International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 3 | ISSN : 2456-3307

Extraction of Feature in Opinion Mining using Domain Relevance

Prasad P. Mahale
Computer Engineering Department, R. C. Patel Institute of Technology, Shirpur, Dhule, Maharashtra, India

ABSTRACT

Opinion mining is determining people's opinions, about entities such as products, services and their attributes. Opinion feature identification from online reviews evaluated in two domain corpora, one is domain specific, other is domain – independent corpus, and this evaluation is based on number of occurrence of that feature. Domain relevance is used to measure distribution. By applying a set of syntactic rules, identify candidate features in user reviews is primary task. Features extracted from this are specific to a domain. For each extracted candidate feature, extrinsic domain relevance and intrinsic domain relevance value are calculated.

Keywords: Extract, Opinion, features, Mining, Domain.

I. INTRODUCTION

People's opinions, sentiments, and attitudes toward entities such as products, services, and their attributes are determined by opinion mining [1].

Opinion is address as a quadruple describe Topic, Holder, Claim and Sentiment that the Holder believes a Claim about the Topic and in many cases associates a Sentiment with the belief. Take the sentence "John said that battery of cell phone was good" for example, John and cell phone are the holder and topic of the opinion respectively; he claims "good" on the cell phone, which involves a positive sentiment.

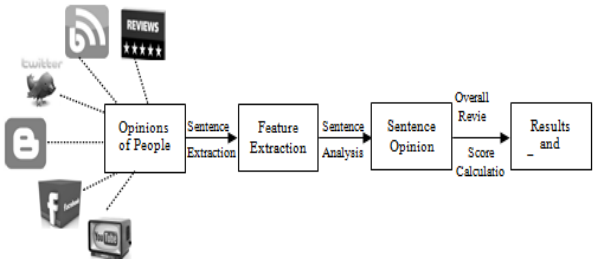

Figure 1. Process of opinion mining [7]

Generally, individuals and companies are constantly excited in other's opinion like if someone wants to purchase a new product, then mostly; he/she tries to know the reviews. Similarly, companies also used consumer reviews to improve the quality of product. Digital ecosystem has a plethora for same in the form of blogs, reviews etc. Opinion mining has major step to extract opinion feature. The process of opinion mining shown by figure 1[7]

Opinion mining also referred as sentiment analyses, which direct analysing social opinions, sentiments and temperament towards entities such as services, products and their features. Opinions expressed in textual reviews are usually scrutinized on various dimensions. Opinion mining generally works at two levels document level and feature level.

In opinion mining, opinion feature indicates an attribute or entity on which user express their opinion In opinion mining the opinion that are express in textual form are analyses at various level such as document level opinion mining and sentence level opinion mining. Document level opinion mining has less performance than sentence level opinion mining. Nowadays customer is not satisfied

with the overall rating of the product but them wants to know the positive and negative attribute of the product. Therefore, it is very important to extract valid opinion feature from the text reviews and associate them to opinion.

To identify opinion features from user opinions on any product. These opinions are primary role in sale of the product. Opinion feature are identify on which user give their opinion in the user review. Opinion feature extraction done in single corpus without considering nontrivial distribution of word IDER method for mining features in user opinions from two types of corpus one is domain dependent and other is domain independent. Supervised learning model give better performance in a domain dependent corpus, but the model give less performance if it is applied to different domains.

The selection of domain independent corpus such that occurrence of feature in user review is more in domain dependent corpus than the domain independent corpus. Screen resolution is one of the features, which may be occurring in both cell phone domain and laptop domain. The occurrence of feature is more in cell phone domain and relatively less in laptop domain. Two domains are used to extract feature to improve the performance by this approach. Domain relevance score is calculated both domain (i.e. domain dependent and domain independent) by this approach. Domain relevance score calculated from domain dependent corpus is called as intrinsic domain relevance, in other case domain relevance score calculated from domain independent corpus is called as extrinsic domain relevance. The application of Intrinsic Extrinsic domain relevance on results of preceding step yields accurate opinion features from user opinions.

