Asian Journal of Information Technology 15 (22): 4546-4550, 2016
ISSN: 1682-3915

Finding Infrequent Features During Feature Extraction in Opinion Mining Using Fuzzy Based Clustering

[1]G. Bharathi Mohan and [2]T. Ravi
[1]Jaya Engineering College, Chennai, India
[2]Madanapalle Institute of Technology and Science, Madanapalle, Andra Pradesh, India

Abstract: With the growing trend of e-commerce sites, blogs and web forums, people are keenly articulating their opinion on various products, topics. If we are buying a product for the first time, we would go through reviews which are already presented by the users who have used it. Manual analysis can be difficult and consumes more time, thus, a method is required to present the summary of the reviews. Reviews recorded by the users are unstructured in nature. Opinion mining is a discipline of web content mining which in turn is a category of web mining. The other categories of web mining are web structure and web usage mining. Opinion mining can be exploited by both companies and individuals. It involves natural language processing, text analysis and computational linguistics. The focus of the proposed system is mainly in extracting the aspects or features of the product which is the first step of opinion mining. An extension to the Intrinsic and Extrinsic Domain relevance method is made in order to support the rare features too. If the extraction step is improvised, the consequent steps will give fine grained outcomes and thus the result will be enhanced greatly.

Key words: Feature extraction, opinion mining, classification, sentiment analysis, web mining

INTRODUCTION

Opinion mining can be done at document level, word or phrase level. Mining in document level leads in finding the overall subjectivity which is extremely generic Word or phrase level too is not much efficient since it doesn't consider feature level granularity. Feature based opinion mining is the one in which the particular aspects (features) of the given product can be considered and their opinion polarities can be generated and analyzed. This is significant because the overall opinion differs from the feature level polarization. The taste of each user in liking a particular set of aspects in a product will vary from person to person. Based on their preferences, they can decide whether to buy the product or not. Thus feature based opinion mining will give fine grained results and this approach is followed in the proposed approach. Classification of opinion mining techniques is discussed by Mishra and Jha (2012). Opinion mining basically consists of the following steps:

- Identifying features of the product from the unstructured review
- Finding polarity of the opinion, i.e., positive or negative opinion
- Summarizing product's features, their overall opinion

Feature identification involves extracting the features of the product from the review. This involves POS tagging i.e tagging the review sentences into various parts of speech. There are various tools for tagging like Brill, OpenNLP, Stanford POS Tagger, GATE, NLTK with python. After tagging the text, a particular pattern is extracted in order to obtain feature terms and opinion words. Usually noun phrases are considered to be product features (Zhang and Liu, 2011).

Finding polarity involves developing a lexicon list. Sentiwordnet is a lexical resource• devised for supporting sentiment classification and opinion mining applications. There are also other techniques like using dictionary or thesaurus to find the related opinion words. Adjectives are usually considered to be opinion words e.g. "The story of the movie was unique". Here, the word 'unique' represents the opinion word. Adverbs also can act as opinion word in a few instances. Syntactic patterns or grammar chunks can be used to identify opinion words from the review sentences. In order to find polarity, Point wise mutual information is applied to the opinion words obtained and seed words to be discriminated as positive and negative. Semantic orientation is used to find the polarity. Supervised methods are also available to classify the reviews into Thumbs up (recommended) and thumbs down (not recommended). Opinion mining is evolved into sentiment analysis. There are many unresolved problems in NLP and new avenues are explored to work on various issues. Current web world involves sentiment analysis as a critical need for companies too. A detailed discussion about the trends, techniques and evolution of opinion mining and sentiment analysis is made by Cambria *et al.* 2013).

Corresponding Author: G. Bharathi, Jaya Engineering College, Chennai, India

Opinion mining is an important domain of the marketing and advertising domains. Advertiser prefers to analyze popularity of their ads that he/she posted on site. Because of automatic responders and other entities star rating based mechanism may go fraud. So, review system needs to be analyzed using natural language processing bases on comments. Fraud comments could be removed by using irrelevant comment removal mechanism suggested in study (Han and Kamber, 2011).

