# Unsupervised Approach for Aspect identification on Movie Reviews

**Enreina Annisa Rizkiasri**
4701224
e.a.rizkiasri@student.tudelft.nl

**Neha Sree Thuraka**
4743598
N.S.Thuraka@student.tudelft.nl

**Nivedita Prasad**
4712099
N.Prasad@student.tudelft.nl

**Chitra Balasubramanian**
4742907
C.Balasubramanian@student.tudelft.nl

## ABSTRACT
UPDATED—April 16, 2018. Sentiment analysis of a movie review helps in understanding the user's opinion about the movie. But, to gain better insights about the movie, one needs to find out about the aspects of the movie that made the user to come up with an opinion regarding the movie. In this project, we focus on using an unsupervised approach for aspect identification on movie reviews as a substep of Aspect Based Sentiment Analysis (ABSA). We identify the aspect words by selecting the nouns within movie reviews from the IMDB website. Then, we combine two techniques of aspect categorization: by finding people names that match the movie crews' names (directors, writers, and actors) and by applying clustering algorithm on the noun phrases. We observe the resulting clusters manually and assign each cluster to corresponding aspect categories of five: General, Plot, Cast, Music, and Directing. We finally evaluate the clustering result by first manually annotating aspects in a separate test set and then assigning nouns extracted from test dataset to the nearest cluster and compare it to the annotated aspects.

## Author Keywords
aspect extraction; aspect categorization; aspect based sentiment analysis; opinion mining;

## INTRODUCTION
As the era of Web 2.0 has emerged, more people have been sharing their opinion and thoughts online. Many organizations are interested in analyzing the opinions of users available on various web platforms. One prominent focus on analyzing opinion is sentiment analysis, which is a field of computational study that analyses the opinion, attitude, and emotion of people toward entities such as products, services, events, and topics [11].



*CHI'16,* May 07–12, 2016, San Jose, CA, USA



ISBN 123-4567-24-567/08/06. . . $15.00
DOI: http://dx.doi.org/10.475/123_4

Sentiment analysis has been applied to various domains: from product reviews on e-commerce website, blog posts, microblog posts, to movie reviews. Movie review is particularly challenging but has been popular in the field because it often contain terms specific to the movie, for example like names of the actors, directors, and certain events that happen in the movie. One of the widely known study of sentiment analysis on movie reviews was by Pang and Lee in their work of categorizing movie reviews in respect to rating scales from one to five "stars". [17].

While generally most approach of movie sentiment analysis is to find the pair $(s, g)$ where $s$ represents the sentiment and $g$ represents the target entity (in the case of movie sentiment analysis, it is the movie), the target could also be an *aspect* of the entity, which can be the characteristic or property of the entity [20]. This is called *Aspect-Based Sentiment Analysis (ABSA)* and would be the focus of our project particularly on movie reviews. The motivation behind ABSA is that a review might address different sentiment towards different aspect of an entity. A short review example would be: "I like the soundtrack accompanying the movie, but the plot is really boring." which expresses a positive sentiment towards the music aspect of the movie, but a negative sentiment towards the plot aspect.

ABSA has been done numerous times, but most of the focus is on the domain of either product reviews or restaurant reviews ([6], [1], [5], [24]). There are several application on movie reviews ([2], [21], [18]), but manually labeled movie reviews dataset for ABSA is hardly available publicly, and labeling aspects is a tedious task. That motivation brings the focus of our project which is to apply ABSA on movie reviews using an unsupervised approach which does not involve any manual labeling on movie review aspects.

Although the whole project was intended to focus on the whole process of ABSA, we decided to narrow the focus to one important subtask of ABSA which is Aspect identification. Aspect identification refers to the task of identifying words or phrases that refers to a certain aspects of the movie in the reviews. We will discuss on how the work can be further improved by discussing the sentiment analysis part in the future work section.

This report is structured as follows. The subsequent section will discuss some previous works related to aspect-based sentiment analysis both inside and outside movie review domain. Then, we discuss the whole methodology of our work reproducing the mentioned related works. The methodology section consists of 4 subsections which discuss the dataset, aspect extraction, aspect categorization, and our evaluation method. After that, we discuss the result of our study and finally gives out some conclusion and future work.

