# Unsupervised acquisition of domain aspect terms for Aspect Based Opinion Mining

## *Adquisición no supervisada de aspectos de un dominio para Minería de Opiniones Basada en Aspectos*

**Aitor García-Pablos,**
**Montse Cuadros, Seán Gaines**
Vicomtech-IK4 research centre
Mikeletegi 57, San Sebastian, Spain
{agarciap,mcuadros,sgaines}@vicomtech.org

**German Rigau**
IXA Group
Euskal Herriko Unibertsitatea
San Sebastian, Spain
german.rigau@ehu.es

**Resumen:** El análisis automático de la opinión, que usualmente recibe el nombre minería de opinión o análisis del sentimiento, ha cobrado una gran importancia durante la última década. La minería de opinión basada en aspectos se centra en detectar el sentimiento con respecto a "aspectos" de la entidad examinada (i.e. características o partes concretas evaluadas en una sentencia). De cara a detectar dichos aspectos se requiere una cierta información sobre el dominio o temática del contenido analizado, ya que el vocabulario varía de un dominio a otro. El objetivo de este trabajo es generar de manera automática una lista de aspectos del dominio partiendo de un set de textos sin etiquetar, de manera completamente no supervisada, como primer paso para el desarrollo de un sistema más completo.
**Palabras clave:** aspectos de dominio, adaptación a dominio, minería de opinión

**Abstract:** The automatic analysis of opinions, which usually receives the name of opinion mining or sentiment analysis, has gained a great importance during the last decade. This is mainly due to the overgrown of online content in the Internet. The so-called aspect based opinion mining systems aim to detect the sentiment at "aspect" level (i.e. the precise feature being opinionated in a clause or sentence). In order to detect such aspects it is required some knowledge about the domain under analysis. The vocabulary in different domains may vary, and different words are interesting features in different domains. We aim to generate a list of domain related words and expressions from unlabeled domain texts, in a completely unsupervised way, as a first step to a more complex opinion mining system.
**Keywords:** aspect based sentiment analysis, unsupervised lexicon generation

## 1 Introduction

Opinion mining and sentiment analysis has attracted the attention of the research community during the last decade (Pang and Lee, 2008; Liu, 2012; Zhang and Liu, 2014). Specially during the last years, when the opinionated content flows thanks to the so called Web 2.0. Review web sites, blogs and social networks, are producing everyday a massive amount of new content, much of it bearing opinions about different entities, products or services. Trying to cope with this data is infeasible without the help of automatic Opinion Mining tools which try to detect, identify, classify, aggregate and summarize the opinions expressed about different topics. The opinion mining systems can be roughly classified into two types, supervised, and unsupervised or semi-supervised since some level of supervision is almost always required to guide or initialize most of the existing systems. Supervised systems require training data, which usually includes manually annotated data, in order to train a model that can "learn" how to label new unseen data. These systems perform quite well, but it is difficult to port to different domains or languages due to the cost of obtaining such manually annotated data. Unsupervised methods (or semi-supervised) try to leverage the vast amount of unlabeled data (i.e. all the content that is constantly generated over the Internet) to infer the required information without the need of big amounts of hand-crafted resources. These systems have the clear advantage of being much more portable to other languages or domains. In this work we will briefly introduce the concept of "aspect based

opinion mining" and some of the existing approaches in the literature. Then we will introduce the Semeval 2014 task 4, which is about detecting opinionated aspect targets and their categories and polarities in customer review sentences. After that we will explain our approach to generate a list of aspect terms for a new domain using a collection of unlabeled domain texts. Finally we show our results after evaluating the approach against Semeval 2104 task 4 datasets, and our conclusions and future work.

## 2 Related Work

Customer reviews are full of fine grained opinions and sentiments towards different aspects, features or parts of a product or service. In order to discover which aspects are being praised and which are being criticized a fine grained analysis is required. Many approaches have been carried out.

