

TJUdeM: A Combination Classifier for Aspect Category Detection and Sentiment Polarity Classification

Zhifei Zhang and Jian-Yun Nie
Dept. of Comp. Sci. and Oper. Res.
University of Montreal
Quebec H3C 3J7, Canada
{zhanzhif, nie}@iro.umontreal.ca

Hongling Wang
Dept. of Comp. Sci. and Tech.
Soochow University
Suzhou 215006, China
hlwang@suda.edu.cn

Abstract

This paper describes the system we submitted to In-domain ABSA subtask of SemEval 2015 shared task on aspect-based sentiment analysis that includes aspect category detection and sentiment polarity classification. For the aspect category detection, we combined an SVM classifier with implicit aspect indicators. For the sentiment polarity classification, we combined an SVM classifier with a lexicon-based polarity classifier. Our system outperforms the baselines on both the laptop and restaurant domains and ranks above average on the laptop domain.

1 Introduction

Sentiment analysis aims at identifying people's opinions, sentiments, attitudes, and emotions towards entities and their attributes (Liu, 2012), which has a wide range of applications on user-generated content, e.g., reviews, blogs, and tweets.

Most previous work in sentiment analysis mainly attempted to identify the overall polarity of a given text or text span (Pang and Lee, 2008; Wilson et al., 2009; Zhang et al., 2009). The document-level or sentence-level sentiment classification is often insufficient for applications. Each document may talk about different entities, or express different opinions about different aspects of the entity even if the document concerns a single entity. Therefore, we need to discover the aspects of entities and determine the sentiment polarity on each entity aspect. This task is called aspect-based sentiment analysis or feature-based opinion mining (Hu and Liu, 2004).

The aspect-based sentiment analysis (ABSA) task (Task 12) (Pontiki et al., 2015) in SemEval 2015 is a continuation of SemEval 2014 Task 4 (Pontiki et al., 2014). The ABSA task consists of two subtasks: In-domain ABSA and Out-domain ABSA. We participated in the former subtask that aims to identify the aspect category (i.e., an entity and attribute pair) and the sentiment polarity given a review text about a laptop or a restaurant.

Each entity and attribute pair is an aspect category chosen from the predefined inventories of entity types and attribute labels per domain. For the aspect category detection, an SVM classifier with the bag-of-words features can be used, and this approach is used as our baseline method. However, if a token implying an aspect, e.g., "overpriced", is not taken as a feature, the SVM classifier cannot correctly identify its corresponding category. Therefore, we enhance the results from the SVM classifier by using implicit aspect indicators (Cruz-Garcia et al., 2014). For the sentiment polarity classification, an SVM classifier with the bag-of-words features plus the category feature is trained and this is used as our baseline. However, again, if a sentiment word does not appear in the training data, the SVM classifier cannot predict its polarity. Therefore, we combined the SVM classifier and a lexicon-based polarity classifier (Taboada et al., 2011).

The remainder of this paper is organized as follows. In Section 2, we describe our approach to the aspect category detection. In Section 3, our approach to the sentiment polarity classification is presented. Experimental results are shown in Section 4. Section 5 provides the conclusion.

2 Aspect Category Detection

The aspect category detection task is to identify the specific entities and their attributes about the laptop or restaurant reviews. We use an SVM classifier enhanced by implicit aspect indicators. The process of the whole system is illustrated in Figure 1. We will describe the details in the following subsections.

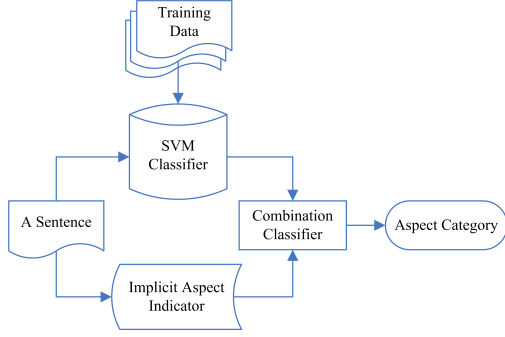


Figure 1: System flowchart for aspect category detection.

2.1 SVM Classifier

The SVM classifier uses words as features to determine the aspect categories. We use the LIBSVM package (Chang and Lin, 2011) to implement an SVM classifier. The “-t” option is set to 0 for linear kernel, and the “-b” option is set to 1 for probability estimates. The top n frequent tokens in the training data are used as the bag-of-words features. We set $n = 1000$ as the number of bag-of-words features.

