CSCI 6509: Advanced Topics in Natural Language Processing

Opinion Mining for Online Product Reviews

Submitted by:
Hardik Dalal (B00696939)
Chahna Dixit (B00695383)
Utsav Patel (B00691151)

Submitted to:
Dr. Valdo Keselj
Professor of Computer Science
Faculty of Computer Science

Abstract

Since the advent of Internet there is consistent increase in online shopping. More and more textual data representing reviews are available. These reviews contain valuable sentiments and opinions of product that is useful for other buyers. However, due to the large number of reviews and the complexity in interpreting them, it is difficult for the buyers to come to a conclusion regarding the product. Hence, it is necessary to have a system that summarizes the information in a meaningful and a concise form. The solution involves language processing and data mining techniques. The report focuses on the most reliable approaches to extract the information like lexicon-based methods and machine learning algorithms. Both approaches have their own pros and cons, and the evaluation of both approaches are depicted.

Table of Contents

Abstract i
1. Introduction 1
2. Related Work 3
3. Problem Definition and Methodology 4
4. Experimental Results 7
5. Conclusion 8
References 9

1. Introduction

The advancement of Internet and E-commerce has encouraged people to buy almost every products online. The buyers, besides buying products, also provide their valuable reviews, which often describe the quality of product or its features. Since thousands of reviews are posted every day for even a single product, it becomes difficult to go through all the reviews at once. Hence a summary of the reviews can give better insight to other buyers as well as the manufacturers. This can be achieved by mining of customer's opinions (aka opinion mining). Opinion mining, also known as sentiment analysis is a field of study that analyses people's opinions, sentiments and emotions towards products, services and other such entities [1]. Sentiment analysis and opinion mining mainly focuses on opinions which express positive or negative sentiments.
The research problem of sentiment analysis can be based on the level of granularities [2, 3]:

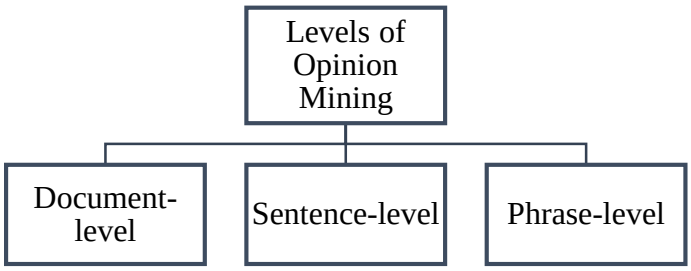

Figure 1. Levels of Opinion Mining

At the highest level of coarseness, a document as an entity is considered for determining the opinion. The document can be an individual review and processing such a review will determine whether the writer have positive or negative sentiments. At finer level, sentence level analysis comes into picture. This level generally include determining whether a give sentence is objective or subjective. The task of determining whether a sentence is subjective or objective is called subjectivity classification. An objective sentence is often a factual statement made by writer to present information. Whereas a subjective sentence generally express some kind of opinion or sentiment and hence it is more explored in our report. It is to be noted that there is not clear distinction between the two of types of sentences, as some of objective sentence may express opinion and vice versa. Entity or aspect level is the finest level of sentiment analysis. This involves determining the opinion directly or in other words we try to determine the precise opinion of the aspect. Moreover this level is more explored in our report as a research problem due to the idea that every opinion have two components: sentiment (positive or negative) and a target (the aspect or feature). Besides due to valuable information available through aspect level analysis, more and more research is carried out. For example, if we consider a review like "even though the battery is not good, I still like this phone", then we know that the review is positive as a whole sentence. However, it also indicate that the battery (aspect) of the phone have negative sentiment.

Such an aspect-based sentiment analysis consists of six main tasks [1]. First, the entity is extracted and categorized. Second, all the aspects of the entity are extracted and categorized. Third, the opinion holder is extracted and categorized, where the opinion holder could be a

person or any organization holding the opinion. Fourth, the times when the opinions were given are extracted and the time formats are standardized. Fifth, the opinion is categorized as positive, negative or neutral. In the end, an opinion quintuple is generated based on the results from previous tasks.

It is interesting to discuss that the opinions can be categorized into two types, namely regular and comparative. Regular opinions are further sub-divided into direct and indirect category. Direct opinions are the most common and easiest to handle. It expresses sentiment about a feature in straight forward manner. For example, "the camera is good". Whereas an indirect opinion expresses sentiment in ambiguous manner which makes it difficult to identify the target feature. For instance consider a review like "after using it for an hour, my hand hurts". Here the negative sentiment is expressed by criticizing the ergonomics of phone. The feature is ergonomics which is quite not clearly stated in the sentence.
Comparative opinions are comparison of aspects or products with the use of comparative and superlative form of adjectives. This category of opinions is less frequent however they hold valuable information about competing features or products. For example, "Apple is better than Samsung" express comparative sentiment of the two competing products.