Literature review: Basically the work of feature extraction can be categorized into two: Supervised and unsupervised approaches. Comparison between supervised and unsupervised methods is presented in (Chaovalit and Zhou, 2005). The dataset used to compare is movie reviews.

In supervised approaches the problem is that it works well for the domain in which the system is trained. If we need to apply for the other domains, then we have to retrain the system according to that particular domain. Some of the previously used supervised approaches are based on CRF (Conditional Random Field), HMM (Hidden Markov Models) (Seerat and Azam, 2012; Jin *et al*., 2009) and other techniques.

In unsupervised technique we have many models in order to find the features. Some of the methods used are Syntax rules, POS (Parts of Speech) tagging and Term frequency. In the Syntactic rule method, words used in the review sentence are defined as a tree format and some dependency rules are defined which identifies the feature. When they are triggered then the particular terms are extracted as CF (Candidate Features). In order to apply these rules we need to tag the reviews using POS tagging. In POS tagging, the parts of speech of the given review sentences are tagged. There are so many tools available to perform POS tagging. Some of them which are used in the previous researches are Brill, Stanford POS tagger, NLProcessor etc. Example: "The pictures are very clear. Overall it is fantastic compact camera." In these review sentences, the terms 'pictures', 'compact' are some terms which describe the features about the camera (product). They are usually nouns. The adjectives which describes about the noun (features) are considered as the opinion word. Example 'clear' is the word which says about the picture. Predefined set of rules and patterns can be used to extract the candidate list of feature sets and also the opinion words. But there is a need of a way to prune the valid set of opinions from this list. Term frequency is used to find the most mentioned features which are considered as essential.

In, compactness pruning and redundancy pruning is used to find the valid set of features. In few works the frequent noun/noun phrases are considered to be the features. But they lack in finding implicit features. Example "It does not fit easily into the pockets", in this sentence the user mean that the product is not portable i.e. about the size of the product. But the term 'size' is not mentioned explicitly here. Finding implicit features require sophisticated techniques.

Su *et al.* (2008), an approach that clusters product features and opinion words simultaneously and iteratively by fusing both their content information and sentiment link information was framed. Under the same framework, based on the product feature categories and opinion word groups, sentiment association set between the two groups of data objects was constructed by identifying their strongest n sentiment links. Thus the hidden links are considered to be the implicit features (Su *et al*., 2008). Thus implicit features were also found by using this technique. But the precision values of the results were low. Some works also included in finding the nearest noun to the one or more opinion words which are already found which can be infrequent features. Another technique is proposed by McAuley and Leskovec (2013) to find hidden features in the text.

Popescu and Etzioni (2007), explicit features are found by identifying parts and properties of the given product class and then separating parts from properties. Implicit features are found by clustering opinion features and using PMI (Point-wise mutual information). Turney (2002), classification of reviews is done based on the semantic orientation of phrases in the review which contains adjective and adverbs. However, the context in adjectives is not sufficient to detect the polarity since reviews are domain dependent. Example- the word 'unpredictable' in a vehicle review (unpredictable steering) will differ from that of in a movie review (unpredictable plot).

There are various issues involved while extracting features. Example non noun features were not addressed; rare but valid features are missed since most work is based on finding frequent features. Grammatical mistakes done by users in reviews will not lead to efficient results after POS tagging.

Some work also includes association rule mining to find the frequent features. Latent Drichlet Association (LDA), a generative three way probabilistic model is also used to deal with aspect level opinion mining. There are also ranking based approaches which give relevant ranks to the various aspects of the products.

Opinion mining is evolved as sentiment analysis and it is one of the contemporary research topics in big data. Existing approaches are based on the following techniques, i.e., keyword spotting, lexical affinity, statistical methods and concept-level techniques.

