Aspect-based Sentence Segmentation for Sentiment Summarization

Jingbo Zhu Muhua Zhu
Natural Language Processing Lab
Northeastern University
Shenyang, Liaoning, China, 110004
zhujingbo@mail.neu.edu.cn
zhumuhua@gmail.com

Huizhen Wang
Natural Language Processing Lab
Northeastern University
Shenyang, Liaoning, China, 10004
wanghuizhen@mail.neu.edu.cn

Benjamin K. Tsou
City University of Hong Kong
Hong Kong
rlbtsou@cityu.edu.hk

ABSTRACT
Aspect-based sentiment summarization systems generally use sentences associated with relevant aspects extracted from the reviews as the basis for summarization. However, in real reviews, a single sentence often exhibits several aspects for opinions. This paper proposes a two-stage segmentation model to address the challenge of identifying multiple single-aspect and single-polarity units in one sentence, namely aspect-based sentence segmentation. Our model deals with both issues of aspect change and polarity change occurring in the input sentence. Experiments on restaurant reviews show that our model outperforms state-of-the-art linear text segmentation methods.

Categories and Subject Descriptors
H.2.8 [**Database Management**]: Database Application – *data mining*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing – *text analysis*.

General Terms
Algorithms, Experimentation

Keywords
Aspect-based sentiment summarization, text segmentation, sentence segmentation

1. INTRODUCTION
In recent years one emerging research field related to opinion mining is *sentiment summarization* that aims to aggregate and represent sentiment information drawn from online customer reviews on products or social issues [1]. In practice, for example, to generate a sentiment summary representing public positive and negative opinions on restaurants, we generally care more about some particular aspects such as *food* or *service* instead of overall sentiment. Such technique is referred to as *aspect-based sentiment summarization* [2-5].

TSA'09, November 6, 2009, Hong Kong, China.

An aspect-based sentiment summarization system takes as input a set of user reviews for a object (e.g. a restaurant) and produces a summary that expresses the aggregated sentiment for some relevant aspects (e.g. *food* or *service*), and supporting textual evidences [2]. There are two major problems in the standard aspect-based sentiment summarization, that is, *aspect mention extraction* and *sentiment classification*. The goal of aspect mention extraction is to extract all textual mentions associated with relevant aspects from reviews. Sentiment classification aims to identify the polarities (i.e. positive or negative) expressed in aspect mentions. An aspect-based sentiment summary looks like the following:

Restaurant-1:
Food aspect
Positive: <food aspect mentions>
Negative: <food aspect mentions>
Service aspect
Positive: <service aspect mentions>
Negative: <service aspect mentions>
… …

Figure 1: An aspect-based restaurant summary

Most modern aspect-based sentiment summarization systems generally use sentences associated with relevant aspects extracted from the reviews as the basis for summarization [2-5]. For example, a review sentence "*The quality of food is so so*" is identified as a negative food aspect mention.

However, from our collected 13,358 Chinese restaurant reviews (54,747 sentences in total), we find that about 20% of review sentences exhibit more than one aspect, namely *multi-aspect sentences*. For example, there is a multi-aspect review sentence "鱼很不错，羊肉口味一般，很贵。/*the fish is great, the taste of the lamb is so so, the food is very expensive*". This sentence expresses positive and negative on *food* aspect, and negative on *charge* aspect, respectively. In other words, this sentence contains three different aspect mentions, that is, a positive food aspect mention "鱼很不错/ *The fish is great*", a negative food aspect mention "羊肉口味一般/ *The taste of the lamb is so so*", and a negative charge aspect mention "很贵/ *The food is very expensive*". In this case, considering such sentence as a single aspect mention is problematic, because it is impossible to label such sentence with an appropriate aspect and its corresponding polarity.

As shown in Fig. 1, a reasonable aspect mention should be a single-aspect and single-polarity unit. As mentioned above, since in practice a single review sentence often exhibits several aspects for opinions, it raises an important and practical problem of how to split a multi-aspect sentence into multiple single-aspect and single-polarity units as candidate aspect mentions, namely *aspect-based sentence segmentation*. To date, the issue of aspect-based sentence segmentation is seldom mentioned in previous studies on aspect-based sentiment summarization.

