Opinion Mining from User Reviews

Amiya Kumar Tripathy[1*], Revathy Sundararajan[2], Chinmay Deshpande[3], Pankaj Mishra[4], Neha Natarajan[5]
[1, 2, 3, 4, 5] Department of Computer Engineering, Don Bosco Institute of Technology, Mumbai, India
[1*] School of Computing and Security Sciences, Edith Cowan University, Perth, Australia
[1]tripathy.a@gmail.com, [2]revathy9@hotmail.com, [3]dchinmay21@gmail.com, [4]panksmish@gmail.com,
[5]natarajan.neha2711@gmail.com

***Abstract* – Due to advancement of technology and mainly Internet, the concept of marketing and selling of product has reached to a new level. Now-a-days, lots of companies rely on user reviews for launching their product. These reviews play an important role or companies to know how their product has been accepted in the market. But, today, thousands of reviews are generated for a product. Companies have to process each of these reviews to get user opinion as well as ideas, which is a very tedious and time-consuming. This paper discourses about extracting opinions from the user reviews is semi-automatic, in the sense that it requires some amount of expert assistance. Expert assistance is required for building the domain knowledge for the system, so as to make the system learn about the domain specific words[1]. The proposed system, using domain knowledge, identifies and extracts the opinions for a given product. These extracted opinions include the opinion words, their polarity in from of weights and for which feature these opinions was provided and system aggregates the extracted opinions them for better display.**

Keywords – data mining, opinion mining, sentiment analysis, user reviews, opinion extraction, hidden markov model, data extraction.

I. INTRODUCTION

Opinion Mining is a natural language processing and information extraction task that aims at obtaining writer's feelings expressed in comments, questions and requests by analyzing a large numbers of documents [1][2]. In recent years, exponential use of web as a medium of communication has led to the generation of a huge quantity of unstructured data. Web data is increasing day by day exponentially and it became next to impossible to analyze and interpret such huge volume of data. The solution to this problem was opinion mining from these available data in the automated way. The key steps in opinion mining are opinion extraction and structurization which helps in aggregation and analysis of opinion on pre-decided subjects [1]. Extraction of opinion also includes identification of opinion holder, subject of the review as well as concluding the response either positive or negative.

Structurization involves transformation of the extracted opinion expressions into structures suitable for assimilation and analysis. In this paper, we have discussed about our working implementation model which uses SentiWordNet as the dictionary, Stanford Core NLP as the language processor along with Hidden Markov Model (HMM) concepts[9][1], over our chosen domain of television. Our system first takes the user review, and then processes it for obtaining all the subject and feature by POS (parts of speech) tagging. It then maps these subject and features with each other. On the basis of this tagging and our algorithms, the system then calculates the weights for each subject-feature pair, based on which polarity is assigned to it. Next, all these weights are combined to have final weight and polarity for a review. The result is then displayed in form of pie-charts, bar-charts. We have also included an alert module which will aid the expert in removing any unwanted words entered into domain knowledge and thus maintaining consistency in domain knowledge.

II. LITERATURE SURVEY

Opinion Mining is a field which has got a lot of attention from researchers in last few years. Some of the works have been discussed here.

Hatzivassiloglou and McKeown proposed the use of a supervised learning algorithm to infer the semantic orientation of adjectives from constraints on conjunctions [1]. They used a list of seed words to determine whether a sentence contains positive or negative sentiments. Yi and Nasukawa (2005) built a dictionary of polarity lexicons to extract sentiments from a sentence. Benamara et al. proposed the use of adverb–adjective combines (AAC) [1]. Adverbs were classified into five categories. Based on this classification, a set of general axioms were defined that were to be satisfied by all adverb scoring techniques.

Ghose et al. proposed an altogether different methodology to measure the strength and polarity of an opinion [1]. The idea here was to use the economic context in which the opinion is evaluated, instead of using human annotators or linguistic resources. The underlying assumption is that a product with more positive opinions sells at a higher price than a product with negative opinions. Thus the actual price of a product is used to assign opinion orientations. Popescu and Etzioni presented OPINE [1], an unsupervised information extraction system which mines reviews in order to build a model of important product features, their evaluation by reviewers, and their relative quality across products.

