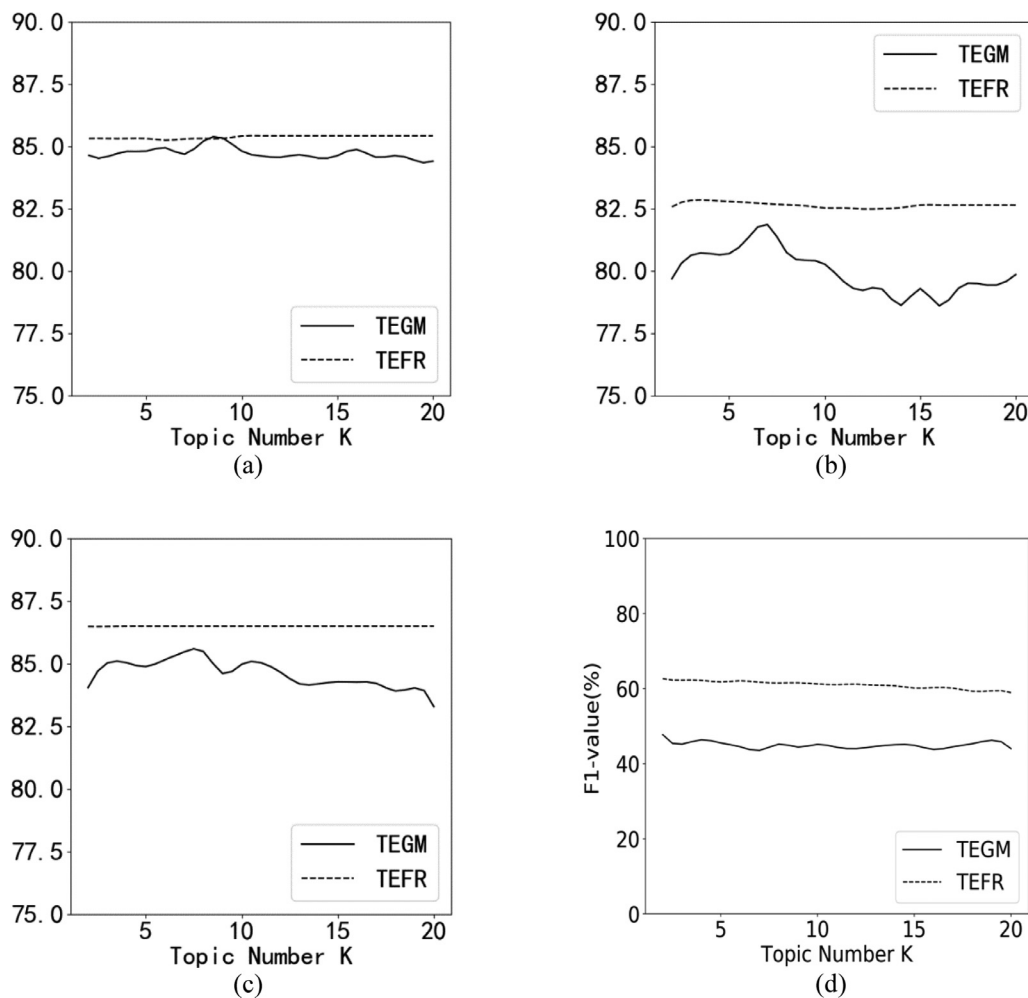
Fig. 3. F1-values of TEGM and TEFR with different topic numbers in "Bitterlemons" dataset (a), Twitter dataset 1 annotated by rater 1 (b) and rater 2 (c), and Twitter dataset 2 (d).

representation of a specific input text, which is more lightweight and robust to topic quality. Overall, the performances of our methods are relatively stable when topic number varies.

5. Conclusion and future work

People tend to publish texts on the Web to express their stances. It is both beneficial and critical for many applications to make sense of the stances implied in Web texts. Most previous work on stance detection aim at identifying the supportive or unsupportive attitudes towards one certain topic or entity, without considering the real standpoints people hold among multiple parties. Although some recent work have made some efforts on standpoint detection, they only focus on the classification of two-sided standpoints. Multiple standpoint detection, which aims to determine the standpoints of people among multiple parties, has been largely ignored. In this paper, we focus on solving the problem of multiple standpoint detection.