## RELATED WORKS

This section will discuss some related works where we based our project on. We will first discuss some general work in the field of ABSA, and then approaches specific to the movie reviews domain. Then we discuss some methods in the subtask of aspect identification. We finally explain the difference between our work and the related works in the field of aspect identification for ABSA on movie reviews.

Several related works have been done and has been summarized by Schouten and Frasincar in their literature survey of ABSA [20]. In order to do aspect detection in reviews, several approaches has been explored ranging from frequency-based ([6], [12]), syntax-based ([25]), supervised machine learning ([7], to unsupervised machine learning ([10]).

Most of the previous works of ABSA had their focus on the domain of product reviews such as camera or laptop reviews, and also restaurant reviews, as the annotated dataset is available publicly, provided for one of the task in SemEval [19]. The number of ABSA work on movie reviews is less popular and a bit more challenging, as the writing style of movie reviews differs from product and restaurant reviews, with larger range of variety of domain-specific terms.

Additionally in the case of movie reviews, the names of characters, actors, and directors are often mentioned. Some works of aspect identification on movie reviews such as [21] and [26] built a dictionary from the metadata (i.e. crew names) of the movies to handle these kind of terms. Most of the work of ABSA on movie reviews used lexicon to identify aspect terms ([18]), while there is also a work that use a supervised ([14]) and semi-supervised approach ([2]).

There are two kinds of aspect expression that can be inferred from a review: implicit and aspect expression. Zhang et al. in [23] defines aspect expressions in a sentence that are noun or noun phrases as explicit aspect expression, while expressions in the adjective form (i.e. "beautiful" refers to the visual aspect) or a phrase such as "easy to understand" is called as implicit aspect expression. Our work will focus on identifying explicit aspect by extracting nouns as potential aspect terms as nouns have been proved to be strong aspect candidates ([4]) and similar approach was also been used in [6] and [26].

A work by Alghunaim et al. tries to utilize vector representation of words to train a model for aspect term extraction, aspect category detection, and aspect sentiment prediction [1]. For the aspect term extraction part, they compute the vector representations of words using the skip-gram model of Word2Vec ([15]) trained on Google News dataset and Yelp restaurant reviews, and then trained CRFSuite [] and SVM-HMM [] on an already annotated restaurant reviews.

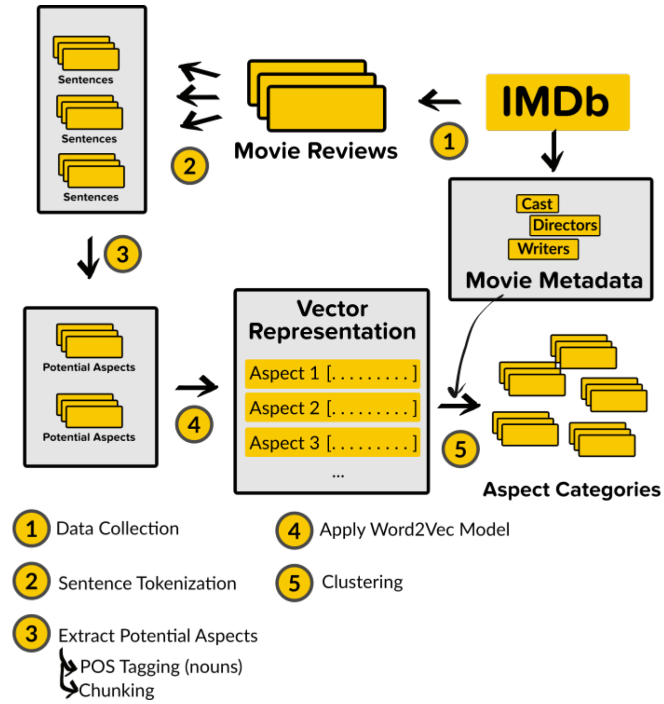


**Figure 1. Overview of Methodology**

For aspect categorization, we take note on the work by Bancken et al. which tries to use K-medoids clustering technique [9] to group extracted aspect terms based on their semantic similarity [3]. Their work does not need any labeling for the aspect categorization part, as it depends on the clustering technique using the WordNet-based similarity measure, Jcn [8].