Hu and Liu (2004) try to summarize customer reviews in a aspect level basis. They employ frequent nouns and phrases as potential aspects, and use relations between aspects and opinions to identify infrequent aspects. Popescu and Etzioni (2005) extract high frequent noun phrases in reviews as candidate product aspects. Then, they compute the Pointwise Mutual Information (PMI) score between the candidates and some meronymy discriminators associated with the product class to evaluate each candidate.

Zhuang, Jing, and Zhu (2006) employ certain dependency relations to extract aspect-opinion pairs from movie reviews. They first identify reliable dependency relation templates from training data to identify valid aspect-opinion pairs in test data.Wu et al. (2009) use dependency parsing to extract noun phrases and verb phrases as aspect candidates. Blair-Goldensohn (2008) refine the approach proposed in Hu and Liu (2004) considering only noun phrases inside sentiment-bearing sentences or in some syntactic patternz indicating sentiment, plus some additional filters to remove unlikely aspects.

Qiu et al. (2009) propose a double propagation method to bootstrap new aspect terms and opinion words from a list of seeds using dependency rules. The process is called double propagation because they use opinion words to obtain new aspect terms and aspect terms to obtain new opinion words.

The acquired opinion words and aspect terms are added to the seed lists, and used to obtain more words in a new loop. The process stops when no more words can be acquired. In Zhang et al. (2010) the double propagation approach is extended to aspect ranking to deal with the noise that double propagation method tends to generate. The authors model the aspect terms and opinion words as a bipartite graph and use HITS algorithm to rank the aspect terms, also using some linguistics patterns (e.g. part-whole relation patterns).

In this work we reuse some of these ideas to build an unsupervised system that bootstrap a ranked list of domain aspect terms just by using a set of unlabeled domain texts (customer reviews of a particular topic). We evaluate our results against the SemEval 2014 task 4 datasets.

## 3 SemEval 2014 Task 4

SemEval 2014 task 4[1] *Aspect Based Sentiment Analysis* (Pontiki et al., 2014) provides two training datasets, one of restaurant reviews and other of laptop reviews. The restaurant review dataset consists of over 3,000 English sentences from restaurant reviews borrowed from Ganu, Elhadad, and Marian (2009). The laptop review dataset consist of over 3,000 English sentences extracted from customer reviews. The task is divided in four different subtasks. Subtask 1 is *aspect term extraction*: given a set of sentences referring to pre-identified entities (i.e. restaurants or laptops), return the list of distinct aspect terms present in the sentence. An aspect term names a particular aspect of the target entity (e.g. *menu* or *wine* for restaurants, *hard disk* or *battery life* for laptops). Subtask 2 focuses on detecting the polarity of a given set of aspect terms in a sentence. The polarity in this task can be one of the following: *positive, negative, neutral* or *conflict*. The objective of subtask 3 is to classify the identified aspect terms into a predefined set of categories. The categories can be seen as a more coarse grained aspects that include the aspect terms. In this SemEval task the predefined set of categories for restaurants are: *food, service, price, ambiance* and *anecdotes/miscellaneous*. No categories have been provided for the laptop do-

---

[1]http://alt.qcri.org/semeval2014/task4/

main. Subtask 4 is analogous to the subtask 2, but in this case the polarity has to be determined for the aspect categories. Again, only the restaurant dataset is suitable for this task since the laptop dataset does not contain aspect category annotations.

In this paper we focus our attention on subtask 1, *aspect term extraction*. Our aim is to develop an unsupervised system able to extract aspect terms from any domain and evaluate it against the SemEval datasets, using the evaluation tools and metrics provided by the tasks organizers.

## 4   Our approach

Our aim is to build a system that is capable of generating a list of potential aspect terms for a new domain without any kind of adaptation or tuning. Such a list can be a useful resource to exploit in a more complex system aiming to perform Aspect Based Sentiment Analysis. Aspect terms, also known as *opinion targets* in the literature, generally refer to parts of features of a given entity. For example, *wine list* and *menu* could be aspect terms in a text reviewing a restaurant, and *hard disk* and *battery life* could be aspect terms in a laptop review. Obviously, each domain has its own set of aspect terms, referring to different aspects, parts and features of the entities described in that domain. The only requirement to generate the list of aspect terms for a new domain is a, preferably large, set of unlabelled documents or review describing entities of the domain. Our method combines some techniques already described in the literature with some modifications and additions.