As people with different standpoints tend to discuss distinct topics, the latent topics in texts can help identify certain standpoints in a more fine-grained way. Motivated by this, we propose a topic enhanced approach which leverages the latent topic information in texts for multiple standpoint detection. In this approach, we make use of topic-term distributions to enhance a generative classification model (i.e. TEGM) and the feature representation of texts (i.e. TEFR). TEGM refines Naïve Bayes model by supplying lexical information of the original texts with topic-related term information. Compared with TEGM, TEFR is more flexible, which leverages topic information to adjust the importance of different lexical features, and can be applied to most classifiers. We also design an adaptive process to determine the values of model parameters automatically for different input texts. This process adopts the idea of SVM and determines the topic and the merging weight of a specific text by maximizing the margin between different standpoints. Experimental studies using three real datasets show the effectiveness of our proposed topic enhanced approach to multiple standpoint detection.

In the future, we shall extend our work in several directions. First, we shall explore the ways to automatically determine the optimal topic number in our approach. To this end, we shall try non-parametric topic models (e.g. Hierarchical Dirichlet Process [37] and Doubly nonparametric sparse nonnegative matrix factorization [45]) to mine the latent topics in texts. In addition, as currently we use static topic model to mine topic information, we shall further take into account the temporal information and apply dynamic topic models to enhance standpoint classifiers. The third direction is to consider the automatic summarization of standpoint keywords and the extraction of standpoint-related contents from massive Web texts.

Declaration of Competing Interest

We wish to confirm that there are no known conflicts of interest associated with this publication.

Acknowledgments

References

[1] R. Abbott, M. Walker, P. Anand, J.E. Fox Tree, R. Bowmani, J. King, How can you say such things?!?: Recognizing disagreement in informal political argument, in: Proceedings of the Workshop on Languages in Social Media, Stroudsburg, PA, USA, Association for Computational Linguistics, 2011, pp. 2–11.

[2] P. Anand, M. Walker, R. Abbott, J.E.F. Tree, R. Bowmani, M. Minor, Cats rule and dogs drool!: classifying stance in online debate, in: Proceedings of the 2Nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, Stroudsburg, PA, USA, Association for Computational Linguistics, 2011, pp. 1–9.

[3] I. Augenstein, T. Rocktäschel, A. Vlachos, K. Bontcheva, Stance detection with bidirectional conditional encoding, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 876–885.

[4] D.M. Blei, Probabilistic topic models, Commun ACM 55 (2012) 77–84.

[5] F. Boltužić, J. Šnajder, Back up your stance: recognizing arguments in online discussions, in: Proceedings of the First Workshop on Argumentation Mining, 2014, pp. 49–58.

[6] A. Bondielli, F. Marcelloni, A survey on fake news and rumour detection techniques, Inform. Sci. 497 (2019) 38–55.

[7] W.-F. Chen, L.-W. Ku, UTCNN: a deep learning model of stance classification on social media text, in: Proceedings of the 26th International Conference on Computational Linguistics, 2016, pp. 1635–1645.

[8] E. D'Andrea, P. Ducange, A. Bechini, A. Renda, F. Marcelloni, Monitoring the public opinion about the vaccination topic from tweets analysis, Expert Syst. Appl. 116 (2019) 209–226.

[9] K. Dey, R. Shrivastava, S. Kaushik, Topical stance detection for twitter: a two-phase LSTM model using attention, in: G. Pasi, B. Piwowarski, L. Azzopardi, A. Hanbury (Eds.), Advances in Information Retrieval, Springer International Publishing, 2018, pp. 529–536.

[10] J. Ebrahimi, D. Dou, D. Lowd, Weakly supervised tweet stance classification by relational bootstrapping, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 1012–1017.