It is important to take into consideration that sentiment analysis is a pure NLP problem. This includes but not restricted by co-reference resolution, ambiguity and handling negative words. Sentiments found in blogs, news, tweets and forum are the most complex. Moreover the sentiments and opinions expressed by writer are subjective which are highly unstructured. The task to convert such information into a structured form is very complicated problem. However, as we have restricted our research to online reviews which have very less irrelevant information it is possible to achieve better results compare to other textual corpuses. It is important to note that the analysis is restricted to lower levels of language processing hence it makes easier for researchers to achieve tangible results.

2. Related Work

There have been lot of research work done so far in opinion mining with some of the work are commendable. We have tried to cover few research papers in opinion mining as well as aspect based analysis.

L. V. Avanco et al. [4] have considered lexicon-based approach for sentiment classification of product reviews in Brazilian Portuguese. They have built a lexicon-based classifier which considers prior polarity of words and some knowledge of contextual valence shifting (negation and intensification). The prior polarities are defined by three sentiment lexicons: OpinionLexicon, SentiLex and subset of LIWC (Linguistic Inquiry and Word Count). The lexicon-based classifier uses three different versions: LBC-p, considers prior polarity of sentiment words; LBC-pn, considers negation contexts too; LBC-pni, considers intensification contexts too. The evaluation was done on the dataset as rated by the writers and also as rated by independent readers. The results obtained favored the version which considers negation and intensification along with SentiLex. The method used here could be further improved considering the reviews that contain both - negative and positive sentiments.

Jurgen Bros and Heiko Ehrig [5] proposed a method for aspect based product review mining where in the reviews that they had chosen, consisted of pros and cons. The list of product aspects and reviews with pros and cons were given as input to the proposed method, and the output obtained was context-aware sentiment lexicon. Finally, the sentiment of the product aspect was considered to be positive if it occurred more in the pros section of the reviews; whereas it was considered to be negative if it occurred more in the cons section of the reviews. High accuracy was achieved with this method and that it also outperformed SentiWordNet.

Another example of feature extraction was provided by [6] in which they have mention that opinion words could be used for feature extraction. The procedure they proposed in the paper consists of three steps: Perform POS tagging, find opinion word and find nearest noun/noun phrase. Opinion words are the adjective words used in the sentence. To quickly find opinion words authors suggested to store the opinion words that are found and also store their synonyms and antonyms for future references. The noun/noun phrase found in the last step will be the feature of product that is described by opinion word. A problem with this procedure is that the noun/noun phrase found can be irrelevant to the given product. The reason is people use same adjective for lots of objects including relevant and irrelevant features. The frequency of irrelevant features will be very low compared to relevant features. To remove irrelevant features machine learning algorithm: association algorithm can be used to find the most frequent features. The frequent feature found after applying association algorithm will be the most likely relevant features.

3. Problem Definition and Methodology

After going through the research in this field, we found a gap between what has been done and what is possible with the wide set of online reviews. The report focuses specifically on the extraction of opinion from online product reviews and summarizes them in structured form. In opinion mining, there are two major approaches to extract sentiments: machine leaning algorithms and lexicon-based methods [7].

Machine learning algorithm: This method uses the renowned machine learning algorithm to find polarity of text. The algorithms are divided into two: Supervised and Unsupervised. Supervised algorithms are used in environment where the decision variable (here the sentiment polarity) is already known for training the model. However in unsupervised algorithms, the reviews are clustered as the model is trained. The commonly used algorithms are Naïve Bayes, Decision Trees, Support Vector machines and K-nearest neighbor. Feature extraction is important in training any machine learning algorithm. The common techniques used to extract features are N-gram (uni-, bi-, tri-). The accuracy achieved with this approach is high although it depends on the domain, dataset, feature extraction and many other factors.

Lexicon-based method: This method mainly relies on linguistic model and requires exhaustive knowledge base to determine opinion. In other words, this method is based on linguistic features such as nouns, adjectives, adverbs, etc. There are several corpuses freely available such as SentiWordNet, WordNet, etc. These sentiment lexicons are widely used in solving sentiment-based problems. The other technique commonly used is Part-of-Speech tagging. In POS tagging, individual lexicons are tagged with a category name, which in turn can be used to extract features or sentiments. It is to be noted that the accuracy achieved is lower compared to the other approach.