Our work will focus on the aspect identification on movie reviews and based on the word embeddings approach in [1] for representing the potential aspect terms as numeric vectors, while also adopting the clustering approach used in [3] to categorize the aspect terms.

## METHODOLOGY

The overall methodology is depicted in Figure 1. This section discusses each step in more detail, first by describing the dataset, then to aspect extraction, and aspect categorization. The evaluation method is also discussed by the end of this section.

### Dataset

The dataset that we use for ABSA is the Large Movie Review Dataset constructed by Maas et al. [13]. It originally consists of 50,000 movie reviews from IMDB. The dataset is originally used for binary sentiment classification of movie reviews. Each review is labeled as either having a positive sentiment or a negative sentiment. The original research divided the dataset into two: 25,000 reviews each for training and testing.

In order to adjust the dataset for ABSA, we did some preprocessing before splitting each review into sentences. The preprocessing consist of removing extra white spaces, quotation marks and some special characters that do not provide much information for sentiment analysis. Then each review is split into sentences as shown by step **2** in Figure 1. This process is done both to the training dataset and the testing dataset of the original dataset. As the preprocessing and the aspect extraction step (described later) takes quite some time, we only managed to use a subset of the original dataset. The details of the dataset is shown in Table 1

**Table 1. Dataset**

| Sentiment | #Movies | #Reviews | #Sentences |
|---|---|---|---|
| Positive | 21 | 610 | 7.330 |
| Negative | 29 | 777 | 11.070 |

The training dataset, in our case, would be used to learn the aspect extraction model that would later be evaluated on the test dataset. As the detail on how do we evaluate our method on the test dataset would be described in the Evaluation section.

Furthermore, as most movie reviews could contain movie titles and crew names in the content, we also collected the metadata of each movie by using the IMDB urls provided in the original dataset. As the urls were already obsolete, we had to extract the movie id from the urls and use the OMDB API to fetch the appropriate movie metadata. This metadata will later be used in the aspect categorization step.

### Aspect Extraction

The aspect extraction task is described as the step to identify words which indicate what aspect is the review is referring to. We give examples of identified aspects in two review sentences in Table 2. The highlighted text denotes the words or phrases which refers to which aspect of the movie the review is discussing about.

In order to identify aspects from the review sentence, the steps taken in our methodology consist of: parts of speech (POS) tagging; chunking; and selecting nouns.

**Table 2. Identified Aspects**

| Sentence #1 |
|---|
| The special effects however do not overwhelm the simple story of good triumphing over evil. |
| **Sentence #2** |
| All of that is mainly due to the great story, the good directing and the good acting performances of the actors. |

#### *POS Tagging*

The first step for the aspect extraction step is to apply POS tagging to each sentence of the reviews. The POS tagging outputs part-of-speech annotation to each word of a review sentence. We use the R package for Ripple Down Rules-based Part-Of-Speech Tagger (RDRPOS) tagger [16] to perform the POS tagging on our dataset. The POS tag of each word will be used for the next steps: chunking and selecting nouns.

#### *Chunking*

Three types of chunking namely: noun-noun, adverb-verb and particle-adjective has been performed. For example, it chunks James and Cameron together so that in further steps we would be considering James Cameron as a single block instead of two different words. Additionally, the later two types of chunking help in better sentiment analysis. $(not, good)$ chunks together into a block. $(really, excited)$ would get chunked. This can be utilized for scoring the sentiment, where, 'excited' can be 1 and 'really excited; can be assigned 2.

#### *Selecting Nouns*

After we have the chunks of a sentence, we filter the chunks which were marked as noun by the POS tagger. The motivation behind selecting only noun aspects, nouns tend to be used for referring aspect explicit aspects. This is a simple heuristic but proven to be satisfactory in [6].

We can also see in Table 2 that all of the highlighted words are actually nouns. Although not every nouns is necessarily an aspect. This is why we use the term "potential aspects" (as seen in Figure 1) for referring to the output of this step of aspect extraction. The next step of aspect categorization tries to group related aspect words together while also identifying words which does not actually refer to an aspect.

