Product Feature Discovery and Ranking for Sentiment Analysis from Online Reviews

Nitish Gupta[1], Shashwat Chandra[2]
Advisor: Dr. Amitabha Mukerjee[2]
{gnitish, chandras, amit}@iitk.ac.in
[1] *Dept. Of Electrical Engineering*
[2] *Dept. Of Computer Science and Engineering*

November 16, 2013

Abstract

Analysis of reviews and opinions, known as review or opinion mining, has attracted a great deal of attention recently due to many practical applications and challenging research problems. An important issue in review mining is to extract people's opinions and sentiments on different aspects or features of the product being talked about in the reviews. Here we propose a method that in a fully unsupervised manner would accomplish the task of discovering features, populating opinion words that are used to modify them and perform a popularity analysis of the features.

1 Introduction

An important task of review mining is to extract people's opinions and sentiments on features of a product. Eg. The sentence *'The phone has a good battery life.'* expresses positive review about the *'battery life'* of the *'phone'*. Here *'battery life'* is a feature of the product *'phone'*and the user expresses positive review about it. In an unsupervised environment, extraction of features is the most important and difficult aspect of review mining. Feature Ranking and Popularity Analysis on the extracted feature-candidates is important for a twofold reason, one to increase the precision of the top-ranked candidates, and secondly, to find those features which the users consider the most while reviewing a particular product. Populating the opinion words that mostly occur with a particular feature will help in giving an insight as what are the general remarks of the population about a particular feature of a product. It will help in careful sentiment analysis.

2 Related Work

The related work in the field mainly comes from researchers like Dr. Bing Liu, Guang Qiu etc. The related work in different aspects is as follow :

2.1 Extracting Product Class

As the product class of the reviews is not specified it is to be computed in an unsupervised manner. Bing Liu et. al.[2] proposed the *part-whole* relation to solve this problem. Part-Whole relations deal with sentences like *'The screen of the phone', 'the valley on the mattress'* which have an inherent dependency in them. The *phone, valley* resp. act as the 'part' and the whole (*phone, mattress*) is the actual product class. Dependencies like these can be used to extract the product being talked about in the reviews.

2.2 Feature Extraction

Extracting features in a fully unsupervised manner is one of the most difficult tasks of review mining. Guang Qiu et. al[1] proposed a novel method called the *Double-Propagation* based on bootstrap aggregation to extract features and opinion words modifying them using several syntactic relations that link opinion words and features.

2.3 Sentiment Analysis

Sentiment analysis on opinion documents and reviews mainly deal with finding people's opinions on various features of a product. It deals with recognizing the positive/negative opinions. Sentiment analysis proposed in [3] uses two-word phrases with compatible POS tage to achieve results. Semi-supervised analysis in [4] uses clustering of synonym opinion words to perform sentiment analysis on opinion documents.

3 Our Approach

We propose a fully unsupervised approach of finding the product being talked about in the reviews, discovering features of the product and the various opinion words used to modify the opinion words. We also perform popularity analysis on the features to predict a desirability of the features and also compute the most relevant features in a product according to their popularity.
We assume that the features of a product are Nouns & Noun Phrases and the opinion words used are Adjectives.

3.1 Predicting Product Class

We propose a very naive approach to find the product class being talked about in the reviews. We observe that the most occuring noun in the reviews is indeed the product being talked about. This naive approach has a very good accuracy over the part - whole relation proposed in [2]. Our approach is based on the heuristic that most of the reviews are mostly written in the form *'This phone..., The washing machine..., The new Hitachi router..., etc.'* which then implies that the most frequent noun in the list is expected to be the product being reviewed.

3.2 Discovering Features

As we know that all the features are Nouns / Nouns Phrases we POS tagged the whole review dataset and computed the frequency of different nouns in the reviews. Also we compared this list to the list of features and found out that more that 99.5% of the features are a part of our Noun List. This means that even this naive 'bag-of-words' approach has a very high recall but a very low precision of the order of 10% because it contains a lot of spurious nouns that are not features. We then try to exploit the high recall of this bag-of-words approach to prune our bag of words to remove words that are not features to increase the precision of our system. We also prepare a similar list of all adjectives to prune later to extract opinion words.

We then try to exploit the fact that there exists naturally occuring syntactic relations in the features and opinion words in reviews. For eg. In the sentence '*The phone has a good screen*' there exists a relation that the adjective *good* modifies the noun *screen.* From this we can extract that the *screen* is a feature and that *good* is an opinion word modifying the word *screen* in a positive manner. Relations like these can be used to discover new features and opinion words by just giving them starting seeds. These relations can be identified using a dependency parser based on dependency grammar and then exploited to perform the extraction tasks. to discover new features and opinion words.

We use modified dependency rules inspired from [1] to remove nouns and adjectives that are not features and opinion words respectively from the bag-of-words we populated in the previous step. The table below gives the dependency rules used to extract the features and opinion words using starting seeds. The dependency rules used are given in the table below.

