International Journal of Contemporary Research in Computer Science and Technology (IJCRCST) e-ISSN: 2395-5325
Volume2, Issue 3 (March'2016)

SECURED SOCIAL MEDIA USING ASPECT BASED SENTIMENT ANALYSIS

Banu.M
B.E scholar,
Computer Science and Engineering,
Velammal Institute of Technology,
Chennai,India.

Muthra.S
B.E scholar,
Computer Science and Engineering,
Velammal Institute of Technology,
Chennai,India.

Hema.S
B.E scholar,
Computer Science and Engineering,
Velammal Institute of Technology,
Chennai,India.

Nalini.S
Assistant Professor,
Computer Science and Engineering,
Velammal Institute of Technology,
Chennai,India.

Abstract: Sentiment analysis, which addresses the computational treatment of opinion, sentiment, and subjectivity in text, has received considerable attention in recent years. In contrast to the traditional coarse-grained sentiment analysis tasks, such as document-level sentiment classification, we are interested in the fine-grained aspect-based sentiment analysis that aims to identify aspects that users comment on and these aspects' polarities. Aspect-based sentiment analysis relies heavily on syntactic features. However, the reviews that this task focuses on are natural and spontaneous, thus posing a challenge to syntactic parsers. In this paper, a set of words are defined in a dictionary. In any chatting application or in social media, the user will have many followers and there will be lots of messages or tweets. If the member of the group uses the offense word will be blocked by the page administrator, thus providing a comfort level to the other users from such harmful users. This will help to maintain a social networks for its dedicated purpose.

I.INTRODUCTION

The internet holds a considerable amount of user generated content describing the opinions of customers on products and services through blogs, tweets and other social media forms. These reviews are valuable for customers making purchasing decisions and companies guiding business activities. However, browsing the extensive collection of reviews to search for useful information is a time-consuming and tedious task. Consequently, sentiment analysis and opinion mining have attracted significant attention in recent years as they pave the way for the automatic analysis of user reviews and the extraction of information most relevant to users. Sentiment classification isa fundamental and most studiedarea in sentiment analysis. Its purpose is to determine the sentiment polarity (positive or negative) of a sentence (or a document) based on its texturalcontent.Sentiment analysis entails several interesting and challenging tasks. One traditional and fundamental task is polarity classification, which determines the overall polarity (e.g., positive or negative) of a sentence or document [1], [2], [3]. However, thesetasks are coarse-grained and cannot provide detailed information, such as the aspects on which the users comment. Recently, there has been ashift towards the fine-grained tasks, such as aspect-based (or "feature-based") sentiment analysis, which not only involves analysing the opinionated text's polarity (e.g.,positive, neutral, negative) and intensity (e.g., weak, medium,strong, extreme), but also identifying the aspect (or the topic, o target entity) of the opinion [4], [5], [6], [7].

II. RELATED WORK

A. Sentiment Analysis

Earlier research on sentiment analysis primarily focused on polarity classification, i.e. determining the sentiment orientation of a sentence or a document [1], [2], [3]. However, these tasks are all coarse-grained and cannot provide more detailed information Recently, there has been a shift towards fine-grained aspect-based tasks that can identify both the text expressing the opinion and the aspect of the opinion as well as analysing its polarity (e.g., positive, neutral or negative) [5], [6]. The A-P collocation extraction is the basic task of aspect-based sentiment analysis.To tackle this task, most methods focused on identifying relationships between the aspects and the polarity words. In earlier studies, researchers recognized the aspect first and then chose its polarity word within a window of size [7].