MATERIALS AND METHODS

Proposed approach

Feature extraction: Although, there are many approaches in feature extraction, the product features techniques

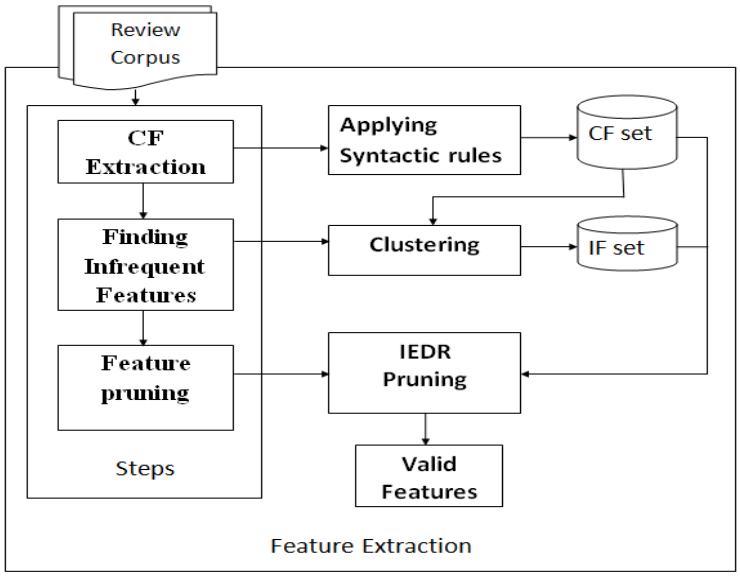

Fig. 1: Architecture of feature extraction of proposed system

focus only the given review corpus without considering their distributional characteristics in another domain corpus. But in this approach one domain dependent corpora and one domain independent corpora is considered. The disparity measure, Domain Relevance (DR) is calculated which characterizes the feature as intrinsic (domain relevant) or extrinsic (domain irrelevant). For each feature, EDR (extrinsic domain score) and IDR (Intrinsic Domain score) is calculated using the IEDR (Intrinsic and Extrinsic Domain Relevance) approach (Hai *et al*., 2014). Then finally valid features are pruned by considering the features which is less generic (EDR value is less than a threshold) and more domain specific (IDR value greater than a threshold).

The proposed approach is an extension of IEDR (Intrinsic and Extrinsic Domain Relevance). The drawback in the previous approaches is that it extracts features based on frequency and hence rare features are missed. The other drawbacks are that IEDR does not consider non-noun opinion features and it is at the mercy of other errors due to grammatical mistakes made by the reviewers. The first problem can be solved using clustering the product features so that infrequent features can be filtered out separately. The overall architecture of the proposed approach is represented in Fig. 1.

Candidate feature extraction: POS tagging is done using Stanford POS Tagger. Penn Tree bank project's Tagging model is used in for tagging. Then typed dependencies between the terms can be determined using the English-left 3 words-distsim Tagger model using Stanford Parser. Then the particular syntactic patterns are extracted by traversing through the parsed tree and extracting only noun phrases with particular dependencies. The resulting terms are indicated as candidate features. Tagging, Parsing can be done with a single tool called Stanford CoreNLP. But the results are not found to be efficient since the tool works well for sentence level and our approach uses a fine-grained model. However the resulting features may contain many invalid features which can be validated using the IEDR approach. The proposed method adds clustering before finally pruning the feature set.

Clustering product features: A micro level clustering is applied before clustering the frequent and infrequent features, i.e., lexically similar features can be clustered together so that a single leader component can be chosen to reduce the search space. This approach was followed in (Zhai *et al*., 2011). This can be accomplished using Wordnet which is a popular tool used in text mining and NLP. Clustering based on lexical similarity would reduce the number of terms to be evaluated during calculating term frequency. But, this is done in each review and not across the whole document since each user's view may vary sometimes. The proposed technique uses an additional algorithm, i.e., fuzzy clustering which comes under probabilistic model-based clustering. Since the

proposed framework has to deal with real time unstructured reviews, whose feature terms rely on the domain they deal with an accurate prior knowledge is not available. Probablistic based clustering is the one which can solve the problem of the previous approach.

Fuzzy clustering: Basically given a set of objects, X = {x1,... .xn}, a fuzzy set S is a subset of X that allows each object in X to have a membership degree between 0 and 1. Formally, a fuzzy set, S can be modeled as a function, $F_S:\times Fo\ [0,1]$.

