Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 35 (2014) 166 – 175

Procedia
Computer Science

18th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems - KES2014

Aspect based summarization of context dependent opinion words

Hitesh Kansal*, Durga Toshniwal

Dept. of Computer Science and Engineering, IIT Roorkee-247667, India

Abstract

Popularity and availability of opinion-rich resources in e-commerce platform is growing rapidly. Before buying any product, one is interested to know the opinion of other people about that product. For any product, there are hundreds of reviews available online so it becomes very difficult for the customers to read all the reviews. Also, one cannot set his mind based on reading some of the review since it gives him a biased view about that product. So we need to automate this process. As we know, there are lots of opinion words present in the sentences of a review which will tell about the polarity of that product. Out of all the opinion words, some words behave in the same manner means they have the same polarity in all contexts, but some words are context dependent means they have different polarity in different context. In this paper, we proposed an Aspect Based Sentiment Analysis and Summarization (ASAS) System, which handles the context dependent opinion words that has been the cause of major difficulties. For finding the opinion polarity, first, we used an online dictionary for classifying the context independent opinion word. Second, we used natural linguistic rules for assigning the polarity to maximum possible context dependent words. These steps create the training data set. Third, for classification of the remaining opinion words, we used opinion words and feature together rather than opinion words alone, because the same opinion word can have different polarity in the same domain. Then we used our Interaction Information method to classify the feature-opinion pairs. Fourth, as negation plays a very crucial role, we found negation words and flipped the polarity of the corresponding opinion word. Finally, after classifying each opinion word, the system generated a short summary for that particular product based on each feature

Peer-review under responsibility of KES International.

Keywords: Opinion Mining; Text Summarization; Sentiment Analysis; Context Dependent Opinions; Feature Based Clustering

* Corresponding author.
E-mail address: hk1990tohana@gmail.com

Peer-review under responsibility of KES International.
doi:10.1016/j.procs.2014.08.096

1. Introduction

The contents on the internet are increasing rapidly day by day and this content is created for different purposes by different users. One type of content is *reviews*, written by different people about different products. All the e-commerce sites enable their customers to write reviews about each product they buy, so that they can enhance the customer shopping experience. So more and more people write their experiences about the product they buy and it is very common for a person to read the other people's review before buying any product. As there are lots of reviews and some review can also be very long, so it is difficult for a person to read all the reviews. Also, one cannot set his mind by just reading some of the top reviews. As the number of reviews is increasing, it is also not easy for the manufacturer to keep track of customers' sentiments. So it becomes very important to produce a short summary of all reviews. In this regard, *Opinion Mining* or *Sentiment Analysis* analyzes customers' opinion on a specific product and tells whether that product has positive, negative or neutral orientation.

As there are a lot of opinion words, some of them have the same orientation in any context, e.g., "amazing", "excellent", "bad" etc. and some of the words are context dependent means they have different meaning when used in different context. For example, "The phone has *long* battery life" and "This program takes *long* time to run". In the above e.g., the word "long" has both positive and negative opinions in the first and second sentence respectively. There is probably no way to find the opinion of context dependent words just by knowing only that word and its associated features, without having the prior knowledge about that product. As there are a huge number of products, one cannot have such knowledge of every product. There are lots of research have been done in the past [1,2,3,4,5], but they all have some limitations which we will discuss in the next section. In this paper, we are proposing an effective approach for classifying the context dependent words.

In this paper, we propose an Aspect Based Sentiment Analysis and Summarization (ASAS) System, which does not look at the single sentence alone, it uses information from other sentences, other reviews, called contextual information, i.e., polarity assigned to same feature and opinion pair in other reviews. It also uses some linguistic rule of natural language to create a data set that will help to understand the product. This is a unique approach for creating a domain independent training data set of context dependent opinion words and that training data is used to classify the remaining context dependent opinion words by using our *Interaction Information* method described in this paper.

We are also handling some special words or word phrases that can change the orientation and must be carefully handled like "but also", "no wonder", "not just" *etc.*

For the text summarization, we are not using any template as in [6]. We are producing a structured summary based on positive and negative opinion about features of the product.

We have evaluated our technique by crawling review from amazon.com which contains a large number of products review and results show that our method perform effectively and outperform the previous ones.

The rest of the paper is organized as follows, section 2 discusses the related work, section 3 discusses our proposed framework in detail, section 4 discusses the experimental results of the proposed method and at last, section 5 presents the conclusion of the paper.