### Aspect Categorization

Aspect Extraction of our methodology outputs the potential aspect words contained in a movie review sentence. We further categorize this aspect words to their respective aspect categories for better understanding. Based on previous work of ABSA on movie reviews ([]), we decide on 5 aspects categories which are *Cast, Music, General, Plot, and Directing*. Table 3 depicts each category with their corresponding aspect term examples.

**Table 3. Aspect Categories**

| Category | Aspect Terms |
|---|---|
| General | movie |
| Plot | storyline, plot, story |
| Cast | actor, actress, acting |
| Music | soundtrack, background music |
| Directing | direction, director |

The aspect categorization (depicted as step 5 in Figure 1) would assign each of potential aspects to the corresponding category. For example, the word "acting" and "actors" would fall into the **Cast** category, whereas the word "story" would fall into the **plot** category.

In order to categorize the aspects from the previous step of the methodology, we use a combination of two kind of techniques. The first technique utilizes the movie metadata explained in the Dataset section to automatically categorize words involving names of the actors, directors, and writers of the movie being reviewed. This technique has been used before in [26], where they use regular expression matching to find people names in the review, and then match it with the cast library of the movie. An example of the usage of this technique is shown in Table 4.

Table 4. Aspects based on Movie Metadata

| Sentence #1 |
| --- |
| Morgan Freeman looked incredibly uncomfortable, especially when made to dance around to rock music for no apparent reason half way through the film after him and Timberlake meet. |
| **Sentence #2** |
| Director Samuel Fuller has crafted an exceptional drama set amongst the seedy underworld of New York City. |

In the highlighted phrases of Table 4, the term "Morgan Freeman" and "Timberlake" would be categorized into the **Cast** aspect, while the term "Samuel Fuller" would be categorized in the **Directing** aspect.

The second technique is used for the categorizing the rest of the nouns: the ones which are not identified as movie crews' names. For this task, we use the unsupervised approach of clustering to automatically groups nouns together based on their relatedness.

In order to do the define the relatedness between aspect terms, we compute the vector representation of each word using the skip-gram model of Word2Vec trained on the whole movie reviews dataset consisting of 75.000 reviews (25.000 training and an additional of 50.000 unsupervised dataset from [13]) with 16.490 unique words. We trained the model to construct 300-dimensional word vectors using Word2Vec implementation in Gensim[1]. Additionally, for comparison purpose, we also use the cut-down version[2] of the Word2Vec model trained on the the Google News dataset [15]. We will mention both models as IMDB Word2Vec and GoogleNews Word2Vec especially in the Evaluation Method and Result section. This approach of using Word2Vec model has been done before by Alghunaim et al. for for ABSA on restaurant reviews [1].

We convert all of the tokens from the aspect extraction part into their respective vector representation based on the Word2Vec models. Then, we feed the numeric vectors to the clustering algorithm.

We use the Scikit Learn[3] implementation of k-means algorithm to group similar aspect words together based on their vector representation, as it was employed in aspect clustering before in [3] and [22]. We assign the number of clusters $k$ to be 6, that is our expectation of the clustering would result into 5 clusters that represents the 5 categories, with an additional one cluster for terms that do not fall into any of the 5 categories.

After that, we observe the resulting clusters manually by skimming through the terms that falls into each cluster. We then assign appropriate category label for each cluster, and the cluster will be used to classify new aspect terms outside the training dataset.

[1]https://radimrehurek.com/gensim/models/word2vec.html
[2]https://github.com/eyaler/word2vec-slim/
[3]http://scikit-learn.org/

### Evaluation Method

In order to evaluate our methodology, we use the clustering result to predict aspect words and categories on a separate test dataset, as explained before in the Dataset section, which is also obtained from the Large Movie Review Dataset [13].

To evaluate our methodology, we first did a manual identification of aspects and their corresponding categories from sentences in the test set. Each word of a sentence is labeled as **Aspect** or **Non-Aspect**, and then for the ones which marked as **Aspect**, we assign the appropriate aspect category (either General, Story, Cast, or Music). The number of aspects that we managed to label is depicted in Table 5.