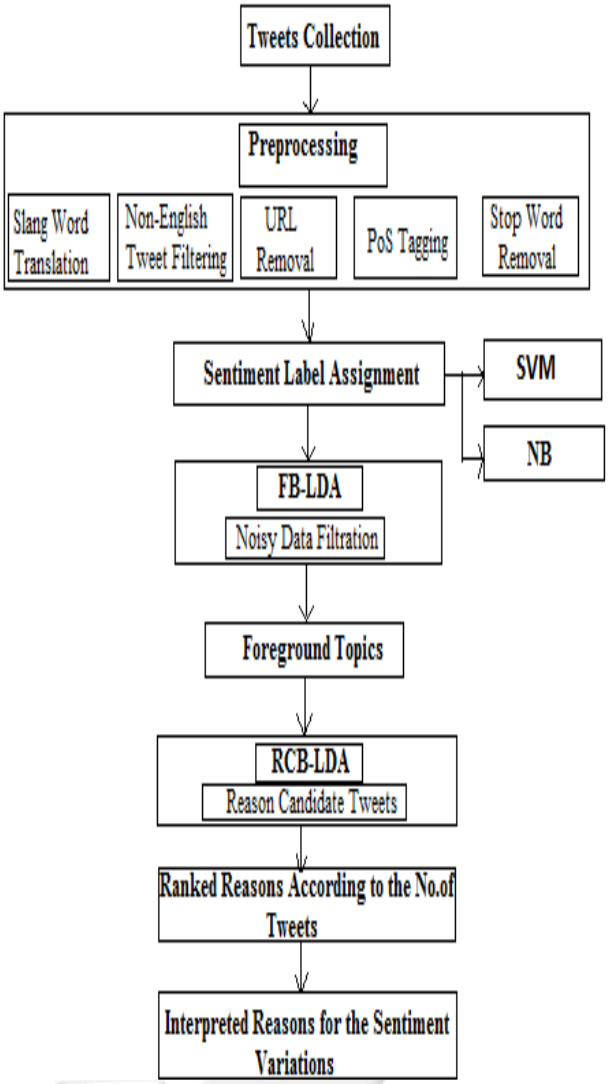
However, considering that this type of method is too heuristic, the performances proved to be extremely limited. To solve this problem, several researchers found that a syntactic pattern can better describe the relationship between the aspects and the polarity words.For example, Bloom et al. [11] constructed a linkage specification lexicon containing patterns. Qiu et al. [6] proposed a double propagation method that introduced eight heuristic syntactic patterns to extract the collocations. Xu et al. [12] used the syntactic patterns to extract the collocation candidates in their two-stage framework. Based on the above discussion, we can conclude that the syntactic features are extremely important in executing aspect based sentiment

analysis tasks. However, the "naturalness"problem can still seriously affect the performance of the syntactic parser. Once our sentiment sentence compression method can improve the quality of parsing, the performance of several aspect-based sentiment analysis tasks can be improved as well. It should be noted that to date, there is no previous study on using sentence compression models to improve aspect-based sentiment analysis.

B. Sentence Compression

Sentence compression is a paraphrasing task that aims at generating sentences that are shorter than the given ones, while pre-serving the essential content [13]. There are many applications that can benefit from a robust compression system. For example, we can use the system to reduce the redundancy in sentences and generate informative summarization systems [38]. Additionally, we can use it to compress the complicated sentiment sentences into easy-to-parse ones to get more accurate syntactic features, and further improve the tasks that primarily rely on the syntactic features, such as semantic role labelling [16], relation extraction[39], etc. Tree-based approaches were commonly used to compress sentences [13], [14], [15], which created a compressed sentence by making edits to the syntactic tree of the original sentence. However, the automatic parsingresults may not be correct ;thus, the compressed tree (after removing constituents from a bad parse) may not produce a suitable compressed sentence .McDonald et al. andNomoto et al. [19] attempted to solve this problem using discriminative models. They studied classifiers to determine which words could be dropped by including features of the words themselves as well as part of the speech tags and parser trees. Here, the parser trees were soft evidence to determine whether to remove a word.

Accordingly, the influencers of the parsing errors were reduced. Moreover, to improve the efficiency of the compression model, recent studies have been performed on polynomial time inference algorithms and approximate inference algorithms for sentence compression. Currently, the existing sentencecompression methods all focus on formal sentences, and few methods have been studied for sentiment sentences. As discussed in the above sections, the current compression models cannot be directly transplanted to sentiment sentences due to the specificity of the aspect-based sentiment analysis. Therefore, a new compression model for sentiment sentences should be established



III. PROPOSED SYSTEM

In this section, we first give the task definition of two tasks, namely sentiment classification and sentence segmentation.