The organization of this paper is as follows. In Section II, we present the work related to the different opinion feature extraction methods. The working of the proposed system is discussed in section III. Section IV explains algorithm. Section V discusses the details of experiment with results. Finally, in section VI we conclude and discuss future scope.

II. RELATED WORK

The problem of extracting product feature in opinion mining has been studied. There are some extracting feature techniques are available as follow.

Blei et al.[2003] proposed work unsupervised topic modeling approaches, such as latent Dirichlet allocation (LDA), which is a generative term-topic-document three way probabilistic model, have been used to solve aspect-based opinion mining tasks. Mining latent topics or aspects is primary goal of this model. Latent topics actually correspond to distinguishing properties or concepts of the commented entities, and may not necessarily be opinion features expressed explicitly in reviews are effective in discovering latent structures of review data; they may be less successful in dealing with identifying specific feature terms commented on explicitly in reviews [3].

Hu et al.[2004] proposed work for mine and summarization all customer review of product. Identification of opinion sentence in review depends on product feature. Data mining and NLP technique extract product feature. Association rule mining (ARM) approach to mine frequent item sets as potential opinion features, which are nouns and noun phrases with high sentence-level frequency. However, ARM, which relies on the frequency of item set [4].

Qiu et al.[2008], has proposed work in which initially the User-generated Content (UGC), is used for opinion mining which is such a kind of novel media content produced by end-users . This method perform two task namely topic extraction and sentiment classification. In this approach topic extraction, first extract topics from opinion sentences

through syntactic parsing of sentences using the dependency grammar. System builds a super set from a corpus to filter out these noises. Contextual information is used to identify polarity of word and apply sentiment classification. The sentiment classification is done by syntactic knowledge. Quantify the content polarity sentiment words algorithm used [2].

Su et al.[2008] has proposed work a novel mutual reinforcement(MRC) approach to deal with the feature-level opinion mining problem. Content information and sentiment link information fuse iteratively and clusters product features and opinion words simultaneously by this approach. MRC construct the sentiment association set among the product feature categories and opinion word groups of data objects by finding their strongest *n* sentiment links. Moreover, knowledge from multi-source is incorporated to improve clustering in the procedure. Based on the pre-constructed association set, MRC approach can largely predict opinions relating to different product features, even for the case without the explicit appearance of product feature words in reviews. Opinion feature word have hidden sentiment link with opinion word. Thus, it provides a more accurate opinion evaluation [5].

Yu et al.[2011] aspect ranking, which aims to automatically identify important product aspects from online consumer reviews. A large number of consumers usually comments the important aspects of a product and consumers' opinions on the important aspects greatly influence their overall opinions on the product. Shallow dependency parser extracts the product aspect from consumer reviews of a product. Consumer's opinions on these aspects determine using a sentiment classifier. System used an aspect-ranking algorithm to identify the important aspects by simultaneously considering the aspect frequency and the influence of consumer's opinions given to each aspect on their overall opinions. Consumer's overall opinion rating on a product is generated based on a weighted sum of his/her specific opinions on multiple aspects of the product, where the weights essentially measure the degree of importance of the aspects. A probabilistic regression algorithm is then developed to derive these importance weights by leveraging the aspect frequency and the consistency between the overall opinions and the weighted sum of opinions on various aspects [6].

III. METHODOLOGY

Figure 2 shows the system work flow of our proposed method. The dataset is an unstructured dataset of documents, which are pre-processed to remove noise. In the pre-processing removal of other data except review sentence.