Table 5. Test Dataset

| Sentiment | #Movies | #Reviews | #Sentences | #Aspects |
| --- | --- | --- | --- | --- |
| Positive | 3 | 89 | 1.416 | 1.787 |
| Negative | 4 | 101 | 1.085 | 1.080 |

Then we apply the same preprocessing as we did on the training set: breaking down the reviews into sentences; POS tagging; chunking; selecting nouns; and assigning appropriate aspect categories for occurrences of director, writer, or cast names. For the rest of the found nouns, we use their vector representation (using the Word2Vec model) and use the resulting clusters from the training phase to assign the closest cluster as their corresponding aspect category.

We then compare the categories assigned by the clustering and the manually annotated ones. From this comparison, we follow a similar approach of evaluation in [26] by calculating the precision and recall as follows:

$$Precision = \frac{\text{True positives}}{\text{True positives + False negatives}}$$

$$Recall = \frac{\text{True positives}}{\text{True positives + False positives}}$$

Both the metrics will be calculated category-wise (i.e. within the **Plot** aspects, within **Cast** aspects, etc), in addition to the precision and recall of discriminating between **Aspect** and **Non-Aspect**.

Furthermore, we will calculate the evaluation metrics using the vector representation generated with the IMDB Word2Vec and GoogleNews Word2Vec model. We intend to compare the performance between the two to have an insight if there is any effect of movie specific domain information on the quality of the word representation.

## RESULT

This section will discuss the result of evaluating our methodology. First we will see the resulting clusters for the aspect categorization step, then we will try to test our cluster model to categorize potential aspects in separate test dataset.

### Cluster Results

Table 6 and Table 7 depicts the top 10 frequent words in each cluster of the clustering result.

**Table 6. Cluster Result (IMDB Word2Vec)**

| C1 | C2 | C3 | C4 | C5 | C6 |
|---|---|---|---|---|---|
| money | flicks | video | community | **performance** | friend |
| point | sequel | role | history | **actors** | team |
| day | picture | star | humans | **characters** | girl |
| kind | series | **director** | country | style | grant |
| friends | flick | name | force | **music** | mark |
| right | stuff | **actor** | problems | character | car |
| way | **films** | version | **tale** | **plot** | family |
| time | **movies** | fan | world | **scene** | home |
| people | **film** | work | self | **scenes** | man |
| game | movie | fans | life | story | house |

**Table 7. Cluster Result (GoogleNews Word2Vec)**

| C1 | C2 | C3 | C4 | C5 | C6 |
|---|---|---|---|---|---|
| war | **actor** | line | dance | friend | john |
| elements | **music** | mark | eyes | brother | smith |
| audience | **character** | time | heart | friends | douglas |
| stuff | **scene** | name | room | girl | god |
| something | films | home | box | family | kelly |
| voice | **story** | way | shop | star | steven |
| **dialogue** | **scenes** | **director** | lines | fan | crap |
| kind | movies | version | town | people | jim |
| style | film | game | car | man | hollywood |
| life | movie | work | house | fans | richard |

As we can see from the words contained in the clusters, the clustering algorithms did not do so well in separating aspect terms based on our pre-defined aspect categories. For example, in the case of using the IMDB Word2Vec model, some words like **director** and **actor** fall into the same cluster, and there are even two clusters which contain mostly non-aspect words (**C1** and **C6**).

In case of clustering using the vector representation generated by the GoogleNews Word2Vec, there is one cluster (**C2** which contains many movie related phrases, while the other clusters just contains a few or none at all.

We will discuss more about the shortcoming of the clustering approach in the conclusion section.

### Evaluation on Test Dataset

As stated in the evaluation methodology, we intended to calculate the precision and recall on the manually labeled dataset (which is available in a public repository mentioned in the end of this paper), but the clustering result turns out to be unreliable for us to separate each aspect to the categories. We will discuss more about work that needs to be done to evaluate our methodology further.