A. Task Definition

The task of sentence-level sentiment classification has been well formalized in previous studies[1], [2]. The objective is to determine the sentiment polarity of a sentence as positive or negative4. For example, the sentence "The new Star Trek movie is visually spectacular." expresses a positive opinion, and "The new Star Trek movie does not have much of a story." Expresses a negative opinion .The task of sentence segmentation aims to split a sentence into a sequence of exclusive parts, each of which is a basic computational unit of the sentence. The original text "that is not bad" is segmented as "that is not bad." The segmentation result is composed of three basic computational units, namely that, is and not bad.

B. The Prediction Process

We describe the prediction process of the proposed joint framework. The intuition of the prediction process is that Segmentation results have a strong influence on the sentiment classification performance, since they are the inputs of the sentiment classification model. We formalize the prediction process in Algorithm 1. Given a sentence whose polarity to be predicted, we first generate its segmentation candidates with CG. We then assign a real-valued score to each segmentation with the segmentation ranking model. Afterwards, we select the top-ranked segmentation results , and predict their polarity labels with the sentiment classification model. Finally, the polarity of a sentence is assigned with the label that majority of its top-ranked segmentations are assigned .

C. The Training Process

The training process of the proposed joint framework is illustrated in Fig. 3. The intuition of the training process is that: The usefulness of a segmentation can be judged by whether the sentiment classifier can use it to predict the correct sentence polarity. Based on this consideration, we formalize the training process in Algorithm 2. Each sentence in the training data is annotated with only sentence level sentiment polarity without any segmentation level annotations. There are two feature extractors for the task of

Algorithm 1 The prediction algorithm of the joint framework for sentence-level sentiment classification

Input:
test data: $T' = [s_i], 1 \le i \le |T'|$
segmentation feature extractor: $sfe(\cdot)$
classification feature extractor: $cfe(\cdot)$
candidate generation model: $\boldsymbol{CG}$
sentiment classifier: $\boldsymbol{SC}$
segmentation ranking model: $\boldsymbol{SEG}$
Output:
test data with predicted polarity: $T' = [s_i, \hat{pol}_i]$, $1 \le i \le |T'|$

```
for i ← 1 ... |T'| do
    Generate segmentation candidates Ω_i for each sentence s_i in T' based on CG,
    Calculate the segmentation scores for Ω_i. based on SEG and sfe(Ω_i.)
    Select the top-ranked K segmentation candidates Ω̄_i from Ω_i
    for j ← 1 ... |Ω̄_i| do
        Predict the sentiment polarity pol_ij for Ω̄_ij based on SC and cfe(Ω̄_ij)
    end for
    pol̂_i ← majority(pol_ij), 1 ≤ j ≤ K
end for
```

sentence segmentation and sentiment classification, respectively. The outputs of the training procedure are segmentation ranking model and sentiment classifier. In Algorithm 2, we first generate segmentation candidates for each sentence in the training set . Each contains no less than one segmentation candidates. We randomly select one segmentation result from each and utilize their classification features to initialize the sentiment classifier.We randomly initialize the segmentation model. Subsequently, we iteratively train the segmentation model and the sentiment classifier in a joint manner. At each iteration, we predict the sentiment polarity.

D. Naive Bayes Algorithm

Naive bayes classifier technique is used to classify the new cases i.e to decide which class label they belong to.We uses this approach to classify the received words to classify whether it is good or bad.If the probability ofoccurance of good word is more than bad word then the new case is likely to have membership of good rather than bad.Inbayesian analysis, this is known as prior probability. It is based on previous experience, so that percentage of good and bad words, and often used to predict outcomes before they actually occurs.