An aspect category (C) is an entity (E) and attribute (A) pair, i.e., $C = E\#A$. For instance,

I received prompt service with a s-mile. $\rightarrow \{\text{Service}\#\text{General}\}$

It would cost too much to repair it. $\rightarrow \{\text{Support}\#\text{Price}\}$

For a test sentence s , the LIBSVM package can predict the probability of assigning each category $E\#A$ to s . The category should be assigned to s only if its probability is higher than a predefined threshold t . We set t to 0.2 for the restaurant reviews and to 0.12 for the laptop reviews. It’s easy to see that our SVM classifier is configured in accordance with the SVM baseline system provided by the task organizers (Pontiki et al., 2015).

$$\text{Aspect}_{svm}(s) = \{E\#A | \text{Prob}(E\#A) > t\} \quad (1)$$

2.2 Implicit Aspect Indicator

If the tokens implying aspects are beyond bag-of-words features, the SVM classifier is unable to predict it. For example,

It was totally overpriced- fish and chips was about \$15.

Both “overpriced” and “\$15” in the above sentence are associated with the “price” aspect. These tokens are considered as the implicit aspect indicators.

The different methods can be used to identify the implicit aspect indicators (Cruz-Garcia et al., 2014). In our case, we do it manually by setting a set of indicators for several aspects (see Table 1). The list of words associated with the “price” aspect includes “cost”, “overpriced”, “expensive”, etc. The list for the “quality” aspect includes “feels”, “durable”, “taste”, etc.

Implicit Aspect	Word List	Size
Price	expensive, overpriced, cheap, discount, cost,	16
Quality	feels, durable, overcooked, taste, breaks,	50
Performance	improves, stable, crashed, performs, powerful,	40
Design	lightweight, heavy, elegant, fit, looks,	27
Usability	access, store, typing, flexible, upgrade,	62

Table 1: Implicit aspect indicator.

In addition, an expression of the amount of money is strongly related to the “price” aspect. To identify these expressions, we use the following regular expression: “\s\$\d+(\.\d+)?\s\$”.

If the word W indicates the implicit aspect A' , the aspects determined by implicit aspect indicators are denoted as follows:

$$\text{Aspect}_{iai}(s) = \{A' | W \in s\} \quad (2)$$

2.3 Combination Classifier

We find the SVM classifier often obtains the category like “ E |General” which means that a general

opinion is expressed and it is not specific to a particular aspect. On the other hand, for the same case, the implicit aspect indicators may suggest other specific aspect categories (e.g., “price”). This case occurs when the words corresponding to the implicit aspect indicators are not included in the features used by SVM. It is in this case that it is the most useful to combine the two classifiers.

Our combination is done as follows: if the “General” category is suggested by the SVM classifier, then we replace it by the categories identified through the implicit aspect indicators. Otherwise, the categories given by the SVM classifier remain unchanged. The method is described in the following algorithm.

Algorithm 1 A combination classifier for aspect category detection.

Input: $Aspect_{svm}(s)$ and $Aspect_{iai}(s)$ for a test sentence s

Output: $Aspect(s)$

```

1: if  $Aspect_{iai}(s) = \emptyset$  then
2:   return  $Aspect_{svm}(s)$ 
3: end if
4:  $Aspect(s) = \emptyset$ 
5: for all  $E\#A \in Aspect_{svm}(s)$  do
6:   if  $A$  is ‘General’ then
7:     for all  $A' \in Aspect_{iai}(s)$  do
8:        $Aspect(s) = Aspect(s) \cup \{E\#A'\}$ 
9:     end for
10:  else
11:     $Aspect(s) = Aspect(s) \cup \{E\#A\}$ 
12:  end if
13: end for
14: return  $Aspect(s)$ 

```

3 Sentiment Polarity Classification

The sentiment polarity classification task is to assign a polarity from a set $\{positive, negative, neutral\}$ to each identified aspect category of a sentence. We use a similar method as for the previous task. The processes of the system are illustrated in Figure 2 that includes three parts: an SVM classifier, a lexicon-based polarity classifier, and their combination classifier.