### 4.1   Used data

Using a web-scraping program we have extracted a few thousand English reviews from a restaurant review website[2], and a similar amount of English reviews from a laptop review website[3]. We have not performed any kind of sampling or preprocessing on the extracted data, it has been extracted "as-is" from the list of entities (restaurants and laptops) available in the respective websites at the time of the scraping. The extracted reviews have been split in sentences unsing Stanford NLP tools and stored into an XML

---

file. A subset of 25,000 sentences have been used to acquire the aspect term lists, combined with the already mentioned 3,000 sentences of the Semeval 2014 task 4 datasets.

### 4.2   Double propagation

We have adapted the double-propagation technique described in Qiu et al. (2009) and Qiu et al. (2011). This method consists of using an initial seed list of aspect terms and opinion words and propagate them through a dataset using a set of propagation rules. The goal is to expand both the aspect term and opinion word sets. Qiu et al. (2009) define opinion words as words that convey some positive or negative sentiment polarities. They only use nouns as aspect terms, and only adjectives can be opinion words. This is an important restriction that limits the recall of the process, but the double-propagation process is intended to extract only explicit aspects (i.e. aspects that are explicitly mentioned in the text, and not aspects implicitly derived from the context). The detection of implicit aspects (e.g. "The phone fits in the pocket" referring to the size) requires a different set of techniques and approaches that are described in many works in the literature Fei et al. (2012; Hai, Chang, and Cong (2012).

During the propagation process a set of propagation rules are applied to discover new terms (aspect terms or opinion words), and the initial aspect term and opinion word sets are expanded with each new discovery. The newly discovered words are also used to trigger the propagation rules, so in each loop of the process additional words can be discovered. The process ends when no more words can be extracted. Because aspect terms are employed to discover new opinion words, and opinion words are employed to discover new aspect terms, the method receives the name of double-propagation.

The propagation is guided by some propagation rules. When the conditions of a rule are matched, the target word (aspect term or opinion word) is added to its correspondent set.

### 4.3   Propagation rules

The propagation rules are based on dependency relations and some part-of-speech restrictions. We have mainly followed the same rules detailed in Qiu et al. (2011) with some

minor modifications. The exact applied rules used this work can be observed in the Table 1.

Some rules extract new aspect terms, and others extract new opinion words. In Table 1, T means *aspect term* (i.e. a word already in the aspect terms set) and O means *opinion word* (i.e. a word already in the opinion words set). W means *any word*. The dependency types used are *amod*, *dobj*, *subj* and *conj*, which stand for *adjectival modifier*, *direct object*, *subject* and *conjunction* respectively. Additional restrictions on the Part-Of-Speech (POS) of the words present in the rule, it is shown in the third column of the table. The last column indicates to which set (aspect terms or opinion words) the new word is added.

To obtain the dependency trees and word lemmas and POS tags, we use the Stanford NLP tools[4]. Our initial seed words are just *good* and *bad*, which are added to the initial opinion words set. The initial aspect terms set starts empty. This way the initial sets are not domain dependent, and we expect that, if the propagation rules are good enough, the propagation should obtain the same results after some extra iterations.

Each sentence in the dataset is analyzed to obtain its dependency tree. Then the rules are checked. If a word and its dependency-related words trigger the rule, and the conditions hold, then the word indicated by the rule is added to the corresponding set (aspect terms or opinion words, depending on the rule). The process continues sentence by sentence adding words to both sets. When the process finishes processing sentences, if new words have been added to any of the two sets, the process starts again from the first sentence with the enriched sets. The process stops when no more words have been added during a full dataset loop.