E. support Vector Algorithm

Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. The n-gram features and Support Vector Machine are widely-used baseline methods to build the sentiment classifier. We use LibLinear to train the sentiment classifier with linear kernel SVM

Algorithm 2 The training algorithm of the joint framework for sentence-level sentiment classification

Input:
training data: $T = [s_i, pol_i^g], 1 \le i \le |T|$
segmentation feature extractor: $sfe(\cdot)$
classification feature extractor: $cfe(\cdot)$
candidate generation model: $\boldsymbol{CG}$
Output:
sentiment classifier: $\boldsymbol{SC}$
segmentation ranking model: $\boldsymbol{SEG}$

```
Generate segmentation candidates Ω_i for each sentence s_i in T based on CG, 1 ≤ i ≤ |T| //corresponding to Fig. 4.
Initialize sentiment classifier SC^(0) based on cfe(Ω_ij), randomize j ∈ [1, |Ω_i|], 1 ≤ i ≤ |T|
Randomly initialize the segmentation ranking model SEG^(0)
for r ← 1 ... R do
    for i ← 1 ... |T| do
        Predict the sentiment polarity pol_i for Ω_i based on SC^(r−1) and cfe(Ω_i.)
        Update the segmentation model SEG^(r) with SEG^(r−1) and [Ω_i, sfe(Ω_i.), pol_i., pol_i^g], 1 ≤ i ≤ |T|
        //corresponding to Fig. 5.
    end for
    for i ← 1 ... |T| do
        for j ← 1 ... |Ω_ij| do
            Calculate the segmentation scores for Ω_ij based on SEG^(r) and sfe(Ω_ij)
        end for
        Select the top-ranked K segmentation candidates Ω̄_i from Ω_i
    end for
    Train the sentiment classifier SC^(r) with cfe(Ω̄_ij), 1 ≤ i ≤ |T|, 1 ≤ j ≤ |Ω̄_i| //corresponding to Fig. 7.
end for
SC ← SC^(R)
SEG ← SEG^(R)
```

E. Module Description

1.Collection of GENERAL Messages:

First step is to collect messages from the buyers in a particular subject through GENERAL messages and from buyer login. Next step is to create database for the collected messages. Database contains number of messages received from the buyers, received message, date and time.

2.Spell check and Grammar Correction:

In this, there is spell checker . We can correct the spelling mistakes in each document, if any. It will suggest some words from dictionary. We select from the list for changing it.In this, we correct the grammar of a sentence in each document , if any. It will show the correct sentence options for changing.

3.Stemming:

It is process of mapping of all the alternatives to the ROOT word. It will identify the tense of the word. For example, the root word "go" has the following variations – go, went,gone.Affix removal conflation techniques are referred to as stemming algorithms and can be implemented in a variety of different methods. All remove suffices and/or prefixes in an attempt to reduce a word to its stem. The algorithms that are discussed in the following sections, and those that will be implemented in this project, are all suffix removal stemmers. During the development of a stemmer the issues of iteration and context awareness must be addressed. Suffices that are concatenated to words are often done so in a certain order, such that a set of order-classes will exist among suffices. An iterative stemming algorithm will remove suffices one at a time, starting at the end of the word and working towards the beginning. It will parse the data and tag the parts-of speech for every word.

4.Sentiment Analyzer:

This is very important module. The system will analyse each and every message in relevant group individually using sentiment analysis. It gives each and every person feedback as positive or negative or average about the lecture in graph format.following are the steps involved in sentiment analysis.

IV. CONCLUSION

We present a framework for using a sentiment sentence compression modelfor aspect-based sentiment analysis. Different from the common sentence compression model, not only compresses the redundancy in the sentiment sentences, but also needs to retain the polarity-related information to maintain the sentences' original polarities. Thus, the over-natural and spontaneous sentiment sentences can be compressed into more formal and easier-to-parse sentences after using the Sentiment analysis model. Accordingly, the most important features for the aspect-based sentiment analysis, i.e., syntactic features, can be more correctly acquired to enhance the performance of this task. The sentiment sentence compression can be converted to determine each word in a sentiment sentence to be classified as "delete" or "reserve." In this paper, we establish a CRF-based compression model with rich features, including sentiment-related features and potential semantic features. We conduct several experiments on the corpora of four product domains to evaluate the effectiveness of the feature sets used for sentiment analysis modeland the effectiveness of thismodel applied in the aspect-based sentiment analysis. Our experimental results can validate the following points: Comparing the feature sets sentiment-related feature (SF) and the potential semantic feature (PSF) used in modelling project, the PSF set, which uses three types of methods, i.e., suffix/prefix character features, Brown word clustering features, and word embedding features, is more effective in generalizing the words in sentiment sentences. Sentence compressionis proven to be effective for the aspect-based sentiment analysis, which can also demonstrate that the CRF based method in Section II is effective for the sentiment sentence compression task. Sentiment analysisis useful for the Offense Word Analysis tasks that rely heavily on syntactic features, such as the aspect-based sentiment analysis. However, for the tasks in which the syntactic features are not necessary, they cannot benefit from using the sentiment analysismodel. Sentiment analysisis domain-independent