We can apply fuzzy set idea on clusters (Li *et al*., 2010). Fuzzy clustering can be represented using a partition matrix, $M=[w_{ij}]$ $(1\leq i\leq n,\ 1\leq j\leq k)$, where w_{ij} is the membership degree of o_i in fuzzy cluster C_j. The values of partition matrix should be between 0 and 1. Sum of all values must be one. Atleast one object must be there whose membership value is non-zero. Sample of the result set of product id and the features present is described in Table 1.

The feature terms are partitioned into two clusters with the help of partitioning matrix.The partition matrix contains two columns since clustering involves only two clusters i.e. frequent features and infrequent features. Row represents the product whose feature terms are clustered i.e. product id. Value one in the matrix represents that the particular feature belongs to that cluster (corresponding column). Example, consider the matrix (1), value 1 in the first column indicates that it belongs to the first cluster. The number of rows will be same as that the number of products considered.

$$M=\begin{bmatrix}1 & 0\\ 1 & 0\\ . & .\\ 0 & 1\end{bmatrix} \quad (1)$$

$$w_{ij}\begin{cases}1, TF(ti)\geq th\\ 0,\ otherwise\end{cases} \quad (2)$$

The value of M is based on the wij which is defined in Eq. 2, where, wij represents the weight of the $term_i$ in the jth document. TF denotes the term-frequency of the term, t_i denotes the term $_i$ and T_h denotes threshold value. However the threshold cannot be pre-determined since the input domain can deal with any kind of product. It can be calculated based on experimental basis and changes according to the input-domain corpus which is considered.

Table 1: Product id and candidate features

Product_id	Features
110813405001	Camera, Sound, Battery
110813405002	Exterior, decor, ambiance, food
110813405003	Processing speed, Battery, Graphic
110813405004	Screen size, Audio, Software
110813405005	Aspect ratio, display, sound

Algorithm 1: Clustering frequent and infrequent features

Algorithm for clustering:

Input: Product id i, candidate features Cf_{ij}
Output: Cluster of frequent and infrequent features
Create and Initialize Cluster1 [], Cluster2 []
Create matrix M of size I×j
// where I = |products| & j = 2 (|clusters|)
Initialize Elements of M to 0
For each feature in each product
wij = 0
$M[i][j] = w_{ij}$
For each feature in each product
// Assign weight based on term frequency
Calculate TF (CFij) //Term frequency
If TF (CFij) = th then
$w_{ij} = 1$
Else
$w_{ij} = 0$
for each feature in each product
// clustering frequent and infrequent features separately
if M[i][j] is equal to 1
Add CFij to Cluster1
Else
Add CFij to Cluster2
Return cluster 1, Cluster 2

Algorithm used in the proposed approach for clustering where frequent and in-frequent features are clustered in to two groups.

RESULTS AND DISCUSSION

The dataset is obtained from Amazon Snap datasets and it contains nearly 35 million user reviews. Cellphone and its accessories are chosen as input dataset which contains about 78930 reviews. Stanford POS tagger and Stanford Parser are used for tagging and dependency parsing, respectively. The detailed description for the parser used is provided by (Witten *et al*., 2005). Figure 2 describes the dependencies between the grammatical parts found in a sentence. The standard of Penn Treebank format is followed.

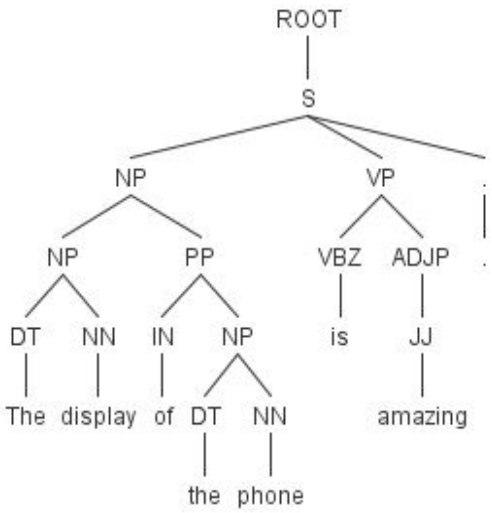

Fig. 2: Dependency tree format for a sentence

CONCLUSION

Thus, the problem of finding the infrequent features is solved using clustering and further soft constraints can be added to the clustering technique in order to obtain more appropriate features. The proposed approach is a fine-grained approach. However grammatical errors are unpreventable and real time reviews will have informal sentences. Non-noun features can be added by improving the syntax rules or instead using another novel technique to get non-noun opinion features.