L. Dey and M. Haque (2009) discussed their work in the field of opinion mining on noisy text data [1]. Their paper presented the framework for a generic web-based opinion finder system. Their system employs a linguistic approach which exploits surface dependency rules to determine opinion expressions within noisy text data. The paper also proposed a new approach to deal with adverbial modifiers.

978-1-4799-8187-8/15/$31.00 ©2015 IEEE

III. METHODOLOGY

The most influential components in opinion mining process are the opinion words. These words basically have positive or negative orientation [2]. Along with them, there are certain words which have influence over these opinion words. These can be classified into two parts: enhancers and reducers. Enhancers are the words which have positive effect on the polarity of opinion word. For e.g., '*very expensive*', *'extremely heavy*'. Reducers are those which have negative effect on the opinion words. For e.g. 'slightly better display'. Hence all these words need to be taken in consideration while extracting the final opinion of the sentence. However these opinion words may not have same polarity across domains [1][13]. The polarity may get reversed if we use a word in another domain or even with different context in the same domain. For e.g. consider an opinion word 'low'. This word, in general, has a negative orientation. For mobile products, this word has different polarity for different review components. In context of mobile price, '*The mobile price is low'* has positive orientation. However, in respect to mobile quality, '*The mobile quality is low'* denotes negative orientation. Hence it is necessary to store these opinion words along with the orientation for each context. The basic process in opinion mining consists of two parts viz. Sentence analysis and opinion extraction. Sentence Analysis consist of finding out the Part-of Speech tags of each words present in the sentence. These parts of speech are important in finding out the required opinion words, modifiers as well as the subject of the sentence.Once the components and opinions words along with modifiers are obtained for a sentence, the next task is to relate them with each other. In simple words, the system should understand for which component a given opinion is given. Hence the main aim of Opinion Extraction process is to generate pairs of components with their relating opinion words and modifiers.

In general, the nature of the user review may not be in the form of simple English sentence. It may consist of multiple components, multiple opinion words and modifiers. Also a component can be related with one or many opinion words. For example, "*The display is huge, sharp and looks really elegant*". Here, the component '*display'* is related with multiple opinion words viz. '*huge', 'sharp'* and '*elegant'*. Further, a sentence may be a complex sentence consisting of various conjunctions like '*and', 'but',* etc. For example, '*The cost price is comparatively low and display is awesome but sound quality is not up to the expectations'*. Here the system must be able to relate '*cost price*' with '*comparatively low*', '*display*' with '*awesome*' and '*sound quality*' with '*not up to expectations*'. Hence the system needs to be trained to match the pairs of components and related opinion words.

The concepts of Hidden Markov Model can be used for training the system to generate the component-feature pair [11]. Here the observable states are the components and features in the sentence. From these observable states, the system has to find the hidden information whether the feature and component are related or not. This information can be found out by using the conditional probability.

Given a sequence of components c = c1, c2, c3… and given a sequence of features f = f1, f2, f3…. The correlation between the given pair (c, f) is 'r' which can take value 0 (the pair does not exist) or 1 (the pair exists). The task of finding the appropriate relation can be done as follows:

$$P(r|c,f) = \frac{P(c,f|r)*P(r)}{P(c,f)} \quad (1)$$

The value P(r|c,f) is then compared with a threshold value which can be predetermined by the expert. If the value is greater than the threshold value, the relation exists else the relation does not exist between the component and the feature. Based upon this method, all related pairs can be obtained for a given sentence.