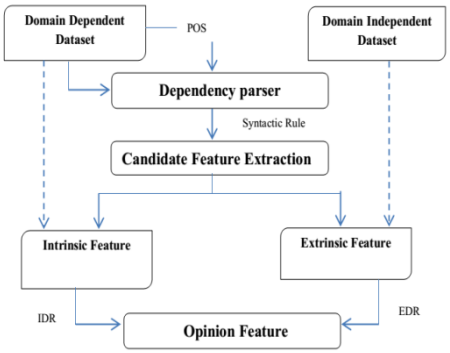

Figure 2. System workflow

Stop Word Removal: Sometimes a very common word, which would appear to be of little beneficial in helping to select documents matching user's need, is completely excluded from the selected documents. These words are treated as stop words and this technique is called stop word removal. The general strategy for determining a stop list is to sort the terms by collection frequency and then to make the most frequently used terms are treated as stop list, the members of which are discarded during indexing. Some of the examples of stop-word are a, an, the, and, are, as, at, be, for, from, has, he, in, is, it, its, of, on, that, the, to, was, were, will, with etc. Here the input stop word file contains 546 words.

Dependency parser used to extract noun and noun phrases. Product features are usually nouns or noun phrases in review sentences. Thus, the part-of-speech tagging is crucial. **Part –of-Speech Tagging (POS)** we used the Stand-ford parser [9] to parse each reviews to split text into sentences to tokenize each word with their proper abbreviation with the help of part of speech tagging. After tagging each word will be, identify with their proper tag like, noun, verb, adjective, etc. This process also identifies noun, proper noun, noun phrases and verb groups, which we call it as syntactic chunking. The following shows a sentence with POS tags

Original/JJ unlocked/VBN nokia/NN c6-01/JJ 3g/CD gps/NNS 8mp/JJ 1981Bluetooth /NN touchscreen/NN cell phone/NN

After tagging, each sentence is store in the transaction database along with their POS tag information for each word in the sentence. Each line contains words of every sentence, which consist only the identified nouns and noun phrases of the sentences. Other part of the sentence is unlikely to be product features. Removal of stop words includes in pre-processing technique. i.e., noise free data. After Tagging, all sentences applied some syntactic rules through which we can extract only Nouns (NN) and Noun phrases (NNP) as an opinion features. From the above example we get the features **Nokia, Bluetooth, GPS, cell phone, touch screen** after applying rules.

In the case of dependence grammar, the subject opinion feature has a syntactic relationship of type subject verb with the sentence predicate (usually adjective or verb). The object opinion feature has a dependence relationship of verb-object on the predicate. In addition, it also has a dependence relationship of preposition-object on the prepositional word in the sentence. Some syntactic relation examples in Chinese are listed in Figs. 4.1 and 4.2, with their corresponding dependence trees. The letter "V" in both SBV and VOB in the figure indicates the predicate of a review sentence.[1]

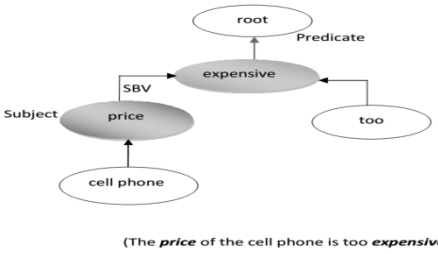

Figure 3. SBV dependency relation

As shown in the dependence tree in Figure 4.1, the opinion feature "price" (underline), which is associated with the adjective "expensive" (italic), is the subject of the sentence.

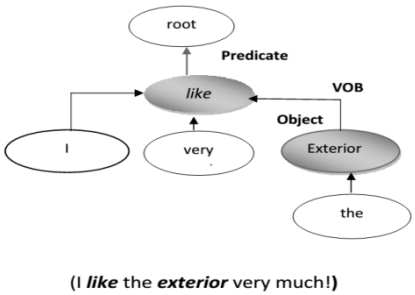

Figure 4. VOB dependency relation

It has a dependence relation of SBV with the adjective predicate. In Fig. 4.2, the noun feature "exterior" is the object of the verb predicate "like," and thus has a VOB dependence relation with the predicate. From the aforementioned dependence relations, i.e., SBV, VOB, and POB, present three syntactic rules in Table 1, where "NN" and "CF" denote nouns (noun phrases) and candidate features, respectively. For example, by employing the first rule in Table 1 to the example, for extract the noun "price" as a candidate feature, as shown in Fig. 4.1, which has an SBV relation with the adjective predicate "expensive."