## 5 Ranking the aspect terms

Although the double-propagation process populates both sets of domain aspect terms and domain opinion words, we focus our attention in the aspect terms set. Depending on the size and content of the employed dataset, the number of potential aspect terms will be quite large. In our case the process

generates many thousands of different potential aspect terms. Much of them are incorrect, or very unusual aspect terms (e.g. in the restaurant domain, a cooking recipe written in another language, a typo, etc.). Thus, the aspect terms needs to be ranked, trying to keep the most important aspects on top, and pushing the less important ones to the long tail.

In order to rank the obtained aspect terms, we have modeled the double-propagation process as a graph population process. Each new aspect term or opinion word discovered by applying a propagation rule is added as a vertex to the graph. The rule used to extract the new word is added as an edge to the graph, connecting the originating word and the discovered word.
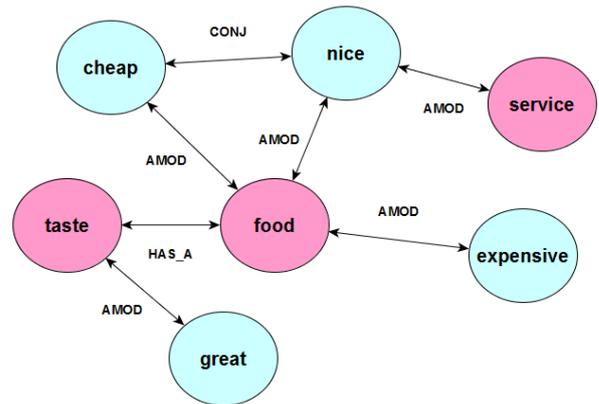


Figure 1: Example of a graph fragment constructed with the bootstrapped words and relations.

Figure 1 presents as an example a small part of the graph obtained by the double-propagation process. Each vertex representing a word maintains the count of how many times that word has appeared in the dataset, and also if it is an aspect term or an opinion word. A word is identified by its lemma and its POS tag. Every edge in the graph also maintains a count of how many times the same rule has been used to connect a pair of words. At the end of the double-propagation process the generated graph contains some useful information: the frequency of appearance of each word in the dataset, the frequency of each propagation rule, the number of different words related to a given word, etc. We have applied the well-known PageRank algorithm on the graph to score the vertices. To calculate the PageRank scores we have

| Rule | Observations | Constraints | Action |
|---|---|---|---|
| R11 | O → amod→W | W is a noun | W→T |
| R12 | O→dobj→W1 ←subj←W2 | W2 is a noun | W2→T |
| R21 | T ← amod ← W | W is an adjective | W→O |
| R22 | T → subj → W1 ← dobj ← W2 | W2 is an adjective | W2→ O |
| R31 | T → conj → W | W is a noun | W → T |
| R32 | T → subj → has *gets* dobj ← W | W is a noun | W → T |
| R41 | O → conj → W | W is an adjective | W→ O |
| R42 | O → Dep1 → W1 ← Dep2 ← W2 | Dep1==Dep2, W2 is an adjective | W2→ O |

Table 1: Propagation rules

| Restaurants | Laptops |
|---|---|
| 1- food | 1-battery life |
| 2- service | 2- keyboard |
| 3- staff | 3- screen |
| 4- bar | 4- feature |
| 5- drink | 5- price |
| 6- table | 6- machine |
| 7- menu | 7- toshiba laptop |
| 8- dish | 8- windows |
| 9- atmosphere | 9- performance |
| 10- pizza | 10- use |
| 11- meal | 11- battery |
| 12- bartender | 12- program |
| 13- price | 13- speaker |
| 14- server | 14- key |
| 15- dinner | 15- hard drive |

Table 2: Top ranked aspect terms for restaurant and laptop domain using our approach

used the JUNG framework[5], a set of Java libraries to work with graphs. The value of the alpha parameter that represents the probability of a random jump to any node of the graph has been left at 0.15 (in the literature it is recommended an alpha value between 0.1 and 0.2).