RuleID	Observation	Extracted	Example
R11	O → amod → T s.t. O ∈ {O}	Feature = T	The phone has a good screen.
R12	O → amod → ‘Prod’ ← nsubj ← T s.t. O ∈ {O}	Feature = T	iPod is the best mp3 player
R31	T1 → conj → T2 OR T2 → conj → T1 s.t. T1 ∈ {F}	Feature = T2	Audio and video quality of the player.
R32	T1 → nsubj → ‘has’ ← dobj ← T2 s.t. T2 ∈ {F}	Feature = T2	Canon“G3” has a great lens.
R21	O → amod → T s.t. T ∈ {F}	Opinion = O	The phone has a goodscreen.
R22	O → amod → ‘Prod’ ← nsubj ← T s.t. T ∈ {F}	Opinion = O	iPodis the best mp3 player
R41	O1 → conj → O2 or O2 → conj → O1 s.t. O1 ∈ {O}	Opinion = O2	Camerais amazingand best.
R42	O1 → amod → ‘Prod’ ← amod ← O2 s.t. O1 ∈ {O}	Opinion = O2	Thesexy, cool mp3 player

Table 1: **Dependency Rules for Feature and Opinion Word Extraction**

In the above table {F} and {O} is the set of features and opinion words respectively either given as starting seeds or extracted during the running of the algorithm. The algorithm to prune our bag-of-words for possible features and opinions words is given below :

```
repeat
    $O_{New} \leftarrow \emptyset$
    $F_{New} \leftarrow \emptyset$
    for $f \in F_{List}$ do
        Look at every sentence with $f$
        if R11 is satisfied with $o \in O_{Seeds}$ then
            $F_{New} \leftarrow F_{New} \cup \{f\}$
        if R12 is satisfied with $o \in O_{Seeds}$ and product then
            $F_{New} \leftarrow F_{New} \cup \{f\}$
        if R31 is satisfied with $f' \in F_{Seeds}$ then
            $F_{New} \leftarrow F_{New} \cup \{f\}$
        if R32 is satisfied with $f' \in F_{Seeds}$ and “has” then
            $F_{New} \leftarrow F_{New} \cup \{f\}$
    for $o \in O_{List}$ do
        Look at every sentence with $o$
        if R21 is satisfied with $f \in F_{Seeds}$ then
            $O_{New} \leftarrow O_{New} \cup \{o\}$
        if R22 is satisfied with $f \in F_{Seeds}$ and product then
            $O_{New} \leftarrow O_{New} \cup \{o\}$
        if R41 is satisfied with $o' \in O_{Seeds}$ then
            $O_{New} \leftarrow O_{New} \cup \{o\}$
        if R42 is satisfied with product then
            $O_{New} \leftarrow O_{New} \cup \{o\}$
        $F_{Seeds} \leftarrow F_{Seeds} \cup F_{New}$
        $O_{Seeds} \leftarrow O_{Seeds} \cup O_{New}$
until $O_{New} = \emptyset$ and $F_{New} = \emptyset$;
```

Algorithm 1: The propagation algorithm

For the algorithm to start the extraction iterations we provide it with the 11 most occuring nouns as starting feature seeds and 11 most occuring adjectives as starting opinion words in the sets {F} and {O} respectively. At the end of the algorithm these sets grow to contain all the features and opinion words respectively.

This method is expected to be computationally less expensive than the *Double-Propagation* due to the fact that we only try to prune our bag-of-words list for non-features and non-opinon words rather than traversing through the whole corpus trying to find all words that satisfy the above given rules.

3.3 Feature Ranking and Popularity Analysis

The feature ranking and popularity analysis is important to gain insight as to what features the users consider important in a product and mostly talk about. The basic idea of feature ranking is that if a feature candidate is correct and frequently mentioned in the corpus, then it should be ranked high. Feature frequency is the occurence frequency of a feature in the review corpus which is easy to obtain. We can also maintain a list of opinion words which are generally used to modify a particlar feature to compute feature relevance. Relevant features are generally modified by various opinion words. This will also help in getting positive / negative reviews for different features of a product.

For ranking features, we arrange the extracted features in the decreasing order of their occurence frequency. We also maintain a list of opinion words used to modify them to get an insight on the polarity of user reviews about different features of a product.

4 Experiments & Results

We ran our method for product class extraction, feature and opinion words discovery and feature ranking and popularity analysis on various review datasets that we acquired from Dr. Bing Liu's webpage. The dataset contains reviews for 15 products each with around 300-400 reviews. Our results on various fronts are as follows :

4.1 Product Class Discovery

Our naive bag-of-words approach outperforms the part-whole relation specified in [2]. On the dataset we tested, our method was able to correctly predict the product being talked about in **92.85% cases** compared to only 71.42% cases using part-whole relation. The table below lists a few examples.