Most of the researchers use both the approaches together to achieve better accuracy. Hence we have also tried to evaluate both the approaches and they are explained below.

Methodology 1:

The process of extraction of opinions from reviews, involves several steps, starting with pre-processing the data. Preprocessing of data is cleaning or removing the parts of reviews that might adversely affect the results. It involves stop word removal, tokenization, stemming and lemmatization. We have used Amazon review data of cellphone and accessories [8, 9] for mining sentiments. The reviews are structured as follows: product/product ID, product/title, product/price, review/user ID, review/profile name, review/helpfulness, review/score, review/time, review/summary and review/text. However, based on the helpfulness of predicting the sentiments we picked product/product ID, review/time, review/summary and review/text for training purpose. Review/score is chosen as the target feature.

As a part of preprocessing, we have used stop words removal, tokenization and stemming. Tokenization is an important step in which the plain text is broken into words or tokens. Stop words removal is getting rid of non-informative words which occur very frequently [10]. They are normally articles, prepositions and conjunctions and sometimes verbs, adjectives and adverbs. Stemming is a technique in which the words are reduced to their grammatical roots [10]. It usually works by removing some suffixes according to a set of rules. We have used Porter stemmer for stemming purpose.

In our methodology to extract sentiments, we have considered adding new fields for the learning purpose. The new fields serve as features in training the algorithm that are number of words of

review/summary, number of words of review/text, number of lines of review/text, length of stemmed review/text, length of stemmed of review/summary, ratio of stemmed and original length review/text and ratio of stemmed and original length review/summary. The dataset consists of around 78,000 were divided into 75:25 ratio for training and testing purpose respectively. According to our data set and the extracted features, we used limited number of algorithms for classification purpose.

To start with, we used the most common approach - Naïve Bayes classification. Naïve Bayes algorithm is based on the Bayes theorem which works on the assumption of independence between every pair of features. The advantages of using Naïve Bayes algorithm is that it is based on the sound theory of conditional probability. Besides, it is simple and highly scalable. The accuracy achieved after training the model was close to 36 %.

We have also used regression based classification technique called Gradient Boosting. Gradient boosting algorithm produces strong model from many intermediate weak prediction models using decision trees. Our motivation to use the technique is based on the assumption that the features have tight correlation with each other. Moreover the last two features that states ratios of stemmed and original text is tightly coupled with the other features and ultimately with the sentiment. Moreover the algorithm is very robust to the problem of over fitting [11]. The large size of data set yielded accuracy of approximately 42%.

Apart from Naïve Bayes classification and Gradient Boosting classifier, we have also used Decision Tree classifier. Decision tree algorithm builds a decision tree that is used for predict the value of the target feature. The modelling process is based on the values of several input features used in the training phase. We have considered review/score as the target feature and the rest of the other features discussed above as input to the decision tree classifier. As known, the decision tree is very efficient for fewer numbers of features and since the number of features in our model is relatively fewer, the accuracy achieved is almost 99%.

The review/score is predicted after applying the classification algorithms. This review/score is an integer ranging from 1 to 5. After going through the review/summary, we assumed that positive reviews has review/score equal or greater than 3, while negative reviews have score below 3.

The implementation has been done using Python as the programming language and libraries – pandas, nltk, sklearn and numpy.

Methodology 2:

In order to find sentiment based on the linguistic knowledge, we have used a data base "knowledgebase" [12]. It consist of a table "wordVals", with fields "word" and "value". "word" field contains all the opinion words and the "value" field represent respective sentiment score. The sentiment score is 1 for positive, 0 for neutral and -1 for negative lexicons. Further we have chosen 10 reviews from the Amazon cellphone and accessories data set for this approach. The subset is manually labeled as positive or negative.

As part of preprocessing in this data set, we have used techniques like stop word removal and tokenization. After preprocessing the reviews have been marked using Part-of-Speech (POS) tagging. According to the linguistic knowledge, the tokens that are important for determining the sentiment are generally adverbs, adjectives and verbs. Considering the knowledge base we determined each of these tokens as positive or negative. The overall sentiment of the lexicons is calculated by adding a sentiment score of the individual tokens. Initially, we defined -1 as the sentiment score for the negative extracted tokens, and 1 for positive extracted tokens. However, on observing the results we found that majority of tokens were misclassified. Hence through trial

and error we decided a threshold value for the sentiment scores, which was -1.7 for negative and 0.52 for positive tokens. The final accuracy was nearly 60%.
This idea to classify the sentiments has been adopted from Python Opinion Mining and Sentiment Analysis Tutorial series by Harrison Kinsley [13].