V. FUTURE WORK

In the future, the idea of using sentiment sentence compression for aspect based Rude Word Analysis can be considered as a basic framework. We believe that more Rude Word Analysis tasks that rely heavily on syntactic features will benefit from the sentiment sentence compression model.

VI. REFERENCES

[1] WanxiangChe, Yanyan Zhao, HongleiGuo, Zhong Su, and Ting Liu *Sentence Compression for Aspect-Based Sentiment Analysis* december 2015

[2] B. Liu, Sentiment Analysis and Opinion Mining, ser. Synt hesis Lectureson Human Language Technologies.San Rafael, CA, USA: Morgan & Claypool, 2012.

[3] E. Cambria, B. Schuller, Y. Xia, and C. Havasi, "New ave nues in opinion mining and sentiment analysis," IEEE Intell. Syst., vol. 28, no. 2, pp. 15–21, 2013.

[4] T. Wilson, "Finegrained subjectivity and sentiment analysi s: Recognizing the intensity, polarity, and attitudes of private s tates," Ph.D. dissertation, Intell. Syst. ProgramUniv. of Pittsbu rgh, Pittsburgh, PA,USA, 2007.

[5] A. Mukherjee and B. Liu, "Aspect extraction through semi-supervised modeling," in Proc. 50th Annu. Meeting Assoc.
Comput. Linguist. (Vol.1: Long Papers), Jeju Island, Korea, Jul. 2012, pp. 339–348, Assoc. for Comput. Linguist..

[6] G. Qiu, B. Liu, J. Bu, and C. Chen, "Opinion word expansion and target extraction through double propagation," Comput. Linguist., vol. 37, no1, pp. 9–27, 2011.

[7] M. Hu and B. Liu, "Mining and summarizing customer reviews," in Proc. KDD'04, 2004, pp. 168–177.

[8] A. Abbasi, H. Chen, and A. Salem, "Sentiment analysis in multiplelanguages: Feature selection for opinion classification in web forums,"ACM Trans. Inf. Syst., vol. 26, no. 3, pp. 12:1–12:34, Jun. 2008.

[9] A. Duric and F. Song, "Feature selection for sentiment analysis basedon content and syntax models," Decis. Support Syst., vol. 53, no. 4, pp.704–711, Nov. 2012.

[10] W. Che, Z. Li, and T. Liu, "Ltp: A Chinese language technology platform," in Coling 2010: Demonstrations, Beijing, China, Aug. 2010,pp. 13–16, Coling 2010 Organizing Committee.

[11] K. Bloom, N. Garg, and S. Argamon, "Extracting appraisal expressions," in Proc. HLT-NAACL'07, 2007, pp. 308–315.

[12] L. Xu, K. Liu, S. Lai, Y. Chen, and J. Zhao, "Mining opinion words andopinion targets in a twostage framework," in Proc. 51st Annu. MeetingAssoc. Comput. Linguist. (Vol. 1: Long Papers), Sofia, Bulgaria, Aug.2013, pp. 1764-1773, Assoc. For Comput. Linguist..

[13] K. Knight and D. Marcu, "Summarization beyond sentence extraction:A probabilistic approach to sentence compressio-n," Artif. Intell., vol.139, no. 1, pp. 91–107, Jul. 2002.

[14] J. Turner and E. Charniak, "Supervised and unsupervised learningfor sentence compression," in Proc. 43rd Annu. Meeting Assoc. forComput. Linguist., Stroudsburg, PA, USA, 2005, pp. 290–297, ser.ACL '05, Assoc. ForComput. Linguist..