Product Name	**Part - Whole**	**Our Approach**
MicroMP3	I	Player
Nokia 6600	Phone	Phone
Norton	Computer	Norton
Hitachi Router	Table	Router

Table 2: **Product Class Prediction: Our Approach vs. Part - Whole Relations**

4.2 Feature Discovery

Our algorithm for feature discovery requires starting seed features and opinion words. We exploit the fact that the most freqent occuring nouns and adjectives are indeed mostly indeed features and opinion words respectively. We remove the most occuring noun as the product class and use the next few most frequent occuring nouns and adjectives as starting seeds. The table below shows some examples of starting seeds.

Product Name	**Starting Feature Seeds**	**Starting Opinion Word Seeds**
Nokia 6610	phones, nokia, features,service, radio	T-Mobile, great, good,other, small . . . . . .
Hitachi Router	table,Hitachi, speed, bit, bits	easy, good, great,other, easy . . .
Nokia 6600	Nokia, features, phones, camera, card	great, good, T-Mobile,other, best . . .
Canon S100	pictures, battery,card, cameras, screen	digital, great,smart, small, good . . . .
Creative MP3 Player	software, iPod, zen, mp3, battery	good,creative, easy, great, other . . .

Table 3: **Examples of Starting Feature and Opinion Word Seeds**

4.2.1 Feature & Opinion Word Extraction

We ran our algorithm and the double - propagation algorithm on various datasets. The results were evaluated by using the already hand-annotated features as groundtruth. We find that our method supercedes the double propagation algorithm in terms of the precision of our feature list. Also our F-Score as compared to the double-propagation does better most of the time with improvements of the order of 10%. The table below compares our results to that of the double-propagation.

Product Name	**Our Method**			**Double-Propagation**		
	Precision	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
Nokia 6610	71.88	28.75	**45.45**	19.4	65	35.51
Hitachi Router	66.67	14.63	**31.23**	23.3	50	26.95
Nokia 6600	58.54	17.02	31.56	23.21	46.1	32.71
Canon S100	54.55	13.95	27.58	28.74	29.07	28.90
Creative MP3 Player	38.1	31.37	**34.66**	15	68.63	32.33

Table 4: **Feature Discovery Results**

We hand-checked the list of extracted opinion words and found that our algorithm finds correct opinion words at a true-positive rate of 90%.

4.3 Feature Populatiry Analysis

Feature Popularity Analysis deals with discovering the most popular features according to the reviewers and also the opinion words associated with them to get an insight into the polarity of sentiments involved with each feature. The most relevant features and the opinion words associated with them are populated in the table below for various products.

Product		Popular Features & Opinion Words modifying them
Nokia 6610	1. battery : good, long,great 2. sound : great,nice, good, great	3. software : good,cool, nice, great 4. service : great, access, good
Hitachi Router	1. use : difficult, great, good 2. control : great,good,	3. weight : light, low, good 4. price : low, good, best
Nokia 6600	1. camera : best, great,cool 2. screen: good, big, color, huge	3. Bluetooth : slow, cool, great, good 4. size : good, big, small
Canon S100	1.picture: good , nice, cool 2. exposure : good,manual, better, cool	3. software : good,cool, nice, great 4. focus : great, better,sharp, manual
Creative Mp3Player	1. music : great,fast, perfect 2. software : good, bad,better, easier	3. sound: good, great, superior, best 4. screen : bright, light, good, perfect

5 Conclusion

Through this project we came to the conclusion that review mining is indeed important to extract features and know the users opinions on them. Our approach of finding the product class outperforms the ‘part-whole’ relation approach. The part-whole relation can be used to capture the semantic relatedness, but it is not very useful in actually finding the product class. Our approach of feature discovery has a much better precision over the state-of-the-art Double-Propagation algorithm. Our recall is not that good, but we think the precision in the list of features extracted is far more important because we would not like spurious features to smear our feature list. The feature popularity analysis really gives an insight to what the important features are and the users’ polarity towards them.

6 Future Work

- The algorithm for feature discovery can be made much better with a good recall with minor tweaks.
- Extensive Sentiment Analysis can done on the reviews using the extracted features and the opinion words.
- Review Summarization can be performed using the relevant features and users’ opinions on them.

7 Acknowledgment

We thank Prof. Amitabha Mukerjee for giving us an opportunity to carry out this project under his guidance and also for his valuable support throughout the project, guiding us from time to time and looking into the project when it was needed. We would also like to thank Dr.Bing Liu from the University of Illinois at Chicago for providing the review dataset on his homepage. We acknowledge you for your support.

References

[1] Qiu, Guang, et al. ”Opinion word expansion and target extraction through double propagation.” Computational linguistics 37.1 (2011): 9-27.

[2] Zhang, Lei, et al. ”Extracting and ranking product features in opinion documents.” Proceedings of the 23rd International Conference on Computational Linguistics: Posters. Association for Computational Linguistics, 2010.

[3] Liu, Bing. ”Sentiment analysis and opinion mining.” Synthesis Lectures on Human Language Technologies 5.1 (2012): 1-167.

[4] Zhai, Zhongwu, et al. ”Clustering product features for opinion mining.” Proceedings of the fourth ACM international conference on Web search and data mining. ACM, 2011.