## POTENTIAL IMPROVEMENTS

Polysemy refers to the association of one word with two or more distinct meanings. The meaning of such word can only be determined by examining its context in the sentence. For example, in *'Which flights **serve** breakfast?'*, serve means present to someone whereas, in *'Does Lufthansa **serve** Amsterdam?'*, serve means providing service. Incorporating Word Sense Disambiguation, be it in the stage of POS tagging or aspect extraction or aspect categorization, would help in attaining improved results. For this, WordNet that mimics human logic and focuses on the word senses and connections between real-world entities can be used instead of Word2Vec. Further, the test set used for evaluation has been annotated manually. To assure that this is more precise, it is recommended to use the data annotated by multiple people.

## CONCLUSION

In this project, we tried to propose a methodology for aspect extraction on movie reviews using clustering approach in order to categorize aspects. We found out that the clustering method used is not sufficient enough to separate aspect terms based on their category. We suspect the following for the cause of such poor performance.

First, the distance metric used in the K-Means might not be appropriate to calculate the distance between word vectors. The implementation of k-means that we use, utilize the Euclidean distance, while we learned that word or text similarity in generally use the cosine distance.

Second, there is the case of imbalance data between number of nouns which are aspect and non-aspect. There are far larger number of non-aspect noun terms than aspect nouns. This may cause the cluster results to be not as expected, as k-means specifically tries to evenly cluster the words.

Additionally, there are many things to do to continue this work and improve the methodology, such as trying out different word representation and features, using different clustering algorithms, and improve the manual annotation procedure in order to evaluate the methodology. Most importantly, the work should also be extended to include the sentiment analysis task itself as part of ABSA.

In conclusion, we learned so much about ABSA by reading through related works and implement our own methodology for aspect identification. Although we did not succeed in evaluating the methodology properly, we think our work, including the scripts and dataset can be a base for future work on ABSA specifically in movie reviews domain.

## SCRIPTS AND DATASET

The scripts and dataset that we use is available in the public repository: https://github.com/enreina/beta-nlp-project.

## REFERENCES


1. Abdulaziz Alghunaim, Mitra Mohtarami, Scott Cyphers, and James Glass. 2015. A Vector Space Approach for Aspect Based Sentiment Analysis. *Proceedings of NAACL-HLT 2015* (2015), 116–122. http://www.aclweb.org/anthology/W15-1516
2. Deepa Anand and Deepan Naorem. 2016. Semi-supervised Aspect Based Sentiment Analysis for Movies Using Review Filtering. *Procedia Computer Science* 84 (2016), 86–93. DOI: http://dx.doi.org/10.1016/j.procs.2016.04.070
3. Wouter Bancken, Daniele Alfarone, and Jesse Davis. Automatically Detecting and Rating Product Aspects from Textual Customer Reviews. (????). http://ceur-ws.org/Vol-1202/paper1.pdf
4. Pavel Blinov and Eugeny Kotelnikov. 2014. Blinov: Distributed Representations of Words for Aspect-Based Sentiment Analysis at SemEval 2014. (2014), 140–144. http://www.aclweb.org/anthology/S14-2020
5. P D Blinov and E V Kotelnikov. 2013. Semantic Similarity for AspectâĂŚBased Sentiment Analysis. (2013). http://www.dialog-21.ru/media/1111/blinovpdkotelnikovev.pdf
6. Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '04*. ACM Press, New York, New York, USA, 168. DOI: http://dx.doi.org/10.1145/1014052.1014073
7. Niklas Jakob and Iryna Gurevych. 2010. Extracting opinion targets in a single-and cross-domain setting with conditional random fields. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing* October (2010), 1035âĂŞ1045. http://www.ukp.tu-darmstadt.de/peoplehttp://portal.acm.org/citation.cfm?id=1870759
8. Jay J. Jiang and David W. Conrath. 1997. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. *Proceedings of International Conference Research on Computational Linguistics (ROCLING X)* (1997). DOI:http://dx.doi.org/10.1.1.269.3598
9. Leonard Kaufman and Peter J Rousseeuw. Finding Groups in Data, An Introduction to Cluster Analysis. (????). https://pdfs.semanticscholar.org/567d/2061396061c928084ea6664f50c4200cc739.pdf
10. Himabindu Lakkaraju, Chiranjib Bhattacharyya, Indrajit Bhattacharya, and Srujana Merugu. 2011. Exploiting Coherence for the Simultaneous Discovery of Latent Facets and associated Sentiments. *Proceedings of the 2011 SIAM International Conference on Data Mining* (2011), 498–509. https://pdfs.semanticscholar.org/4b8d/a9dabbd7488dbb44690ba903e135f8b3be5c.pdf
11. Bing Liu. 2012. Sentiment Analysis and Opinion Mining. *Synthesis Lectures on Human Language Technologies* 5, 1 (5 2012), 1–167. DOI: http://dx.doi.org/10.2200/S00416ED1V01Y201204HLT016
12. Bing Liu and South Morgan Street. 2005. Opinion Observer : Analyzing and Comparing Opinions on the Web. *Proceedings of the 14th International Conference on World Wide Web* (2005), 342–351. DOI: http://dx.doi.org/10.1145/1060745.1060797
13. Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning Word Vectors for Sentiment Analysis. (????). http://ai.stanford.edu/~amaas/papers/wvSent_acl2011.pdf
14. Asha S Manek, P. Deepa Shenoy, M. Chandra Mohan, and K. R. Venugopal. 2017. Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier. *World Wide Web* 20, 2 (2017), 135–154. DOI: http://dx.doi.org/10.1007/s11280-015-0381-x
15. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. (????). https://arxiv.org/pdf/1301.3781.pdf
16. Dat Quoc Nguyen, Dai Quoc Nguyen, Dang Duc Pham, and Son Bao Pham. 2014. RDRPOSTagger: A Ripple Down Rules-based Part-Of-Speech Tagger. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*. 17–20. DOI: http://dx.doi.org/10.3115/v1/E14-2005
17. Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. (6 2005). http://arxiv.org/abs/cs/0506075
18. Viraj Parkhe and Bhaskar Biswas. 2014. Aspect based sentiment analysis of movie reviews: Finding the polarity directing aspects. *Proceedings - 2014 International Conference on Soft Computing and Machine Intelligence, ISCMI 2014* (2014), 28–32. DOI: http://dx.doi.org/10.1109/ISCMI.2014.16
19. Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, and Ion Andrutsopoulos. 2016. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. 342–349. DOI: http://dx.doi.org/10.18653/v1/S16-1055