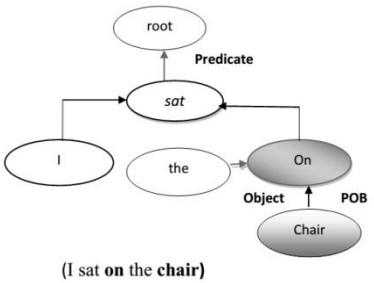

Figure 5. POB dependency relation

The candidate feature extraction process works in the following steps: 1) Dependence parsing is first employed to identify the syntactic structure of each sentence in the given review corpus; 2) By firing three rules which are listed in table no 1 identify dependency structure and corresponding nouns or noun phrases are extracted as candidate features.

Table 1. Syntactical Rules

Rules	Interpretation
NN + SBV -> CF	Identify NN as a CF, If NN has a SBV dependency relation
NN + VOB -> CF	Identify NN as a CF, If NN has a VOB dependency relation
NN+ POB-> CF	Identify NN as a CF, If NN has a POB dependency relation

Opinion Feature Extraction

There could be many invalid features in the extracted candidate feature list; the next step is to prune the list via the IEDR criterion. Opinion feature extraction is dependents on some statistical terms.

Domain Relevance

How much a term is related to a particular corpus is calculated with domain relevance. Domain relevance is calculated with help of two statistics namely Dispersion and Deviation.

How significantly a term is mentioned across all documents by measuring the distributional relevance of the term over different documents in the entire corpus is called as dispersion. How frequently a term is mentioned in a particular document by measuring its distributional significance in the document is called as deviation

Term frequency –inverse document frequency term weight is used to calculate both dispersion and deviation. Each term Ti has a term frequency TF_{ij} in a document D_j, and a global document frequency DF_i.The weight w_{ij} of term T_i in document D_j is then calculated as follows:

$$W_{ij} = \begin{cases} (1+\log TF_{ij}) \times \log \frac{N}{DF_{ij}} & if\ TF_{ij} > 0, \\ 0 & otherwise \end{cases} \quad \ldots(1)$$

Where i= 1,....M for the total numbers of M terms, and j=1,.....N for a total number of N document in the corpus.

The standard variance s_i for term T_i is

$$s_i = \sqrt{\frac{\sum_{j=1}^{N} (w_{ij} - \overline{w_i})^2}{N}} \quad \ldots(2)$$

calculated as follows:

Here the average weight w_i of term T_i across all documents is calculated by

$$\overline{w_i} = \frac{1}{N} \sum_{j=1}^{N} w_{ij} \quad \ldots(3)$$

Dispersion $disp_i$ of each term Ti in the corpus is defined as follows:

$$disp_i = \frac{\overline{w_i}}{s_i} \quad \ldots(4)$$

Dispersion thus measures the normalized average weight of term Ti. It is high for terms that appear frequently across a large number of documents in the entire corpus. The deviation $devi_{ij}$of term Ti in document D_j is given by

$$devi_{ij} = w_{ij} - \overline{w_j} \quad \ldots(5)$$

Where the average weight W_j in the document D_j is calculated over all M terms as follows:

$$\overline{w_j} = \frac{1}{M} \sum^{M} w_{ij} \quad \ldots(6)$$

Deviation $devi_{ij}$ indicates the degree in which the weight w_{ij} of the term Ti deviates from the average w_j in the document D_j. The deviation thus characterizes how significantly a term is mentioned in each particular document in the corpus. The domain relevance dr_i for term Ti in the corpus is finally defined as follows:

$$dr_i = disp_i \times \sum_{j=1}^{N} devi_{ij} \quad \text{.....(7)}$$

The domain relevance dr_i incorporates both horizontal ($disp_i$) and vertical ($devi_{ij}$) distributional significance of term T_i in the corpus. The domain relevance score thus reflects the ranking and distributional characteristics of a term in the entire corpus.