2. Related work

Sentiment classification of the documents is a very important research area and lots of research has been done in the past. Sentiment analysis can be done at three levels: document level, sentence level and feature level [7]. Document level analysis provides information about each review whether it is positive or negative. Sentence level analysis provides information about each sentence of a review. But the drawbacks are that both provide a high level classification. For example, a sentence can have multiple features with both labels, positive as well as negative. So we don't get information about each feature, i.e., which is positive and which is negative. For knowing the polarity of each feature, feature level analysis comes into the picture. It tells about each feature, whether it is positive or negative, and it is very helpful in exploring the product more effectively.

One method for sentiment classification is a supervised approach, where the training data set is used for the classification. Pang et al[8], compares the Naive based, SVM and Max Entropy approaches on unigram, bigram, a combination of both with the frequency of the features or presence in review. And SVM using feature presence with unigram performs better than others. But the drawback with supervised approaches is that they depend on large training data, which is time consuming to collect for each domain. Another approach for classification is unsupervised, which does not require any training data set and uses the sentiment lexicon for classification, like Word Net [9], SentiWordNet [10]. They give the sentiment score in the range from -1 to 1. But the drawbacks with unsupervised approaches are, they do not classify the context dependent opinion words correctly.

So to classify a context dependent opinion word, we have to utilize the context of the other reviews on the same product, i.e., polarity of that word in the other reviews, and for that, various techniques have been proposed like ΔTF-IDF[3], PMI[4], RMI[11]. But they all have certain drawbacks. In ΔTF-IDF, it only considers the total number of document that contains the term. It doesn't consider the distribution of the terms across various documents for classification. For example, we have 3 documents, one is positive and other two are negative. The frequency of the term in the positive document is high and it present only once in the negative document. The term should have positive polarity but if we use formula mentioned in [3], ΔTF-IDF comes out to be zero. So we cannot use this method for classification. PMI [4] works on the distribution of the terms across the document, but for classification, it only considers the opinion word for finding the polarity, i.e. it assumes a context dependent word has the same polarity across the same domain. But an opinion word can have different polarity in the same domain. For example, "Screen Resolution of this camera is low." and "This camera is very impressive because of its low price". The word "low" has a different polarity, although used in the same domain, i.e., Camera. So using only opinion words are not sufficient, we must consider its associated feature too. RMI [11] eliminates this problem by considering both feature and opinion word combined, to calculate the polarity by using conditional mutual information (CMI). Our work is closely related to [11], but [11] has some drawbacks. Firstly, it uses training data set, i.e. it becomes domain dependent and second, the training data used is labelled at sentence level, i.e. each sentence has one label associated with it. So if a sentence contains multiple features, some are positive and some are negative, this method applies the same label to all the features of that sentence, as training data are sentence level labeled and results in wrong classification.

So we eliminate these shortcomings by, firstly, automatically constructing the training data according to a domain which makes it domain independent and secondly, interaction information [12] is used to calculate the polarity of opinion words at aspect level rather than at sentence level.

3. Proposed technique

In this paper, we propose an ASAS system for handling context dependent opinion words. In this system, we are proposing an *Interaction Information* method for finding the polarity of the context dependent opinion words. As we discussed earlier, this system considers the features and opinion word, both as a deciding factor for polarity detection different from [4], which only considered the opinion word for polarity detection. But there are certain words which have a different opinion in the same domain. For example, "Screen resolution of the camera is low" and "This camera is very impressive because of low price". Here the word "low" has both negative and positive polarities respectively, even it is used in the same domain. So it is necessary to use both features as well as opinion word for finding the polarity. We divide our process into three parts:

- Feature Set Construction
- Opinion Analysis
- Summarizing using Aspect Based Clustering.

The flow diagram of the proposed system is shown in Fig 1. and red colour rectangle shows major contribution of our paper. We will explain each part of the system in details in the following subsections.

3.1. Feature Set Construction

After getting the review database, the first step we do is to handle the misspelled words. For that, we have a huge dictionary of commonly misspelled words obtained from [13]. After handling misspelled words, we tag the words in sentences using a Stanford POS tagger [14]. In POS tagging, words in the sentence are tagged with their respective part of speech and an XML file containing tagged sentences is generated.