4. Experimental Results

The different classification algorithms used for the first methodology have been evaluated using the most common measures, i.e. precision, recall, F-measure and accuracy. Precision is the percentage of true positives out of all returned documents. Recall is the percentage of true positives out of all relevant documents in the collection. F-measure is the weighted harmonic mean between precision and recall. Accuracy is the ration between the total number of opinions correctly classified and the total number of opinions given to the classifier.
Table 1 below shows the results obtained for the classifiers Naïve Bayes, Gradient Booster and Decision Tree.

	Precision (%)	***Recall (%)***	***F-measure (%)***	***Accuracy (%)***
Naïve Bayes	38.26	36.33	34.01	36.79
Gradient Booster	60.79	39.17	37.51	42.70
Decision Tree	99.99	99.99	99.99	99.99

Table 1. Experimental Results of first methodology

As the results show, the decision tree outperforms Naïve Bayes and Gradient Booster which is very obvious since decision tree is known to be very efficient with lesser number of input features. In our model, since we have used seven features, decision tree gives better results. If the number of features would be large, the decision tree would not have given such results.

On the other hand, while using lexicon-based method for sentiment analysis in the second methodology, 60% accuracy was achieved.

Hence, overall if you evaluate, lexicon-based methods give better results compared to machine learning algorithms. This is because in machine learning algorithms, the selection of algorithm is an important step. Although, in lexicon-based method, the approach chosen is important, if the steps are planned well, higher accuracy can be achieved. Sometimes, combination of both the methods can give better results rather than using either of them.

5. Conclusion

The report has addressed the research problem of summarizing large number of reviews. The problem of extracting the opinions from the review was carried out using the two possible approaches. The steps and accuracy is highly dependent on the data, preprocessing steps and feature extraction. The feature extraction is a complicated task, which requires in depth understanding of the data and the approach. In the end, we were able to classify the opinions based on the explicit sentiments in the reviews. However, due to time constraint we were not able to summarize the reviews in more structured form. We also felt that there is more scope of research in aspect based opinion mining. We are motivated to carry forward the research.

References

[1] Bing Liu, "Sentiment Analysis and Opinion Mining", Morgan & Claypool Publishers, May 2012.

[2] Samaneh Moghaddam, Martin Ester, "Aspect-based Opinion Mining from Online Reviews", *Special Interest Group on Information Retrieval*, August 12-16, 2012, Portland, Oregon, USA.

[3] Malik Muhammad Saad Missen, Mohand Boughanem, Guillaume Cabanac, "Opinion mining: reviewed from word to document level", *Social Network Analysis and Mining*, Vol. 3, Issue 1, pp. 107-125, March 25, 2012.

[4] Lucas V. Avanco, Maria G. V. Nunes, "Lexicon-based Sentiment Analysis for Reviews of Products in Brazilian Portuguese", *Brazilian Conference on Intelligent Systems*, pp. 277-281, October 18-22, 2014, Sao Paulo.

[5] Jurgen Bros, Heiko Ehri, "Generating a Context-Aware Sentiment Lexicon for Aspect-Based Product Review Mining", *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pp. 435-439, August 31, 2010 – September 3, 2010, Toronto, ON.

[6] Minqing Hu, Bing Liu, "Mining and Summarizing Customer Reviews", *10th ACM International Conference on Knowledge Discovery and Data Mining*, pp. 168-177, NY, USA, 2004.

[7] Emma Haddia, Xiaohui Liua, Yong Shib, "The Role of Text Pre-processing in Sentiment Analysis", *1st International Conference on Information Technology and Quantitative Management*, Vol. 17, pp. 26-32, 2013.

[8] http://snap.stanford.edu/data/web-Amazon-links.html

[9] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text", RecSys, 2013, Proceedings of the 7th ACM conference on Recommender systems, pp. 165-172, NY, USA, 2013.

[10] J. R. Méndez, E. L. Iglesias, F. Fdez-Riverola, F. Díaz, J. M. Corchado, "Tokenising, Stemming and Stopword Removal on Anti-spam Filtering Domain", 11th Conference of the Spanish Association for Artificial Intelligence, pp 449-458, November 16-18, 2005, Santiago de Compostela, Spain.

[11] "scikit learn sklearn.ensemble.GradientBoostingClassifier", http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html.

[12] http://pythonprogramming.net/downloads/knowledgeBase.db

[13] "Python Opinion Mining and Sentiment Analysis Tutorial", http://pythonprogramming.net/accuracy-testing-basic-nlp