Intrinsic and Extrinsic Domain Relevance

Intrinsic-domain relevance of an opinion feature is computed domain relevance in domain specific review corpus. Extrinsic-domain relevance is referred as domain relevance of the same opinion feature calculated on a domain-independent corpus. IDR reflects the specificity of the feature to the domain review corpus (e.g., cell phone reviews), while EDR characterizes the statistical association of the feature to the domain-independent or generic corpus. Candidate feature is related to either one or the other, but not both. The irrelevance of a feature to the given domain review corpus gives by EDR.

The domain relevance is calculated with help of opinion and its reliance upon the domain. Opinion feature when calculated on a domain related review gives intrinsic-domain relevance. Similarly, opinion feature calculated on a different, independent domain called extrinsic-domain relevance. IDR represents how much the candidate feature is related to the given domain corpus and EDR represents the relevance of the candidate to the domain independent corpus. Candidate feature with low EDR and high IDR are confirmed as opinion feature. This threshold approach is called the intrinsic extrinsic domain relevance criterion

Pearson Correlation

The correlation coefficient is a measure of how two domains are related to each other. A correlation of 1 means both domains have perfect positive linear relationship and -1indicates negative relationship. Pearson correlation is given by the preceding equation. X and y represents two cross-domain dataset

$$Corr(x,y) = \frac{c_i\ (x,y)}{s_i\ (x) \times s_i(y)} \quad \text{.....(8)}$$

Where,

c_i is covariance between x and y

s_i is standard deviation

IV. ALGORITHM

The procedure for computing the domain relevance is the same regardless of the corpus, as summarized in Algorithm 1. When the procedure is applied to the domain-specific review corpus, the scores are called IDR, otherwise scores are called EDR.

Algorithm 1: Calculating Intrinsic / Extrinsic Domain Relevance (IDR/EDR)

Input: A domain specific / Independent corpus C

Output: Domain relevant scores

for each candidate feature CF_i do

for each document $D_j$in the corpus C do

Calculate weight W_{ij}

Calculate standard deviation S_i

Calculate dispersion $disp_i$

for each document $D_j$in the corpus C do

Calculate deviation $devi_i$

Compute domain relevance dr_i

Return a list of domain relevance features for all candidate features;

IEDR criteria pruned candidate feature with high EDR score or low IDR score. Algorithm 2 summarizes the gives IEDR approach, where the minimum IDR threshold i_{th} and maximum EDR threshold eth can be determined experimentally. A sample run of the IEDR algorithm on a toy example is given:

Example– "The screen of Iphone5 looks really beautiful and its battery is okay for me. I am one of

its many fans and I really want to have one, but it is too expensive, and I have no money now!"

It shows a sample product review on iPhone 5. Here both nouns "screen" and "battery" is opinion features. Applying Algorithm 2 on the example as follows: First, apply the syntactic rules defined in Table 1 to extract a list of candidate features (nouns): "screen," "battery," "fans," and "money." Next, prune the four candidate feature using IEDR, to obtain the final confirmed set of opinion features: "screen" and "battery,".

Algorithm 2: Identifying opinion features via IEDR

Input: Domain review corpus R and domain-independent corpus D

Output: A validated list of opinion features

Extract candidates from the review corpus R;

for each candidate feature CF_i **do**

Compute IDR score idr_i

Via algorithm 1in review corpus R;

Compute EDR score edr_i

Via algorithm 1 in domain-independent corpus D;

If ($idr_i >= i^{th}$) AND ($edr_i <= eth$) **then**

Confirm candidate CF_i as a feature;

return validate set of opinion features;

For comparison, list of the extracted opinion features when only one of the two measures is used. IEDR combines both thresholds to prune both "fans" and "money," resulting in two correct features

V. EXPERIMENTAL RESULTS

All experiments have been performed on an Intel Core i5 @ 2.20GHz with 8GB of main memory under Windows 7. The proposed system is implemented in Java using jdk1.8 version.