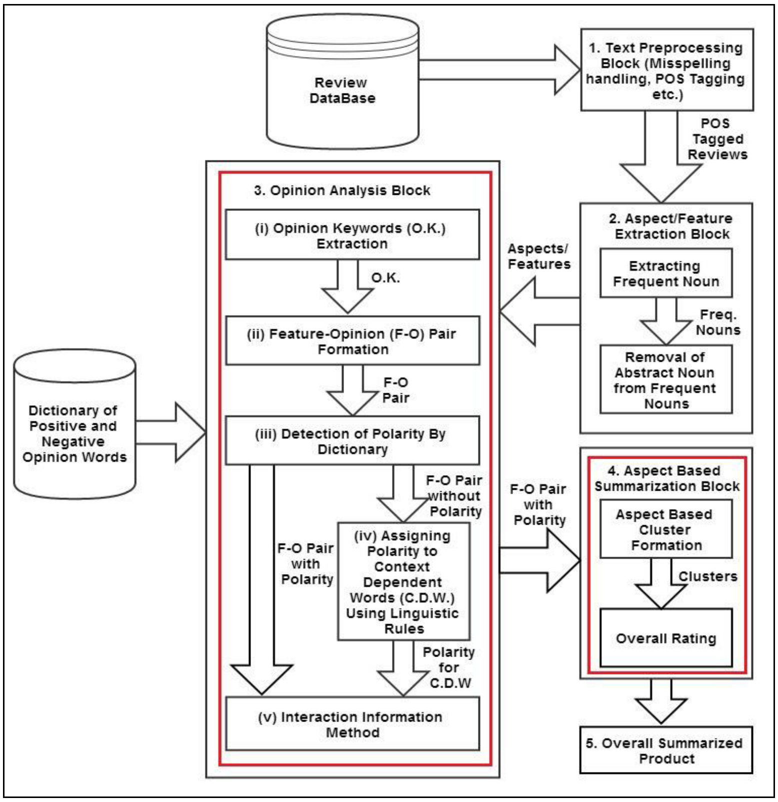

Fig. 1. Flow Diagram of Proposed System

After pre-processing of data, the next task is to identify the feature of the product. In this process, we extract not only the generic feature, but also the domain specific feature. Earlier research indicates that the features of the products are usually represented by nouns [5]. So, the tagged sentences which we got from the previous step are analyzed and respective nouns from each sentence are extracted. The nouns or noun phrases that are domain synonyms are clustered under the same group using semi-supervised approach used in [15].

The important thing to understand is that almost all the features are nouns, but not all nouns are features. For example: "I have several problems with this camera". Here "problems" is a noun, but it is not a feature of the product. So we need to eliminate these kind of non-feature nouns and for that, we are using the combination of association rule mining and probabilistic model [16]. We are using association rule mining because most users will mention the majority of the features. So most frequent nouns are extracted, but the drawbacks with this approach are that some nouns can still occur frequently. For e.g. some frequently occurring nouns are "Comment", or "Problem" etc.

To remove these frequently occurring nouns that are not features, we use the probabilistic model [16]. The basic rationale is that each product has its particular language, i.e. nouns or noun phrases representing features. For example: Camera has a lens, zoom, mega pixel etc. as its language. So the probability of the presence of noun representing features is more in that product than in any other product of different domains. In the above e.g. “Comment” which does not represent the feature has almost a similar probability of occurring in all the documents. So it's probabilities difference, from (1), will be less than theta (Θ) and it will be discarded. Value of Θ is determined experimentally.

$$prob(n \in C) - prob(n \in G) \geq \Theta \tag{1}$$

$$prob(n \in C) = \frac{|n \text{ in } C|}{|w \text{ in } C|} \tag{2}$$

$$prob(n \in G) = \frac{|n \text{ in } G|}{|w \text{ in } G|} \tag{3}$$

where *n* represents frequent noun, *w* represents total words present in corpus, *C* and *G* represents specific user product corpus and generic user product corpus respectively.

3.2. Opinion Analysis

The main focus of the paper is on opinion analysis, mainly opinion analysis of context dependent words. Opinion analysis is carried out in steps described below:

- Extraction of Opinion Keywords.
- Formation of Feature-Opinion Pair.
- Detection of Polarity of each Opinion Word.

In this process, we identify the subjective words, i.e., words that express opinions, and most of the subjective opinions are expressed by adjectives and verbs in sentences. So, for that purpose, we extract all the adjectives and verbs from the tagged sentences. After extracting all the opinion words, we form the feature-opinion pair. We map each opinion word to the nearest feature noun and form the pair.