A. Expert Knowledge

As discussed earlier, there is a need to store the domain relate opinion words and their orientation in the domain. Keeping these requirements in consideration, the system has an expert review module which will help in creating and maintaining the domain knowledge of the system. Here input to the system are domain-specific components, features assigned to them by experts along with suggested orientation of the feature. Each feature is then passed to SentiWordNet. Also SentiWordNet provides basic domain-free polarity for the feature along with a group of synonyms called synets [7]. Since these synets are words with same meaning as that of the feature, they too will have same orientation with respect to the given domain [19]. Hence the system also relate all the synets with the input component. This helps in expanding the domain knowledge of the system and thus increasing the performance of the system. It is found out that there can be multiple polarities and weights assigned to a word by SentiWordNet. This is because a word can have different orientations as per different domains. Hence SentiWordNet incorporates all the possible polarities for a word [7]. To find out the final polarity we have developed following algorithm.

// Algorithm to find the weight of opinion word from SentiWordNet

Step 1: Calculate the sum of all the positive polarities → p

Step 2: Find the average polarity weight → avg_p

Step 3: Calculate the sum of all the negative polarities → n

Step 4: Find the average polarity weight → avg_n

Step 5: The actual weight, w → $max(avg_p, avg_n)$

Step 6: If (w is equal to avg_p)then
Polarity → positive
Else
Polarity → negative

The expert polarity and SentiWordNet polarity are then cross-checked. If they differ then preference is given to the polarity given by expert as they have better understanding of the system. The style of writing a review varies from user to user. Also, amount of words in English Dictionary is very vast. Hence certain opinion words entered in the user review may not match the entry in domain knowledge. In such cases, the system does not neglect those words. Instead it finds out the general polarity of those words and stores them in the domain knowledge. Later the expert can refer the domain knowledge to check these entries, correct or confirm them and keep the domain knowledge consistent.

As shown in Fig., the system takes the components, features and polarity from the expert. After processing the input, analyzing and processing the final result is stored in the database. This includes component, feature, expert polarity, SentiWordNet polarity, the final weight assigned to the feature and finally all the synets related to the feature. This is shown in Fig. 2

Select the Component	Enter the feature	Select the Polarity
display	sharp	Positive
color quality	good	Positive
color quality	dull	Negative
sound quality	audible	Positive
sound quality	irritative	Negative
operating	easy	Positive

SUBMIT

Fig. 1. Interface for Domain Knowledge Input

Feature No.	Component Name	Feature Name	Average Weight	Polarity by Expert	Polarity by Senti Word Net	Synets
1	display	sharp	0.143	Positive	Positive	shrewd sharp astute crisp precipitous abrupt shrill piercing penetrative penetrating knifelike keen# tart

Fig. 2: Result Analysis after processing the input data

B. Data Extraction

The user reviews are obtained through blogs, websites, forums, etc. Many companies also give provision for signed-up users to enter reviews, comments and feedbacks for their products. For our work, we have targeted these reviews which are readily available to the companies. The extraction of reviews present in blogs, forums requires customized crawlers which will go through the suggested site and extract the reviews for the system. However, we have limited our work only on the reviews available with the company. Having said this, one can easily extend the system by integrating the customized crawlers so as to include the reviews present on blogs, etc. [1]

C. Sentence Analysis and Opinion Mining

The main function of Sentence Analysis is to find out the individual part of speech for each word of the sentence. This is done using the Stanford CoreNLP libraries [18]. On passing a sentence to these libraries a table of part-of-speech tags is obtained for every word in the sentence. Along with this POS Tagging, a set of dependencies is obtained along with word position of every word in the sentence [18]. The dependencies provide a basic idea how a group of words are related to each other. The word position information is helpful in determining the position of a given word. POS Tagging combined with dependency relation and word count can be referred as Tokenization process[18]. The aim of tokenization process is to find out the opinion words, modifiers and components from the sentence.

Generally, for a sentence, opinions can be determined by analyzing the adjective, adverbial parts of the sentence whereas the subject upon which opinion is exercised can be easily determined by the noun words in the sentence. The result obtained from Tokenization process is then passed to the Opinion Extraction Module. The opinion extraction module then finds out the relation between the component and feature using the HMM concept and creates the pairs based upon this relation. The system handles every type of relation along with negation. The opinion extraction process is describe in detail with a flowchart in Fig. 3.