A. Corpus Description

Experimental Description is based on three different real word reviews, which are taken from different social networking sites. We perform experiment on cell phone [10][11], camera[12] review dataset.

Table 2. Corpus Description

	Cellphone	Camera
Number of Reviews Entries	2051	2007
Number of Sentences Entries	3051	2322
Review entries after preprocessing	2051	2058
Number of feature in domain	1245	1211
Total number of retrieved feature	173	40
Number of corrected feature	50	19

B. Evaluation Metric

To evaluate the effectiveness of our proposed features extraction algorithm we use standard evaluation measures i. e., Precision, Recall and F-measure. We first extracted candidate features from the given review domains, i.e., cellphone, camera and hotel reviews, using the syntactic rules defined in Table 1.We calculated precision ,recall and f-measure value for three dataset. This value are calculated as follow,

- Precision= Corrected Feature/ Retrieved Features
- Recall= Retrieved Features/ Features in domain
- F-measure= (2(Precision*Recall))/(Precision + Recall)

Table 3. Precision ,Recall and F-measure value for three dataset

Dataset	Precision	Recall	F-measure
Cellphone	0.2890173	0.138956	0.18767832
Camera	0.6	0.024773	0.04758128

Table shows Precision, Recall and F-measure value calculate for candidate feature for datasets.

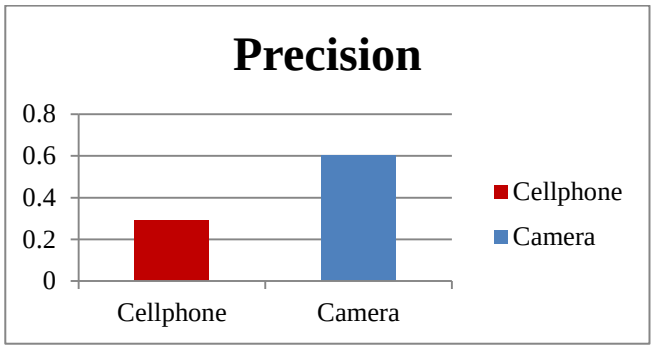

Figure 6. Precision comparison for three dataset

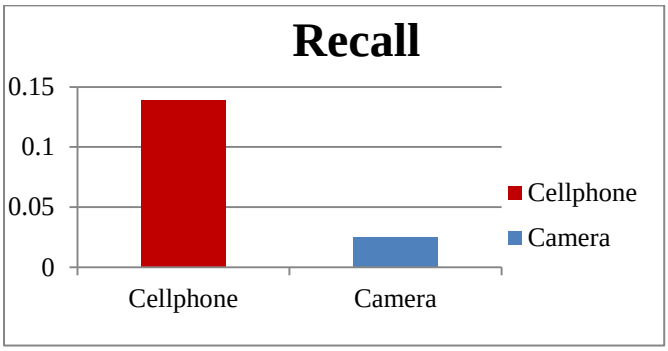

Figure 7. Recall comparison for datasets

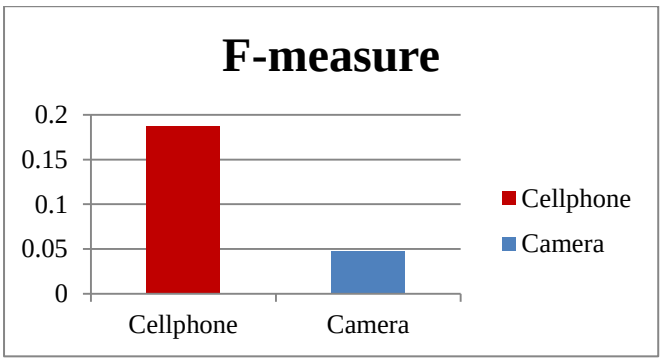

Figure 8. F-measure comparison for datasets

Table 4. Percentage of F-measure for different percentage threshold

Threshold in %	5	10	15	20	25
F-measure in % cellphone	12.33	18.44	18.8	19	19.03
F-measure in % camera	3.84	4.44	4.9	5.05	5.2