In this paper, we focus on this segmentation issue, and propose an unsupervised segmentation model to segment multi-aspect review sentences. Experiments on real restaurant reviews show that our approach outperforms state-of-the-art linear text segmentation methods.

2. MOTIVATION

An aspect-based sentence segmentation system takes as input a sentence and produces multiple single-aspect and single-polarity units (i.e. segments). In this work, we study aspect-based sentence segmentation models on the sub-sentence[1] level. A segment might be a sub-sentence, or a combination of some consecutive sub-sentences.

Let $C=c_1c_2...c_n$ be a sentence consisting of n sub-sentences, and $U=u_1u_2...u_k$ be its segmentation consisting of k segments. Our goal is to find the most likely segmentation U^* of C. Our first intuition is to think of aspect-based sentence segmentation as the problem of linear text segmentation, by assuming that a sentence is a text, and an aspect is a subject.

The goal of linear text segmentation is to divide a text into homogeneous segments [6]. Each segment expresses a particular subject while contiguous segments exhibit different subjects. Due to the lack of space, we cannot describe state-of-the-art linear text segmentation algorithms in detail, and only analyze their effectiveness in the following evaluation experiments. To segment multi-aspect sentences, some state-of-the-art linear text segmentation techniques can be utilized, such as *dotplotting* [7][8], *C99* [9] and *Fragkou method* [10].

In practice, however, there are two challenges. First, as reported in previous studies [8-10], these state-of-the-art linear text segmentation methods work well at the document-level instead of the sentence-level. It seems a possible risk to apply linear text segmentation techniques for sentence-level segmentation, because one sentence cannot provide sufficient context to determine topic changes for the purpose of segmentation. Second, linear text segmentation techniques only consider topic change occurring in the input text. However, aspect-based sentence segmentation models should deal with both issues of aspect change and polarity change occurring in the sentence.

To overcome these challenges, it seems an appealing solution to incorporate aspect and polarity information of sub-sentences into the design of segmentation models. In other words, the aspect and the polarity information of each sub-sentence is very useful to implement aspect-based sentence segmentation. In Section 4, we present a two-stage segmentation model in which the aspect and polarity information conveyed by the input sentence are utilized.

In practice, since it is often expensive to build sufficient labeled training data to design supervised classifiers for aspect identification and polarity analysis, it is worthwhile studying how to use unsupervised or semi-supervised methods without requiring labeled training data. In this work, we employ bootstrapping methods to learn aspect-related terms for each aspect from unlabeled data (discussed in Section 3). These aspect-related terms are used for aspect identification. An extended sentiment lexicon is utilized for polarity analysis (discussed in Section 4.2).

3. ASPECT-RELATED TERM LEARNING

In this section, we apply bootstrapping methods to learn *aspect-related terms* (ARTs) of each aspect to be used for aspect identification. A bootstrapping method starts learning with a small number of seed ARTs under the help of unlabeled data [11]. Bootstrapping can be viewed as iterative clustering where in each learning cycle the most valuable candidate ART is chosen to augment the current seed set, and the learning procedure continues until the predefined stopping criterion is satisfied. In bootstrapping, we utilize the *RlogF* metric [12] to evaluate each candidate ART t by

$$\text{RlogF}(t) = \log frq(t,T) \times R(t,T), \quad (1)$$

where T is the current seed set, $frq(t,T)$ denotes the frequency of co-occurrence of t and T within a limited context (i.e., l words to left or right of t), $frq(t)$ denotes the frequency of occurrence of t in the corpus, and $R(t,T)=frq(t,T)/frq(t)$.