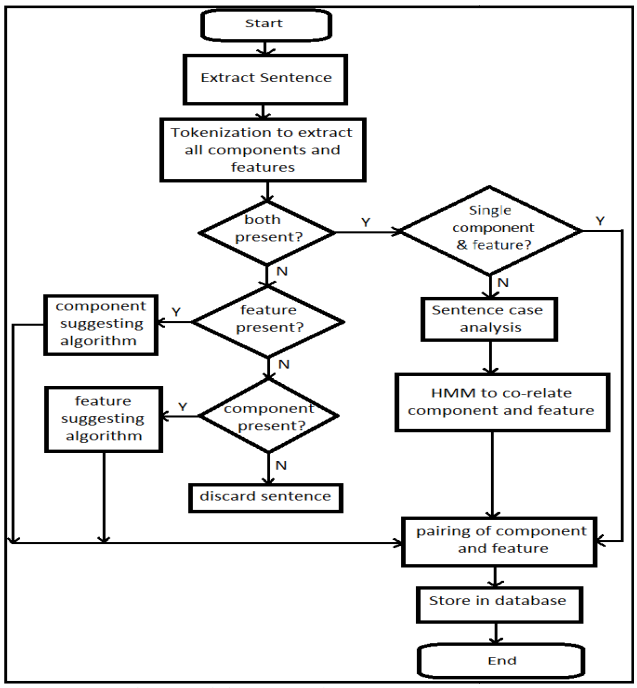

Fig. 3. Opinion Extraction Process

D. Weight Assignment

The next step after storing the component-feature (opinion words) pair is to assign weights and polarity to them. The stored pairs are sent to domain knowledge for determining the domain specific orientation of the feature along with the weights assigned to the features. Firstly, it is checked whether an entry for a given pair exist in the domain knowledge or not. If it is present, then the respective polarity and weights are retrieved from the domain knowledge and are assigned to the pair. However, if the entry is not present, the system does not discard the pair. Instead general polarity and weights are obtained for a given feature using SentiWordNet [7][19] and are assigned to the pair. The weight calculation is done as discussed in the domain knowledge section of this paper. Along with the weight assignment process, a new entry of the given pair is made in domain knowledge. This system-input can be then re-checked by the expert.

Next, the weight and polarity of the modifiers are also obtained through SentiWordNet [7]. The modifiers, as stated earlier, are the words which either enhance or reduce the polarity of opinion words. Hence we need to calculate the combined polarity of the modifier and opinion words.

Let P(o) and P(m) be the polarity of the opinion words and modifiers respectively. Let W(o) and W(m) be their respective weights. Then the final weight W(f) can be calculated as

$$W(f) = P(o) * W(o) * [1 + W(m)]^{P(m)} \qquad (2)$$

Once the weight is obtained for every pair in the sentence, the final weight is calculated by taking the average of the weights of all the pairs of the sentence. Suppose there are 'n' component-feature pairs in a sentence having 'w_i' weights respectively. Then the final weight (W) of the sentence is calculated as follows:

$$W = \frac{\Sigma wi}{n} \qquad (3)$$

Based upon the final weight obtained, the polarity is obtained for the sentence. If the final weight is less than 0, the sentence polarity is negative. If the final weight is greater than 0, the polarity is positive.

Following is the basic system architecture as shown in Fig. 4.