20. Kim Schouten and Flavius Frasincar. 2016. Survey on Aspect-Level Sentiment Analysis. *IEEE Transactions on Knowledge and Data Engineering* 28, 3 (3 2016), 813–830. DOI: http://dx.doi.org/10.1109/TKDE.2015.2485209
21. Tun Thura Thet, Jin Cheon Na, and Christopher S.G. Khoo. 2010. Aspect-based sentiment analysis of movie reviews on discussion boards. *Journal of Information Science* 36, 6 (2010), 823–848. DOI: http://dx.doi.org/10.1177/0165551510388123
22. Shufeng Xiong and Donghong Ji. 2015. Exploiting Capacity-Constrained K-Means Clustering for Aspect-Phrase Grouping. In *International Conference on Knowledge Science, Engineering and Management*, Vol. 9403. Springer, Cham, 370–381. DOI: http://dx.doi.org/10.1007/978-3-319-25159-2{_}34
23. Lei Zhang and Bing Liu. 2014. Aspect and Entity Extraction for Opinion Mining. 1–40. DOI: http://dx.doi.org/10.1007/978-3-642-40837-3{_}1
24. Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. 2010a. Jointly Modeling Aspects and Opinions with a MaxEnt-LDA Hybrid. (2010), 56–65. http://wing.comp.nus.edu.sg/~antho/D/D10/D10-1006.pdf
25. Yanyan Zhao, Bing Qin, Shen Hu, and Ting Liu. 2010b. Generalizing syntactic structures for product attribute candidate extraction. *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics* June (2010), 377–380. http://www.aclweb.org/anthology/N10-1059
26. Li Zhuang, Feng Jing, and Xiao-Yan Zhu. 2006. Movie review mining and summarization. *Proceedings of the 15th ACM international conference on Information and knowledge management - CIKM '06* (2006), 43. DOI: http://dx.doi.org/10.1145/1183614.1183625