After forming the feature-opinion pair, we find the polarity of each opinion word by using opinion lexicon [5], an online dictionary, containing the collection of positive and negative opinion words. However, user review can contain some context dependent opinion words whose polarities depend on the context, in which they are used. For example, "The battery life of this phone is long" and "This program takes long time to execute". Here in the first sentence, "long” is positive and it is used in the cell phone domain, and in the second sentence, "long" carries a negative opinion and it is used in the software domain. So in the different domain, same opinion word can have different meaning.

For finding the polarity of context dependent opinion words, we use the linguistic rules like "and”, "but" and "however". We can predict the polarity of opinion words present before or after the conjunction using these rules, as in the case of "and", the polarity before and after the conjunction remains same but in the case of "but" and "however", the polarity reverses after conjunction. For example, "Picture Quality of this phone is excellent and battery life is awesome" and "Picture Quality of this phone is excellent, but battery life is bad" are valid sentences, but "Picture Quality of this phone is excellent, but battery life is awesome" and "Picture Quality of this phone is excellent and the battery life is bad" is invalid because in first sentence, "Picture Quality" and "battery life" both has positive polarity and the conjunction used is "but". And we know in case of "but" conjunction, there should be different polarity before and after conjunction but in this case, there is same polarity, i.e., positive, before and after conjunction. Similarly, in second sentence, "Picture Quality" and "battery life" has positive and negative polarity respectively and the conjunction used is "and". And we know in case of "and" conjunction, there should be same polarity before and after conjunction but in this case, there is different polarity before and after conjunction. So this type of sentences are against the linguistic rules that is why these sentences are invalid. So if we know the polarity of opinion word before the conjunction then we can predict the polarity after the conjunction and vice-versa. But there is some phrases which are "but-like" and contain the "but" word, but do not change the orientation after conjunction. For example, "but also" in the following sentence "I not only like camera of this phone but also its keypad". So we are also handling these phrases in our approach.

We also detect negation words [17,18] in the neighbourhood of the opinion word. If any negation word is found near the opinion word, then the polarity of the opinion word is reversed. Similar to the "but" clause, there are also some phrases that contain negation words, but do not change the orientation. For example, "Not just", "Not only", "No wonder", "No problem" etc. We are also handling these phrases in our approach.

One important observation from [19] says, as one review is written only by a single person, so it is more likely that a person will assign the same polarity to the same feature throughout the review although it may appear more than once in a review. So by this observation, we assign polarity to the opinion words that are paired with that feature.

Now the feature opinion pairs that have assigned the polarity will act as a training data set for the context dependent opinion words which remain unclassified.

As earlier, we stated that, for assigning the polarity to context dependent words, we cannot simply use opinion words alone. But [4] is using Pointwise Mutual Information (PMI) for determining the polarity of opinion words using (4) where it only considers opinion words for classification.

$$PMI(\mathrm{X},\mathrm{Y}) = \log\frac{p(x,y)}{p(x)p(y)} \tag{4}$$

where X is an opinion word and Y is a label. $p(x,y)$ represents the total number of times x and y occur together and $p(x)$, $p(y)$ represents the total occurrence of x and y respectively in the document.

However, we will have to use both feature and opinion words together to determine the polarity and a similar approach is used in [11], called Revised Mutual Information (RMI). It considers both feature and opinion together to find the polarity of context dependent words. But there are some drawbacks with this approach which we have discussed in the previous section. To remove the first drawback, we are collecting our training data set by using opinion lexicons and linguistic approach which is domain independent, i.e. it can be used with any kind of domain. The second drawback is, in their training data set, they have labelled the polarity for each sentence, i.e., polarity is assigned for whole sentence not for individual feature. For example, "This phone has incredible battery life, amazing camera but bad keypad". In [11], a "mixed" label is assigned to sentence because it has both positive and negative polarities, and as a whole, sentence has one label i.e. mixed. The opinion with "camera" and "battery life" is positive, but with "keypad" it is negative. So if we calculate the polarity by using RMI method, it uses the same label for all of the features, but we should use different polarity, i.e. positive with "camera" and "battery life", negative with "keypad". So to make it correct, we assign the polarity at the feature level. We are labelling each feature and opinion pair separately rather than one label for complete sentences. We are using an *Interaction Information* method which will find that how much an opinion word is related to a particular label. Our proposed approach uses the (5) for calculating the polarity of opinion words. We find contextual information (CI) for both positive and negative labels using (5) and whichever gives more relevant value, that label is assigned to related opinion word.