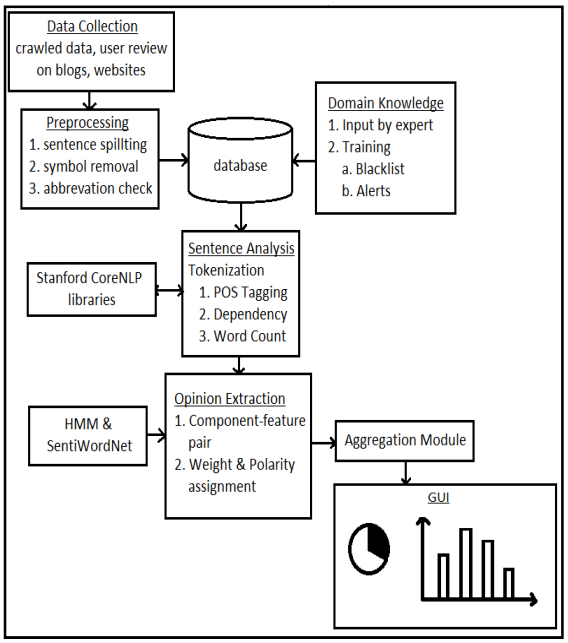

Fig. 4. System Architecture

IV. Results and Discussions

The system has a user input system through which reviews can be given as input. This input is first preprocessed to remove any unwanted symbols. For a review having multiple sentences, then sentence splitting is done. Then every sentence goes through Sentence Analysis and Opinion Extraction Modules to and component-feature pairs are obtained. Then these pairs are sent through Weight Analysis Module in which weights are assigned to pairs as well as sentence. Finally based upon these weights the polarity is assigned. Input is shown in Fig. 5. The Result for a review having single sentence is shown in Fig. 6.

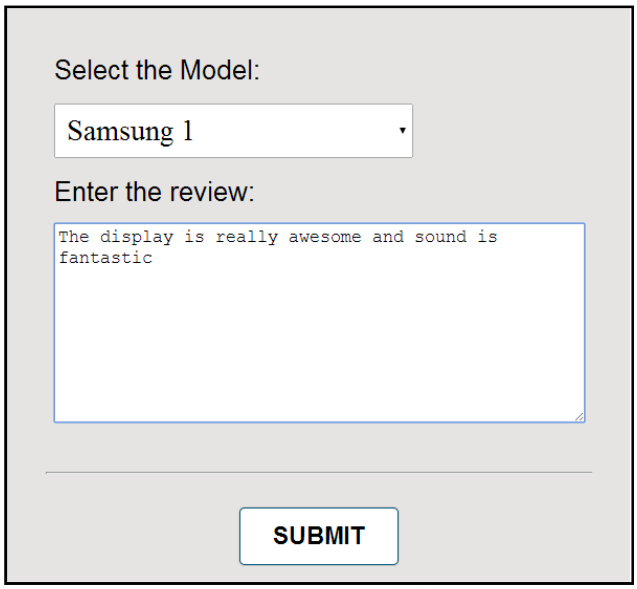

Fig. 5. User Input Frontend of the system

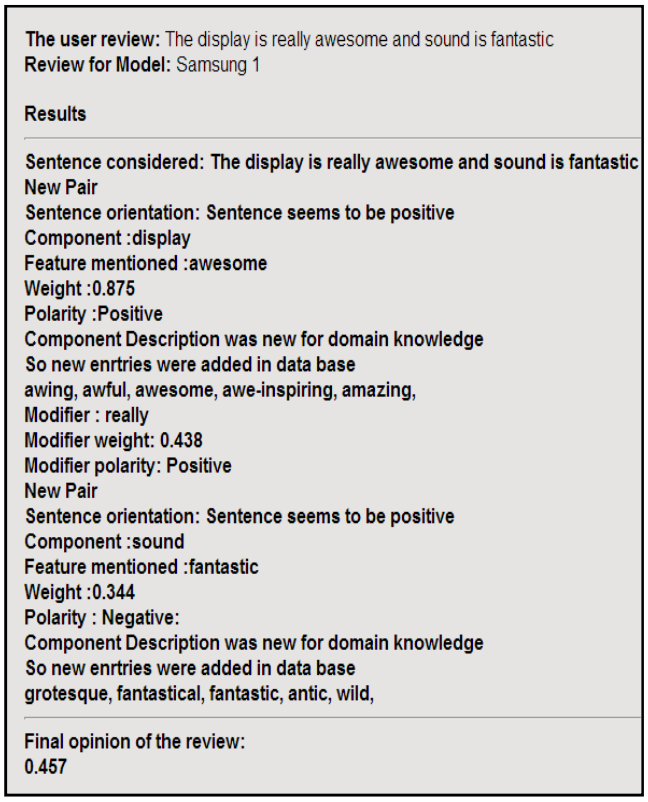

Fig. 6. The result obtained after processing and analyzing the sentence.