[11] J. Ebrahimi, D. Dou, D. Lowd, A joint sentiment-target-stance model for stance classification in tweets, in: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics, 2016, pp. 2656–2665.

[12] M. Hagen, M. Potthast, M. Büchner, B. Stein, Webis: an ensemble for twitter sentiment detection, in: Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015), 2015, pp. 582–589.

[13] H.L. Hammer, P.E. Solberg, L. Øvrelid, Sentiment classification of online political discussions: a comparison of a word-based and dependency-based method, in: Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2014, pp. 90–96.

[14] K.S. Hasan, V. Ng, Stance classification of ideological debates: data, models, features, and constraints, in: Proceedings of the Sixth International Joint Conference on Natural Language Processing, 2013, pp. 1348–1356.

[15] K.S. Hasan, V. Ng, Extra-linguistic constraints on stance recognition in ideological debates, in: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2013, pp. 816–821.

[16] K.S. Hasan, V. Ng, Why are you taking this stance? identifying and classifying reasons in ideological debates, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 751–762.

[17] A. Hogenboom, F. Frasincar, F. de Jong, U. Kaymak, Using rhetorical structure in sentiment analysis, Commun ACM 58 (7) (2015) 69–77.

[18] D. Hyun, C. Park, M.-C. Yang, I. Song, J.-T. Lee, H. Yu, Target-aware convolutional neural network for target-level sentiment analysis, Inform. Sci. 491 (2019) 166–178.

[19] K. Johnson, D. Goldwasser, All I know about politics is what I read in Twitter": weakly supervised models for extracting politicians' Stances From Twitter, in: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, 2016, pp. 2966–2977.

[20] D. Kuang, J. Choo, H. Park, Nonnegative matrix factorization for interactive topic modeling and document clustering, in: M.E. Celebi (Ed.), Partitional Clustering Algorithms, Springer International Publishing, Cham, 2015, pp. 215–243.

[21] W.-H. Lin, T. Wilson, J. Wiebe, A. Hauptmann, Which side are you on?: identifying perspectives at the document and sentence levels, in: Proceedings of the Tenth Conference on Computational Natural Language Learning, Stroudsburg, PA, USA, Association for Computational Linguistics, 2006, pp. 109–116.

[22] M. Lukasik, P.K. Srijith, D. Vu, K. Bontcheva, A. Zubiaga, T. Cohn, Hawkes processes for continuous time sequence classification: an application to rumour stance classification in twitter, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2016, pp. 393–398.

[23] S.M. Mohammad, P. Sobhani, S. Kiritchenko, Stance and sentiment in tweets, ACM Trans. Internet Technol. 17 (26) (2017) 1–26 23.

[24] B. Pang, L. Lee, S. 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, Stroudsburg, PA, USA, Association for Computational Linguistics, 2002, pp. 79–86.

[25] M. Qiu, J. Jiang, A latent variable model for viewpoint discovery from threaded forum posts, in: Proceedings of NAACL-HLT, 2013, pp. 1031–1040.

[26] Y. Rao, Q. Li, X. Mao, L. Wenyin, Sentiment topic models for social emotion mining, Inform. Sci. 266 (2014) 90–100.

[27] Y. Ren, R. Wang, D. Ji, A topic-enhanced word embedding for Twitter sentiment classification, Inform. Sci. 369 (2016) 188–198.

[28] M. Ringsquandl, D. Petkovic, Analyzing political sentiment on twitter,, 2013 AAAI Spring Symposium Series, 2013.

[29] J. Serrano-Guerrero, J.A. Olivas, F.P. Romero, E. Herrera-Viedma, Sentiment analysis: a review and comparative analysis of web services, Inform. Sci. 311 (2015) 18–38.

[30] P. Sobhani, D. Inkpen, X. Zhu, A Dataset for Multi-Target Stance Detection, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, Valencia, Spain, Association for Computational Linguistics, 2017, pp. 551–557.

[31] P. Sobhani, D. Inkpen, X. Zhu, Exploring deep neural networks for multitarget stance detection, Comput. Intell. 35 (1) (2019) 82–97.