REFERENCES

Cambria, E., B. Schuller, Y. Xia and C. Havasi, 2013. New avenues in opinion mining and sentiment analysis. IEEE. Intell. Syst., 28: 15-21.

Chaovalit, P. and L. Zhou, 2005. Movie review mining: A comparison between supervised and unsupervised classification approaches. Proceedings of the 38th Hawaii International Conference on System Sciences, (HICSS`05), Hawaii, pp: 1-9.

Hai, Z., K. Chang, J.J. Kim and C.C. Yang, 2014. Identifying features in opinion mining via intrinsic and extrinsic domain relevance. IEEE. Trans. Knowl. Data Eng., 26: 623-634.

Han, J. and M. Kamber, 2011. Data Mining: Concepts and Techniques: Concepts and Techniques. 3rd Edn., Elsevier, San Francisco, ISBN: 9780123814807, Pages: 744.

Jin, W., H.H. Ho and R.K. Srihari, 2009. A novel lexicalized HMM-based learning framework for web opinion mining. Proceedings of the 26th Annual International Conference on Machine Learning, June 14-18, 2009, ACM, Montreal, Quebec, ISBN:978-1-60558-516-1, pp: 465-472.

Li, F., C. Han, M. Huang, X. Zhu and Y.J. Xia *et al*., 2010. Structure-aware review mining and summarization. Proceedings of the 23rd International Conference on Computational Linguistics, August 23, 2010, Association for Computational Linguistics, Stroudsburg, Pennsylvania, pp: 653-661.

McAuley, J. and J. Leskovec, 2013. Hidden factors and hidden topics: Understanding rating dimensions with review text. Proceedings of the 7th ACM Conference on Recommender Systems, October 12-16, 2013, ACM, Hong Kong, China, ISBN:978-1-4503-2409-0, pp: 165-172.

Mishra, N. and C.K. Jha, 2012. Classification of opinion mining techniques. Int. J. Comput. Appl., 56: 1-6.

Popescu, A.M. and O. Etzioni, 2007. Extracting Product Features and Opinions from Reviews. In: Natural Language Processing and Text Mining. Anne, K. and R.P. Stephen (Eds.). Springer, Berlin, Germany, ISBN:978-1-84628-175-4, pp: 9-28.

Seerat, B. and F. Azam, 2012. Opinion mining: Issues and challenges (a survey). Int. J. Comput. Appl., 49: 42-51.

Su, Q., X. Xu, H. Guo, Z. Guo and X. Wu *et al*., 2008. Hidden sentiment association in chinese web opinion mining. Proceedings of the 17th International Conference on World Wide Web, April 21-25, 2008, ACM, Beijing, China, ISBN:978-1-60558-085-2, pp: 959-968.

Turney, P.D., 2002. Thumbs up or thumbs down?. Semantic orientation applied to unsupervised classification of reviews. Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, July 15, 2002, Association for Computational Linguistics, Stroudsburg, Pennsylvania, pp: 417-424.

Witten, H.I., E. Frank and M.A. Hall, 2005. Data Mining: Practical Machine Learning Tools and Techniques. 2nd Edn., Morgan Kaufmann Publ., Massachusetts.

Zhai, Z., B. Liu, H. Xu and P. Jia, 2011. Clustering product features for opinion mining. Proceedings of the fourth ACM International Conference on Web Search and Data Mining, February 9-12, 2011, Hong Kong, China, pp: 347-354.

Zhang, L. and B. Liu, 2011. Identifying noun product features that imply opinions. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. June 19-24, 2011, Portland, pp: 575-580.