The graph is treated as an undirected graph because the propagation rules represented by the graph edges can be interpreted in both directions (e.g. A modifies to B, or B is modified by A). The aspect terms are then ordered using their associated score, being the most relevant aspect term the one with the highest score.

### 5.1  Filtering undesired words

The double-propagation method always introduces many undesired words. Some of these undesired words appear very frequently and are combined with a large number of words. So, they tend to also appear in high positions in the ranking.

Many of these words are easy to identify, and they are not likely to be useful aspect terms in any domain. Examples of these words are: *nothing, everything, thing, anyone, someone, somebody,* etc. They are extracted during the double-propagation process because they appear in common expressions like *It was a good thing, It is nothing special, I like everything.* The process also extract other words, like *year, month, night,* and other time expressions. Also, some common words, like *boy, girl, husband* or *wife.* The reason for this is that the input texts are customers reviews, and it is quite common to find anecdotes and personal comments like *I saw a nice girl in the bar.* It would be interesting to find an automatic method to safely remove all these words, valid for many domains. A TF-IDF weigthing of the words may be a useful preprocessing to identify noisy content. For this work we have chosen the simple approach of adding them to a customizable stop word list. The final list contains about one hundred words that are not likely to be aspect terms in any domain. The list has been crafted observing the most common unwanted words after running the system, and using intuition and common sense. Our purpose was not to tune the stop word list to work better with any of our evaluation domains, and the same stop word list has been used in the evaluation in both restaurant and laptop domains.

### 6  Dealing with multiword terms

Many aspect terms are not just a single word, but compounds and multiword terms. For some domains this is more critical than for others. As it can be observed in Table 2, the top ranked aspect term for laptops is *battery life.* The laptop domain is a very challenging domain due to the amount of technical vocabulary that usually combine several words (e.g. *hard disk drive, Intel i7 processor,* etc.). In

---
[5] http://jung.sourceforge.net

order to improve the precision and the recall of the generated set of aspect terms, multi-word aspect terms must be detected and included in the resulting sets. We have tried different approaches, trying increase the recall without adding incorrect terms.

## 6.1 Using WordNet

One of the approaches included in the system exploits WordNet [6], and some simple rules. Each time a word is going to be processed during the double-propagation algorithm, the combination of the current word plus the next word is checked. If some conditions are satisfied then we treat both words as a single multiword term. The conditions are the following:

- If word n and word n+1 are nouns, and the combination is an entry in WordNet (or in Wikipedia, see below). E.g.: *battery life*

- If word n is an adjective and word n+1 is a noun, and the combination is an entry in WordNet. E.g.: *hot dog, happy hour*

- If word n is an adjective, word n+1 is a noun, and word n is a relational adjective in WordNet (lexical file 01). E.g.: *Thai food, Italian food*

## 6.2 Using Wikipedia

In order to improve the coverage of the Word-Net approach, we also check if a combination of two consecutive nouns appears as a Wikipedia article title. Wikipedia articles refer to real word concepts and entities, so if a combination of words is a title of a Wikipedia article it is very likely that this word combination is also meaningful for the domain under analysis (e.g. *DVD player, USB port, goat cheese, pepperoni pizza*). However, since Wikipedia contains many entries that are titles of films, books, songs, etc., that would lead to the inclusion of erroneous multiword expressions, for example *good time*. For this reason we limit the lookup in Wikipedia titles just to combination of nouns, avoiding combinations of adjective + noun. This gives a good balance between extended coverage and inclusion of incorrect aspect terms.