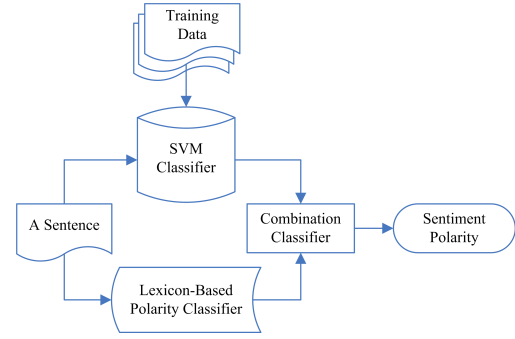


Figure 2: System flowchart for sentiment polarity classification.

3.1 SVM Classifier

We also use the LIBSVM package (Chang and Lin, 2011) to implement an SVM classifier with linear kernel. Again, n ($n = 1000$) bag-of-words features are extracted from the training data. In addition, a feature that indicates the aspect category is used. Our SVM configurations are also the same with that of the SVM baseline system (Pontiki et al., 2015).

The SVM classifier can predict a polarity (*positive*, *negative*, or *neutral*) for each aspect category C within a test sentence s . We represent three polarity labels with three respective numbers.

$$Polarity_{svm}(s, C) \in \{1, -1, 0\} \quad (3)$$

3.2 Lexicon-Based Polarity Classifier

If the sentiment words are beyond the bag-of-words features, the SVM classifier assigns the neutral polarity, and what’s worse, it assigns the reverse polarity if the sentence contains negation words (Zhu et al., 2014), like “not” and “no”. In fact, the lexicon-based methods can also be effective in sentiment classification (Taboada et al., 2011). We therefore adopt a simple lexicon-based method in our system.

The sentiment lexicons, such as Bing Liu’s Opinion Lexicon (Hu and Liu, 2004) and MPQA Subjectivity Lexicon (Wilson et al., 2009), are used to generate our sentiment word list. We denote all positive words and negative words by *POS* and *NEG* respectively.

We use the Stanford Parser package (Klein and Manning, 2003) for POS tagging and parsing. The typed dependency “*neg*(X, Y)” shows that one sentence contains a negation Y modifying X , and “*root*($ROOT, X$)” shows that X is a core word.

Assume that one sentiment word X is in a test sentence s and $X \in POS \cup NEG$, if $X \in POS$, then $Polarity(X) = 1$, otherwise $Polarity(X) = -1$. The polarity for the aspect category is determined by,

$$Polarity_{lex}(s, C) = \begin{cases} -Polarity(X) & \exists neg(X, Y) \\ -Polarity(X) & \exists neg(Z, Y) \\ Polarity(X) & \wedge root(ROOT, Z) \\ & otherwise \end{cases} \quad (4)$$

where $Y \in s$ is a negation word, and $Z \in s$ but $Z \notin POS \cup NEG$.

The following examples are corresponding to three circumstances in the above equation:

*Overpriced and **not** tasty* $\{neg(tasty, not)\}$

*Our experience did **not** ever get any better* $\{neg(get, not), root(ROOT, get)\}$

Overpriced and not tasty $\{root(ROOT, overpriced)\}$

3.3 Combination Classifier

If none of the sentiment words in the lexicon appear in a sentence, the lexicon-based polarity classifier is helpless, but the SVM classifier could still determine a reasonable polarity (Pang et al., 2002).

We propose a classifier combining the SVM classifier and the lexicon-based polarity classifier. It works as follows: If there is disagreement between the polarity of SVM classifier and the lexicon, we will rely on the polarity based on the lexicon if the latter is not neutral (0). Otherwise, we take the polarity of the SVM classifier.

Algorithm 2 A combination classifier for sentiment polarity classification.

Input: $Polarity_{svm}(s, C)$ and $Polarity_{lex}(s, C)$
for an aspect category C of a test sentence s

Output: $Polarity(s, C)$

- 1: **if** $Polarity_{svm}(s, C) = Polarity_{lex}(s, C)$ **then**
- 2: $Polarity(s, C) = Polarity_{svm}(s, C)$
- 3: **else if** $Polarity_{lex}(s, C) = 0$ **then**
- 4: $Polarity(s, C) = Polarity_{svm}(s, C)$
- 5: **else**
- 6: $Polarity(s, C) = Polarity_{lex}(s, C)$
- 7: **end if**
- 8: **return** $Polarity(s, C)$

4 Experiments

4.1 Data Sets

The training and test data is described in Table 2.

Domain		Training	Test
Laptop	Sentence	1739	761
	Category	Positive	1103
		Negative	541
		Neutral	765
	Total	106	79
Restaurant	Sentence	1974	949
	Category	Positive	1315
		Negative	685
		Neutral	1198
	Total	403	454
		53	45
		1654	845

Table 2: Data sets.