V. Further Development

The scope of the system can be extended in many ways. Firstly the data extraction process can be generalized to extract data for all the sources available on the Internet. This includes blogs, forums, websites and even chats. Also reviews sent through emails can also be integrated into the system. Secondly, working of the system can be made independent of the domain. This can be done by expanding the domain knowledge extensively to cover all the domain-related words for every field. Thirdly, the system can be designed for multiple languages so as to cover a greater area of user reviews. Finally, the system is not trained to detect sarcasm in the reviews. This can be done by training the system to detect phrases, words and senses which are generally used while giving sarcastic comments. All these can be incorporated into the system to yield better results.

References

[1] Lipika Dey, Sk. Mirajul Haque, “Opinion mining from noisy text data”, International Journal on Document Analysis and Recognition (IJDAR), Vol. 12, Issue 3, pp. 205-226, September 2009.

[2] Subhabrata Mukherjee, "Sentiment Analysis, A Literature Survey",Indian Institute of Technology, Bombay, India, 29 June 2012

[3] Amiya.T, Suman.R, Rylan.M, Sonu.P, Rilesh.R, "Extracting new product ideas from consumer blogs", International Conference on Communication, Information & Computing Technology(ICCICT), 2012.

[4] Ion Smeureanu, Cristian Bucur (2012), "Applying Supervised Opinion Mining Techniques on Online User Reviews", Informatica Economica Vol. 16, Issue 2, 2012.

[5] M. Rushdi Saleh, M.T. Martín-Valdivia, A. Montejo-Ráez, L.A. Ureña-López, "Experiments with SVM to classify opinions in different domains", Pergamon Vol. 38, Issue 12, pp. 14799-14804, 31 December 2011.

[6] A Agarwal, B Xie, Ilia Vovsha, O Rambow, R Passonneau, "Sentiment Analysis of Twitter Data", Proceedings of the Workshop on Language in Social Media (LSM 2011), pp. 30–38, Portland, Oregon, 23 June 2011.

[7] A Esuli, F Sebastiani, "SentiWordNet: A publicly available lexical resource for SentiWordNet", In Proceedings of the 5th Conference on Language Resources and Evaluation (LRE), 10 October 2006.

[8] Ross.S.M., "Probability and Random Process", Pearson Publication, 10th edition.

[9] Chuan Sheng Foo, "Hidden Markov Models: Decoding and Evaluation", Lecture Notes.

[10] Diana Maynard, Kalina Bontcheva, Dominic Rout, "Challenges in developing opinion mining tools for social media".

[11] Wei Jin, Hung Hay Ho, "A Novel Lexicalized HMM based Learning Framework for Web Opinion Mining", Proceedings of the 26th International Conference on Machine Learning, Montreal, Canada, 2009.

[12] G. Vinodhini, R.M. Chandrasekaran, "Sentiment Analysis and Opinion Mining: A Survey", International Journal of Advanced Research in Computer Science and Softare Engineering, 2012.

[13] Bo Pang, Lillian Lee, "Opinion Mining and Sentimental Analysis", 2008.

[14] Bing Lui, "Sentiment Analysis and Subjectivity", Handbook of Natural Language Processing, Second Edition, (editors: N. Indurkhya and F. J. Damerau), 2010.

[15] Veerarajan. T, "Probability, Statistics and Random Process", Tata McGraw Hill, 2002.

[16] Kavita Ganesan, Hun Duk Kim, "Opinion Mining Tutorial", University of Illinios.

[17] http://www.slideshare.net/KavitaGanesan/opinionminingkavitahyunduk00 [Date of Access: 5th Oct 2013].

[18] WordNetWikipedia, http://en.wikipedia.org/wiki/WordNet [Date of Access: 21st Sept, 2013].

[19] Stanford CoreNLP, http://nlp.stanford.edu/ [Date of Access: 26th March, 2014].

[20] SentiWordNet, http://sentiwordnet.isti.cnr.it/ [Date of Access: 15th March, 2014].