In this study, we consider two different types of ARTs. First, like word-type features used in previous studies [13-15], *nouns*, *verbs*, *adjectives* and *adverbs* are considered as the first type of candidate ARTs for bootstrapping learning. Second, some previous studies [16][17] reported that higher-order n-grams such as bigrams and trigrams in some sentiment analysis settings outperform unigrams. It motivates us to consider *multi-word terms* as the second type of ARTs.

To extract multi-word terms from unlabeled reviews, we utilize the *C-value* method[2] [18] which takes as input a review set and produces a list of multi-word terms ranked in the descending order of *C-value score*. The C-value score of a multi-word term t can be calculated as follows [18]:

- If t is not contained by any other terms

$$\text{C-value(t)} = \log(|t|) \times frq(t),$$

- Otherwise

$$\text{C-value(t)} = \log(|t|)\left(frq(t) - \frac{1}{n(L)}\sum_{l \in L} frq(l)\right)$$

where $|t|$ denotes the number of words contained by t, $frq(t)$ indicates the frequency of occurrence of t in the corpus, L is the set of multi-word terms containing t, and $n(L)$ denotes the number of terms in L.

[1] The separation mark between two adjacent sub-sentences is defined as a comma or a semicolon in a sentence.

[2] The linguistic filter used by C-value method [18] is described as $(noun|verb|adjective|adverb)^*$.

The algorithm of bootstrapping-based ART learning is summarized as follows:

Algorithm: Bootstrapping-based ART Learning
Input: the initial aspect seed sets $S=\{S_1, S_2, ..., S_m\}$ for m aspects, and a pool of unlabeled data U
Stage 1: Candidate ART Extraction
Extract nouns, verbs, adjectives, adverbs and top-n multi-word terms recognized by C-value method from U to form a candidate ART set Ω for bootstrapping.
Stage 2: **Bootstrapping Learning**
Start learning with the seed set S_i for the i^{th} aspect
Repeat
1. Use Equation (1) to calculate *RlogF* score of each candidate in Ω;
2. Select the candidate with the highest *RlogF* score to augment S_i, and remove it from Ω;

Until the predefined stopping criterion[3] is met.
Output: Final ART sets S for m aspects.

Figure 2. The bootstrapping learning algorithm

For the purpose of aspect identification using these learned ART sets, we need to assign each learned ART with an importance score that indicates the degree of its ability to reflect its corresponding aspect. Our first intuition is to use their RLogF values. But this is problematic because as shown in Equation (1), the RlogF value of each candidate ART is estimated using the current seed set T which is changing dynamically during the bootstrapping procedure. In such case, the RLogF values of two terms learned in different learning cycles cannot be compared with each other. To estimate the importance degree of a learned ART for each aspect, we consider two factors given as follows:

1) Bootstrapping methods tend to learn the most valuable ARTs at the earlier learning iterations [11]. Based on this assumption, the *rank degree* $\eta_i(t)$ of an ART t for the i^{th} aspect can be measured by means of a function of the rank of t in S_i as

$$\eta_i(t) = 1 - \frac{r_i(t)}{|S_i|}. \quad (2)$$

where $S_i=\{t_{i1}, t_{i2}, ..., t_{ik}\}$ is the ART set of the i^{th} aspect produced by bootstrapping. Notice that t_{ij} is learned in the j^{th} iteration, $|S_i|$ indicates the number of ARTs in S_i, and $r_i(t)$ represents the rank of t in S_i, indicating in which iteration it was learned. A higher $\eta_i(t)$ value indicates that t is a more important ART for the i^{th} aspect.

2) From the bootstrapping results, we find that many ARTs belong to more than one aspect, named *multi-aspect ARTs*. For the purpose of aspect identification based on these learned ART sets, we prefer single-aspect ARTs to multi-aspect ARTs. In this case, multi-aspect ARTs are considered to be ambiguous. The *ambiguity degree* $\varphi(t)$ of a multi-aspect ART t can be measured by means of an entropy-like function of ranks of t in m learned ART sets as

$$\varphi(t) = \frac{-\sum_{i=1}^{m} \frac{r_i(t)}{\sum_{1\le j\le m} r_j(t)} \log \frac{r_i(t)}{\sum_{1\le j\le m} r_j(t)}}{\log m} \quad (3)$$

where $S=\{S_1, S_2, ..., S_m\}$ represents ART sets for m aspects. The denominator[4] is used for normalization. A higher $\varphi(t)$ value indicates that the ART t is more ambiguous.