For IEDR approach, we calculate percentage F-measure value to the different percentage of threshold value. Table shows that extracted opinion feature remain constant after specific threshold value. For IEDR approach, we calculate percentage F-measure value to the different percentage of threshold value.

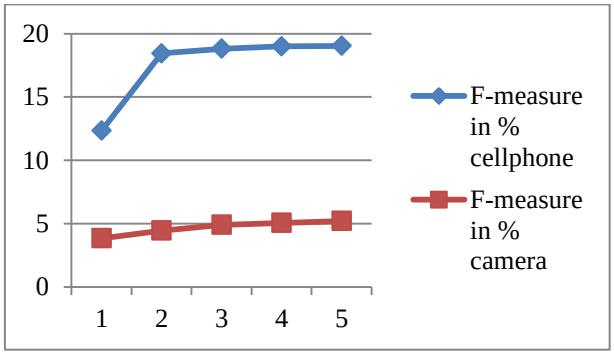

Figure 9. IEDR F-measure performance verse IDR threshold

VI. CONCLUSION

IEDR filtering criteria is used to extract opinion feature from domain dependent and domain independent corpus. IEDR algorithm recognize candidate feature that are more specific to the given reviews. Experimental result show that extracted opinion feature remains constant after specific threshold value. Precision value decreases according to the threshold value increases.

VII. REFERENCES

[1]. Kuiyu Chang, Jung-Jae Kim, and Christopher C. Yang Zhen Hai, "Identifying Features in Opinion Mining via Intrinsic and Extrinsic Domain Relevance," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, vol. 26, no. 3, pp. 623-634, March 2014.

[2]. B. Liu, "Sentiment Analysis and Opinion Mining," Synthesis Lectures on Human Language Technologies, vol. 5, no. 1, pp. 1-167, May 2012.

[3]. C. Wang, J. Bu, K. Liu, and C. Chen G. Qiu, "Incorporate the Syntactic Knowledge in Opinion Mining in User-Generated Content," Proc. WWW 2008 Workshop NLP Challenges in the Information Explosion Era, 2008.

[4]. A.Y. Ng, and M.I. Jordan D.M. Blei, "Latent Dirichlet Allocation," J. Machine Learning Research, vol. 3, pp. 903-1022, March 2003.

[5]. M. Hu and B. Liu, "Mining and Summarizing Customer Reviews," Proc. 10th ACM SIGKDD Int'l Conf. Knowledge Discovery, pp. 168-177, 2004.

[6]. X. Xu, H. Guo, Z. Guo, X. Wu, X. Zhang, B. Swen, and Z. Su, Q. Su, "Hidden Sentiment Association in Chinese Web Opinion Mining," Proc. 17th Int'l Conf. World Wide Web, pp. 959-968, 2008.

[7]. Z.-J. Zha, M. Wang, and T.-S. Chua, J. Yu, "Aspect Ranking:Identifying Important Product Aspects from Online ConsumerReviews," Proc. 49th Ann. Meeting of the Assoc. for Computational Linguistics: Human Language Technologies, pp. 1496-1505, 2011.

[8]. MohdShahid Husain Pravesh Kumar Singh, "METHODOLOGICAL STUDY OF OPINION MINING AND SENTIMENT ANALYSIS TECHNIQUES," International Journal on Soft Computing (IJSC), vol. 5, no. 1, pp. 11-21, February 2014.