[32] S. Somasundaran, J. Wiebe, Recognizing stances in ideological on-line debates, in: Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, Stroudsburg, PA, USA, Association for Computational Linguistics, 2010, pp. 116–124.

[33] D. Sridhar, J. Foulds, B. Huang, L. Getoor, M. Walker, Joint models of disagreement and stance in online debate, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, 2015, pp. 116–125.

[34] D. Sridhar, L. Getoor, M. Walker, Collective stance classification of posts in online debate forums, in: Proceedings of the Joint Workshop on Social Dynamics and Personal Attributes in Social Media, 2014, pp. 109–117.

[35] Q. Sun, Z. Wang, S. Li, Q. Zhu, G. Zhou, Stance detection via sentiment information and neural network model, Front. Comput. Sci. 13 (1) (2019) 127–138.

[36] Q. Sun, Z. Wang, Q. Zhu, G. Zhou, Stance detection with hierarchical attention network, in: Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 2399–2409.

[37] Y.W. Teh, M.I. Jordan, M.J. Beal, D.M. Blei, Hierarchical Dirichlet processes, J. Am. Statist. Assoc. 101 (2006) 1566–1581.

[38] T. Thonet, G. Cabanac, M. Boughanem, K. Pinel-Sauvagnat, VODUM: a topic model unifying viewpoint, topic and opinion discovery, in: N. Ferro, F. Crestani, M.-F. Moens, J. Mothe, F. Silvestri, G.M. Di Nunzio, C. Hauff, G. Silvello (Eds.), Advances in Information Retrieval, Springer International Publishing, 2016, pp. 533–545.

[39] A. Trabelsi, O.R. Zaïane, A joint topic viewpoint model for contention analysis, in: E. Métais, M. Roche, M. Teisseire (Eds.), Natural Language Processing and Information Systems, Springer International Publishing, 2014, pp. 114–125.

[40] P.D. Turney, Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews, in: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Stroudsburg, PA, USA, Association for Computational Linguistics, 2002, pp. 417–424.

[41] S. Vosoughi, M. 'Neo' Mohsenvand, D. Roy, Rumor Gauge, Predicting the veracity of rumors on twitter, ACM Trans. Knowl. Disc. Data 11 (50) (2017) 1–50 36.

[42] M.A. Walker, P. Anand, R. Abbott, R. Grant, Stance classification using dialogic properties of persuasion, in: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Stroudsburg, PA, USA, Association for Computational Linguistics, 2012, pp. 592–596.

[43] L. Wang, C. Cardie, Improving agreement and disagreement identification in online discussions with a socially-tuned sentiment lexicon, in: Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2014, pp. 97–106.

[44] W. Wei, X. Zhang, X. Liu, W. Chen, T. Wang, pkudblab at semeval-2016 task 6: a specific convolutional neural network system for effective stance detection, in: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), 2016, pp. 384–388.

[45] J. Xuan, J. Lu, G. Zhang, R.Y.D. Xu, X. Luo, Doubly nonparametric sparse nonnegative matrix factorization based on dependent Indian buffet processes, IEEE Transactions on Neural Networks and Learning Systems 29 (2018) 1835–1849.

[46] Y. Zhou, A.I. Cristea, L. Shi, Connecting targets to tweets: semantic attention-based model for target-specific stance detection, in: A. Bouguettaya, Y. Gao, A. Klimenko, L. Chen, X. Zhang, F. Dzerzhinskiy (Eds.), Web Information Systems Engineering – WISE 2017, Springer International Publishing, 2017, pp. 18–32.

[47] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, R. Procter, Detection and resolution of rumours in social media: a survey, ACM Comput. Surv. 51 (32) (2018) 1–32 36.

[48] A. Zubiaga, E. Kochkina, M. Liakata, R. Procter, M. Lukasik, Stance classification in rumours as a sequential task exploiting the tree structure of social media conversations, in: Proceedings of the 26th International Conference on Computational Linguistics, 2016, pp. 2438–2448.