We prefer to select ARTs with highest rank degree and lowest ambiguity degree, which will be of high value for aspect identification. Based on this assumption, by considering rank degree and ambiguity degree simultaneously, the importance score $\psi_i(t)$ of an ART t for the i^{th} aspect can be calculated by

$$\psi_i(t) = \eta_i(t) \times (1 - \varphi(t)) \quad (4)$$

A higher $\psi_i(t)$ value indicates that t is a more important ART for the i^{th} aspect.

4. UNSUPERVISED SEGMENTATION MODEL

We formulate an aspect-based sentence segmentation model by introducing a criterion function $J(.)$ that aims to evaluate each candidate segmentation U of the sentence C, that is

$$U^* \overset{def}{=} \arg\max_U J(C,U) \quad (5)$$

The goal of this model is to find the most likely segmentation U^* with maximum $J(.)$ score. In the most likely segmentation U^*, two adjacent segments express different aspects or the same aspect with different polarities, and each segment is constrained to have only one aspect and one polarity. To design an appropriate criterion function $J(C,U)$, we need to consider three factors indicating: 1) what the aspect of each segment in U is, 2) what the polarity of each segment in U is, and 3) whether any two adjacent segments in U express different aspects or the same aspect with different polarities.

In the most likely segmentation U^*, there are two different cases: 1) two adjacent segments express different aspects, and 2) two adjacent segments express the same aspect with different polarities. To deal with both cases, we think it is difficult to deal with both issues of aspect change and polarity change simultaneously under a simple framework for aspect-based sentence segmentation.

To solve this problem, we propose a *two-stage segmentation framework* in which the issues of aspect change and polarity change are handled separately. In this case, it naturally raises a question about which issue we prefer to deal with firstly in this framework. As mentioned above, our segmentation model aims to handle sentences that express several aspects or the same aspect with different polarities. From our collected 13,358 Chinese restaurant reviews, we find that most sentences needed to be

[3]The bootstrapping learning can end when a desirable number of ARTs for each aspect have been learned.

[4] $\log(m)$ is the maximum entropy of probability distribution over m classes.

segmented belong to the multi-aspect cases. In such case, we prefer to first handle the issue of aspect change in our two-stage segmentation framework.

In our framework, the first stage aims to identify multiple single-aspect segments in one sentence by determining the aspect change occurring in the input sentence, regardless of polarity change. If a resulting single-aspect segment expresses several polarities, we should further re-segment it into multiple single-polarity units in the second stage. Finally, the most likely segmentation U^* of C can be generated by combining the segmentation results of both stages.

4.1 The First Stage

In the first stage, to segment the input sentence C into multiple single-aspect segments, the first issue we need to consider is what aspect information each sub-sentence conveys, namely aspect identification. In this work, aspect identification of a textual unit[5] is implemented by using ART sets $S=\{S_1, S_2, ..., S_m\}$ for m aspects produced by bootstrapping methods. The j^{th} aspect score of a textual unit ξ is computed by summing the importance scores of all ARTs of the j^{th} aspect in this unit, that is

$$\Phi_j(\xi) = \sum_{t\in\xi, t\in S_j} \psi_j(t) \quad (6)$$

where t is an ART. The most likely aspect j^* of the textual unit ξ is given by

$$j^* = \arg\max_j \Phi_j(\xi) \quad (7)$$

To determine whether two adjacent segments express different aspects, we adopt an aspect indicator function $\delta_A(u_i,u_j)$ whose value is 1 if segments u_i and u_j are labeled as two different aspects, and 0 otherwise.