$$CI(\mathrm{X},\mathrm{Y},\mathrm{Z}) = \log \frac{p(x,y)p(y,z)p(x,z)}{p(x)p(y)p(z)p(x,y,z)} \tag{5}$$

where *X* is an opinion word, *Y* is a label and *Z* is a noun associated with that opinion word. *p(x,y,z)* represents the total number of times *x,y* and *z* occur together. Similarly, *p(x,z)* represents the total number of times *x* and *z* occur together, *p(x,y)* represents the total number of times *x* and *y* occur together, *p(y,z)* represents the total number of times *y* and *z* occur together and *p(x), p(y), p(z)* represents the total occurrence of x, y and z respectively in the document.

3.3. Summarizing using Aspect Based Clustering

In [6], the author presented a question-answer type system which uses opinion summery for answering, and for summary representation, the author proposed a "scenario templates". But our approach is not same. We are not using any template. In the summarization step, for each feature, we will form two clusters; one with positive reviews and second with negative reviews. After that, we extract the respective excerpts based on the feature opinion pair and their polarity and put it into their respective cluster. The benefit of putting excerpts into their cluster with their respective feature opinion pair is that we can easily get the excerpts based on feature opinion pair. After this part, we calculate the star rating of each feature by dividing all the positive review by total review represented by that feature. Here we consider that a feature has an important significance if it is frequently mentioned by the reviewer, either may be positive or negative. So each feature is assigned a particular weight according to its significance which is calculated based on total number of occurrences of that feature. And finally, total rating of the product is calculated by adding rating of the entire feature set considering weight of each feature divided by the total number of features.

4. Experimental results

An ASAS System helps to classify the context dependent opinion words effectively. Here, we examine the efficiency of our Interaction Information method against PMI method [4] and also check the scalability of our system. We conducted our experiment on three products: Apex AD-1500 DVD Player, Canon PowerShot SX510 Camera and Nokia C2-01.5 Phone crawled from amazon.com in March 2014 in English Language. Table 1 shows the details of the reviews crawled.

Table 1. Reviews details (Total number of reviews and total sentences in each review)

Product Name	Total Reviews	Total Sentences
Apex DVD Player	179	846
Canon Camera	176	660
Nokia Phone	162	596

The procedure for our implementation is as follows: Firstly, the features corresponding to each opinion word are extracted and then subjective words that are represented by adjectives and verbs are extracted. After that, each feature is mapped with corresponding opinion word. Then our method for creating the domain independent training data set is applied to these three review corpus. For validation, we had a team of 5 experts in this field and compare the results produces by our ASAS system with the actual polarity assigned by those 5 experts and then finding the accuracy. Table 2 shows the accuracy of our created training data set which is 92%, 94% and 95% for Canon Camera, Apex DVD Player and Nokia Phone respectively. For calculating accuracy of training data set, we are considering only that feature-opinion (F-O) pairs which have assigned some polarity and leaving that F-O pair which have assigned neutral polarity. F-O pair with neutral polarity will later be assigned polarity using Information Interaction method. Now this training data set is used to classify the remaining opinion words by using our Interaction Information method. Results shown in Table 3 tells that 84.6% of all the excerpts are correctly classified and 15.4% are wrongly classified by our method for Nokia Phone. Whereas using PMI, only 82.3% are correctly classified. The reason for improvement in accuracy is because of the use of both feature and opinion word combined for determining the polarity of opinion words while PMI uses only opinion word for classification, ignoring the fact that the same opinion word can have different polarity in the same domain.

Table 2. Accuracy of Training Data Set Construction

Product Name	Accuracy
Apex DVD Player	94%
Canon Camera	92%
Nokia Phone	95%

Table 3. Accuracy of Opinion Polarity Detection

Product Name	PMI	Interaction Information
Apex DVD Player	78.3%	79.1%
Canon Camera	79.4%	81.3%
Nokia Phone	82.3%	84.6%

Also, to calculate the contextual information, we are considering polarity for each feature in the training data set rather than polarity for complete sentences. The results for Apex DVD Player, Canon Camera and Nokia Phone, by using Interaction Information and PMI, are shown in the Table 3. Fig 2 and Fig 3 is a graphical representation of accuracy of the training data set construction and comparison of accuracy of both Interaction Information method and PMI method for sentiment classification, respectively, which shows that our method performs efficiently.