---

[6] http://wordnet.princeton.edu/

| SemEval Restaur. | Precision | Recall | F-score |
|---|---|---|---|
| SemEval Baseline | 0.539 | 0.514 | 0.526 |
| Our system (S) | 0.576 | 0.649 | 0.610 |
| Our system (W) | 0.555 | 0.661 | 0.603 |
| Our system (W+S) | 0.551 | 0.662 | 0.601 |
| SemEvaml-Best | 0.853 | 0.827 | 0.840 |

Table 4: Result comparison for SemEval restaurant review dataset

## 6.3 Using simple patterns

In this work we have limited the length of the multiword terms to just bi-grams. But in some cases it is interesting to have word combinations of a bigger size. For that purpose we have included some configurable patterns to treat longer chains of words as a single aspect term. The patterns are very simple, being expressed with a simple syntax like *A of N*. It means that a known aspect term (represented by the uppercased A) followed by the word *of*, followed by a noun (represented by the uppercased N) must be processed as a single aspect term. Similar patterns would be *N of A, A with N, N with A*, etc. These patterns are useful to extract expressions like *chicken with onion*, or *glass of wine*.

## 7 Evaluation

To evaluate the quality of the resulting aspect term lists, we have used our method to annotate the SemEval 2014 datasets of task 4, *Aspect Based Sentiment Analysis* which provides two datasets, one for "restaurants" domain and another for "laptops" domain. An example of the format can be seen in the Figure 3. The datasets are composed by individual sentences. Each sentence contains annotated data about the aspect terms present in that sentence. The aspect terms are the span of characters inside the sentence that holds the mention to the aspect.

The SemEval task provides an evaluation script which evaluates standard precision, recall and F-score measures. Both datasets (restaurants and laptops) contain 3,000 sentences each. The restaurant dataset contains 3,693 labeled gold aspect term spans (1,212 different aspect terms), and the laptop dataset contains 2,358 labeled gold aspect term spans (955 different aspect terms). We use these gold aspect terms to evaluate the experiments.

The experiment using our approach consists of using the generated aspect term lists

```
<sentence id="270">
        <text>From the incredible food, to the warm atmosphere, to the
        friendly service, this downtown neighborhood spot doesn't miss a beat.
        </text>
        <aspectTerms>
            <aspectTerm term="food" polarity="positive" from="20" to="24"/>
            <aspectTerm term="atmosphere" polarity="positive" from="38" to="48"/>
            <aspectTerm term="service" polarity="positive" from="66" to="73"/>
        </aspectTerms>
</sentence>
```

Table 3: Example of SemEval 2014 Task 4 dataset sentence

| SemEval Laptops | Precision | Recall | F-score |
|---|---|---|---|
| SemEval Baseline | 0.401 | 0.381 | 0.391 |
| Our system (S) | 0.309 | 0.475 | 0.374 |
| Our system (W) | 0.327 | 0.508 | 0.398 |
| Our system (W+S) | 0.307 | 0.533 | 0.389 |
| SemEval-Best | 0.847 | 0.665 | 0.745 |

Table 5: Result comparison for SemEval laptop review dataset

(for restaurants and laptops) to annotate the sentences. The generated aspect term lists have been limited to the top 550 items. In this particular experiment, we have observed than using longer lists increases the recall, but decreases the precision due to the inclusion of more incorrect aspects terms. The annotation process is a simple lemma matching between the words in the dataset and the words in our generated lists.

We compare the results against the SemEval baseline which is also calculated by some scripts provided by the Semeval organizers. This baseline splits the dataset into *train* and *test* subsets, and uses all the labeled aspect terms in the *train* subset to build a dictionary of aspect terms. Then it simply uses that dictionary to label the test subset for evaluation. We also show the result of the best system submitted to SemEval (SemEval-Best in the table) for each domain. However the results are not comparable since our approach is unsupervised and just a first step to a more complex system and does not use any machine learning or other supervised techniques to annotate the data.

Tables 4 and 5 show the performance of our system with respect to the baselines in both datasets. "Our system (S)" stands for our system only using the SemEval provided data (as it is unsupervised it learns from the available texts for the task). (W) refers to the results when using our own dataset

scraped from the Web. Finally (W+S) refers to the results using both SemEval and our Web dataset. On the restaurant dataset our system outperforms the baseline and it obtains quite similar results on the laptop dataset. Interestingly, the results are quite similar even if the learning datasets are very different in size. Probably this is because it only leverages more documents if they include new words that can be bootstrapped. If the overall distribution of words and relations does not change, the resulting aspect term list would be ranked very similarly.