The laptop training data, consisting of 1739 sentences, includes 1974 aspect category instances. The laptop test data, consisting of 761 sentences, includes 949 aspect category instances. The restaurant training data, consisting of 1315 sentences, includes 1654 aspect category instances. The restaurant test data, consisting of 685 sentences, includes 845 aspect category instances.

There are 22 entity labels and 9 attribute labels on the laptop domain, and there are 6 entity labels and 5 attribute labels on the restaurant domain.

4.2 Experimental Results

Aspect category detection Table 3 lists the results of our system for the aspect category detection.

	Laptop	Restaurant
SVM Baseline	0.4631	0.5133
Top	0.5086	0.6268
Average	0.4548	0.5383
Our System	0.4649	0.5245

Table 3: F-score comparison for aspect category detection.

Our system clearly outperforms the SVM baseline on both two domains. This indicates that the implicit aspect indicators can further improve the performance. Our system ranks above average on the laptop domain. But our system is far from the top system. This is possibly due to the simple features

used by the SVM classifier. Globally, our method is comparable to the average performance of all the participating systems.

Sentiment polarity classification Table 4 lists the results of our system for the sentiment polarity classification. The majority baseline is obtained by majority voting in all the participating results.

	Laptop	Restaurant
SVM Baseline	0.6997	0.6355
Majority Baseline	0.5701	0.5373
Top	0.7935	0.7870
Average	0.7131	0.7132
Our System	0.7323	0.6888

Table 4: F-score comparison for sentiment polarity classification.

The performance of our system is obviously better than two baselines on both two domains, but fails to reach the average on the restaurant domain. The conclusion of this experiment is that the lexicon-based method is helpful to sentiment classification when it is combined with a baseline method. As for the task of aspect category detection, a possible reason lies in the simple bag-of-words features we used. With more sophisticated features, one can likely improve the performance of the baseline methods, and as a result, the combination method.

Comparing the results on the two domains, we observe that our system produced lower performance than average for the restaurant reviews, but higher performance for the laptop reviews. A possible reason can be the lexicon we defined for the two domains. The Opinion Lexicon is originally designed for the customer reviews about 5 digit products, which is more related to the laptop domain.

5 Conclusions

In this task, we proposed a combination classifier for the aspect category detection which combines an SVM classifier with implicit aspect indicators, and a combination classifier for the sentiment polarity classification which combines an SVM classifier with a lexicon-based polarity classifier. Our system ranks above average on the laptop domain and outperforms the baselines, but is still lower than the av-

erage for the restaurant domain. Our experiments show that implicit aspect indicators and polarity lexicon are both useful in these tasks. For the future work, more and better features will be examined to help to improve the classification performance.

Acknowledgments

We are really grateful to the organizers and reviewers for this interesting task and their helpful suggestions and comments. This research is supported by the Quebec-China Postdoctoral Scholarship (File No. 188040).

References

- Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27:1–27:27.
- Ivan Omar Cruz-Garcia, Alexander Gelbukh, and Grigori Sidorov. 2014. Implicit aspect indicator extraction for aspect-based opinion mining.
- Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of KDD*, pages 168–177, New York, NY, USA.
- Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of ACL*, pages 423–430, Sapporo, Japan.
- Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.
- Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135.
- Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of EMNLP*, pages 79–86, Philadelphia, PA, USA.
- Maria Pontiki, Haris Papageorgiou, Dimitrios Galanis, Ion Androutsopoulos, John Pavlopoulos, and Suresh Manandhar. 2014. SemEval-2014 Task 4: Aspect based sentiment analysis. In *Proceedings of SemEval*, pages 27–35, Dublin, Ireland.
- Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 Task 12: Aspect based sentiment analysis. In *Proceedings of SemEval*, Denver, CO, USA.
- Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307.

- Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics*, 35(3):399–433.
- Changli Zhang, Daniel Zeng, Jiexun Li, Fei-Yue Wang, and Wanli Zuo. 2009. Sentiment analysis of Chinese documents: From sentence to document level. *Journal of the American Society for Information Science and Technology*, 60(12):2474–2487.
- Xiaodan Zhu, Hongyu Guo, Saif Mohammad, and Svetlana Kiritchenko. 2014. An empirical study on the effect of negation words on sentiment. In *Proceedings of ACL*, pages 304–313, Baltimore, MD, USA.