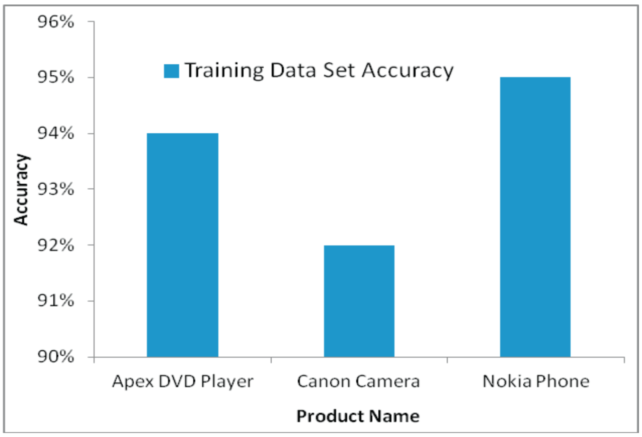

Fig. 2. Training Data Set Accuracy

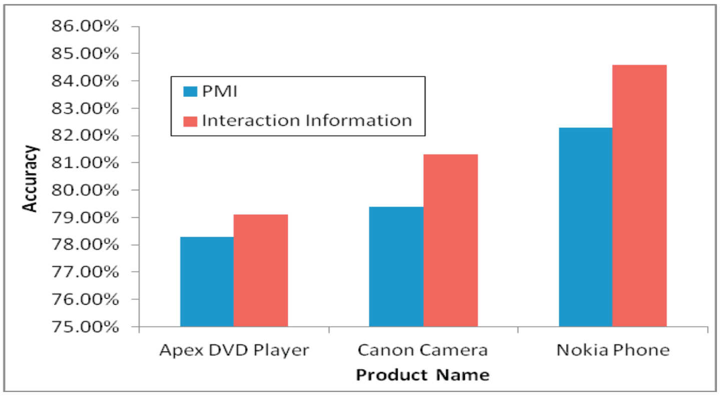

Fig. 3. Opinion Polarity Detection Accuracy

4.1. Scalability

In this section, We will show that our system is highly scalable, i.e., when we increased the number of sentences, our system's accuracy doesn't degrade. For scalability, firstly we run our system on small number of sentences then we add more number of sentences to it and check the accuracy and same process is repeated. In this process, we see that accuracy of our system remains almost same when more and more number of sentences is added. Figure 4 shows the accuracy variation of two products review. Values are shown in the form of (X,Y) where X is number of sentences and Y is accuracy of the system when X number of sentences are used.

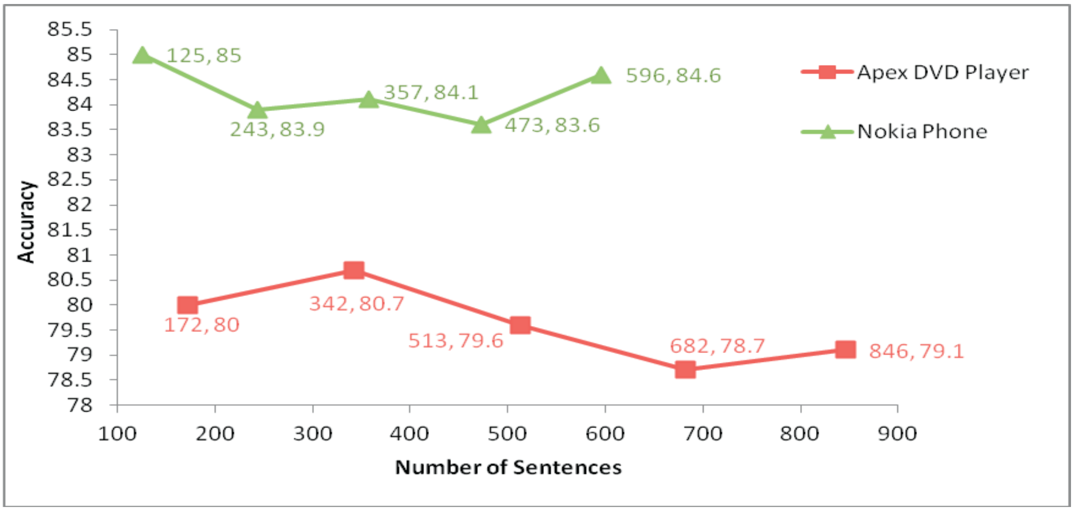

Fig. 4. Accuracy Variation of System for Different Number of Sentences

5. Conclusion and future work

In this paper, we presented an effective way for domain independent aspect based summarization of context dependent words, by using natural language tools and opinion mining techniques. Results show that our proposed approach, *Interaction Information*, performs much better than the previous approaches. As there are a good number of context dependent words present in the reviews, our technique helps customers to get an effective information present in large review corpus.