Apart from the non-recognized aspect terms (i.e. not present in the generated list) another important source of errors is the multiword aspect term detection. In the SemEval training dataset, about the 25% of the gold aspect terms are multiword terms. In the restaurant dataset we find a large number of names of recipes and meals, composed by two, three or even more words. For example *hanger steak au poivre* or *thin crusted pizza* are labeled as single aspect terms. In the laptop domain multiword terms are also very important, due to the amount of technical expressions (i.e. hardware components like "RAM memory", software versions like "Windows 7" and product brands like "Samsung screen"). These aspect terms cannot be present in our automatically acquired aspect term list because we limit the multiword length up to two words.

There are also errors coming from typos and variations in the word spelling (e.g. *ambience* and *ambiance*) that our system does not handle.

## 8 Conclusions and future work

Aspect term extraction (also known as *features* or *opinion targets*) is an important first step to perform fine grained automatic opinion mining. There are many approaches in

the literature aiming to automatically generate aspect terms for different domains. In this paper we propose a simple and unsupervised method to bootstrap and rank a list of domain aspect terms using a set of unlabeled domain texts. We use a double-propagation approach, and we model the obtained terms and their relations as a graph. Then we employ PageRank algorithm to score the obtained terms. We evaluate the approach in the SemEval 2014 Task 4 and our unsupervised system performs better than the supervised baseline. In our future work we will try to improve the way we deal with multi-word terms and the propagation method to reduce the amount of erroneous aspect terms and generate a better ranking of the resulting terms.

## References

Blair-Goldensohn, S. 2008. Building a sentiment summarizer for local service reviews. *WWW Workshop on NLP in the Information Explosion Era.*

Fei, Geli, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. 2012. A Dictionary-Based Approach to Identifying Aspects Im-plied by Adjectives for Opinion Mining. 2(December 2012):309–318.

Ganu, Gayatree, N Elhadad, and A Marian. 2009. Beyond the Stars: Improving Rating Predictions using Review Text Content. *WebDB*, (WebDB):1–6.

Hai, Zhen, Kuiyu Chang, and Gao Cong. 2012. One seed to find them all: mining opinion features via association. *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 255–264.

Hu, Minqing and Bing Liu. 2004. Mining opinion features in customer reviews. *AAAI.*

Liu, Bing. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

Pang, Bo and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Pontiki, Maria, Dimitrios Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. *Proceedings of the International Workshop on Semantic Evaluation (SemEval).*

Popescu, AM and Oren Etzioni. 2005. Extracting product features and opinions from reviews. *Natural language processing and text mining*, (October):339–346.

Qiu, Guang, Bing Liu, Jiajun Bu, and Chun Chen. 2009. Expanding Domain Sentiment Lexicon through Double Propagation. *IJCAI.*

Qiu, Guang, Bing Liu, Jiajun Bu, and Chun Chen. 2011. Opinion word expansion and target extraction through double propagation. *Computational linguistics*, (July 2010).

Wu, Yuanbin, Qi Zhang, Xuanjing Huang, and Lide Wu. 2009. Phrase dependency parsing for opinion mining. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pages 1533–1541. Association for Computational Linguistics.

Zhang, L, Bing Liu, SH Lim, and E O'Brien-Strain. 2010. Extracting and ranking product features in opinion documents. *Proceedings of the 23rd International Conference on Computational Linguistics*, (August):1462–1470.

Zhang, Lei and Bing Liu. 2014. Aspect and Entity Extraction for Opinion Mining. *Data Mining and Knowledge Discovery for Big Data.*

Zhuang, Li, F Jing, and XY Zhu. 2006. Movie review mining and summarization. *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 43–50.

---

[7]http://www.opener-project.eu/
[8]http://nlp.lsi.upc.edu/skater/