For future work, we will try to improve the accuracy of our system by making feature extraction more accurate and including more opinion words, i.e., nouns, adverbs, etc. As including nouns and adverbs will lead to increase more non-opinion words, so, we will also try to create an effective mechanism that will select only opinionated words from nouns and adverbs.

References

1. Ding, Xiaowen, Bing Liu, and Philip S. Yu. "A holistic lexicon-based approach to opinion mining." In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pp. 231-240. ACM, 2008.
2. Wu, Chunxu, Lingfeng Shen, and Xuan Wang. "A New Method of Using Contextual Information to Infer the Semantic Orientations of Context Dependent Opinions." In *Artificial Intelligence and Computational Intelligence, 2009. AICI'09. International Conference on*, vol. 4, pp. 274-278. IEEE, 2009.
3. Martineau, Justin, and Tim Finin. "Delta TFIDF: An Improved Feature Space for Sentiment Analysis." In *ICWSM*. 2009.
4. Lin, Yuming, Jingwei Zhang, Xiaoling Wang, and Aoying Zhou. "An information theoretic approach to sentiment polarity classification." In *Proceedings of the 2nd Joint WICOW/AIRWeb Workshop on Web Quality*, pp. 35-40. ACM, 2012.
5. Hu, Minqing, and Bing Liu. "Mining and summarizing customer reviews." In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 168-177. ACM, 2004.
6. Cardie, Claire, Janyce Wiebe, Theresa Wilson, and Diane Litman. "Low-level annotations and summary representations of opinions for multi-perspective question answering." *New Directions in Question Answering* (2004).
7. Liu, Bing. "Sentiment analysis and opinion mining." *Synthesis Lectures on Human Language Technologies* 5, no. 1 (2012): 1-167.
8. Pang, Bo, Lillian Lee, and Shivakumar 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*, pp. 79-86. Association for Computational Linguistics, 2002.
9. Miller, George A., Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. "Introduction to wordnet: An on-line lexical database*."*International journal of lexicography* 3, no. 4 (1990): 235-244.
10. Baccianella, Stefano, Andrea Esuli, and Fabrizio Sebastiani. "SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining." In *LREC*, vol. 10, pp. 2200-2204. 2010.
11. Saad, Farag, and Brigitte Mathiak. "Revised mutual information approach for german text sentiment classification." In *Proceedings of the 22nd international conference on World Wide Web companion*, pp. 579-586. International World Wide Web Conferences Steering Committee, 2013.
12. Van de Cruys, Tim. "Two multivariate generalizations of pointwise mutual information." In *Proceedings of the Workshop on Distributional Semantics and Compositionality*, pp. 16-20. Association for Computational Linguistics, 2011.
13. http://introcs.cs.princeton.edu/java/44st/misspellings.txt
14. http://nlp.stanford.edu/software/tagger.shtml
15. Zhai, Zhongwu, Bing Liu, Hua Xu, and Peifa Jia. "Clustering product features for opinion mining." In *Proceedings of the fourth ACM international conference on Web search and data mining*, pp. 347-354. ACM, 2011.
16. Bafna, Kushal, and Durga Toshniwal. "Feature based Summarization of Customers' Reviews of Online Products." *Procedia Computer Science* 22 (2013): 142-151.
17. Jia, Lifeng, Clement Yu, and Weiyi Meng. "The effect of negation on sentiment analysis and retrieval effectiveness." In *Proceedings of the* 18th *ACM conference on Information and knowledge management*, pp. 1827-1830. ACM, 2009.
18. Na, Jin-Cheon, Christopher Khoo, and Paul Horng Jyh Wu. "Use of negation phrases in automatic sentiment classification of product reviews." *Library Collections, Acquisitions, and Technical Services* 29, no. 2 (2005): 180-191
19. Qiu, Guang, Bing Liu, Jiajun Bu, and Chun Chen. "Opinion word expansion and target extraction through double propagation." *Computational linguistics* 37, no. 1 (2011): 9-27.