We can employ the model defined in Equation (5) to implement the first stage, in which the criterion function $J(C,U)$ is designed by incorporating two factors indicating what the aspect of each segment is, and whether two adjacent segments express different aspects, that is

$$J(C,U) = \sum_{1\le i\le k} \delta_A(u_{i-1},u_i) \times \Phi_{j^*}(u_i) \quad (8)$$

where $\delta_A(u_0,u_1)$ is assumed to be 1. The second term in the summation denotes the aspect score of the most likely aspect of a candidate segment u_i. To find the most likely segmentation U^* with maximum $J(.)$ score, we can utilize *dynamic programming* (DP) algorithm.

4.2 The Second Stage

If a single-aspect segment produced in the first stage expresses several polarities, in the second stage we apply a simple algorithm to further re-segment it in terms of polarity change. To determine whether two adjacent segments express different polarities, we adopt a polarity indicator function $\delta_P(u_i,u_j)$. The value of $\delta_P(u_i,u_j)$ is 1 if segments u_i and u_j are labeled as two different polarities, and 0 otherwise. The re-segmentation algorithm used in the second stage is summarized as follows:

Input: a segment $u=\{c_1, c_2, ..., c_m\}$ produced in the first stage. c_i is the i^{th} sub-sentence of u, $m>1$.
Steps:
FOR i =1 TO m-1 DO
  IF $\delta_P(c_i,c_{i+1})$ is equal to 1 THEN
    Mark a segment boundary between c_i and c_{i+1}.
  ENDIF
ENDFOR
Output: the final segmentation of u generated based on the segment boundary marks

Figure 3. A re-segmentation algorithm used in the second stage.

The core of this re-segmentation algorithm used in the second stage is to analyze the polarity of each sub-sentence. When labeled training data is not provided, sentiment-lexicon-based methods seem to be a feasible choice for polarity analysis. We use a Chinese sentiment lexicon[6] released by *Hownet* [19]. This sentiment lexicon contains 3730 positive words and 3116 negative words. Since the sentiment lexicon released by Hownet is a general-purpose knowledge base, using only this sentiment lexicon will yield unsatisfactory performance in a specific domain (e.g. restaurants). To improve the lexicon-based method for polarity analysis, we adopt a simple method for the purpose of domain adaptation on this sentiment lexicon. Adjectives are good subjectivity indicators in the restaurant domain. We first extracted all adjectives from the unlabeled restaurant reviews, and manually labeled their polarities (i.e. positive, negative or neutral). All adjectives[7] associated with a positive or negative polarity were added into our extended sentiment lexicon. Thirteen negation words such as "不/not" are used in handling negation issue.

We adopt a sentiment-lexicon-based method [20] for polarity analysis, in which the semantic orientation value of a textual unit is computed by summing the polarity values of all sentiment words in the unit. If a sentiment word w is negated, it is converted into a new token "*NOT-w*" associated with an opposite polarity. The polarity value is empirically set to +1 for a positive word, and -1 for a negative word. The resulting semantic orientation value of a textual unit indicates its corresponding polarity, that is, >0 for positive, <0 for negative, and equals to 0 for neutral, as in [20].

5. EXPERIMENTAL SET-UP

5.1 Evaluation Datasets

We evaluate our aspect-based sentence segmentation method on a corpus of Chinese restaurant reviews taken from the website *DianPing.com*, which contains 13,358 reviews (54,747 sentences in total) for 100 restaurants. In the preprocessing step we utilized the NEUCSP[8] tool to implement Chinese word segmentation and POS tagging.

[5] The term *textual unit* we used here can represent a text, a sentence, a segment, or a sub-sentence.

[6] http://www.keenage.com/html/c_index.html

[7] In this study, 910 adjectives have been manually added in our extended sentiment lexicon.

[8] NEUCSP is a Chinese word segmentation and POS tagging tool at (http://www.nlplab.com/chinese/source.htm)

We randomly select 1000 review sentences for testing (TEST) and 500 review sentences as development set (DEV). The rest of review sentences (53,247 sentences in total) were used as unlabeled data for bootstrapping-based ART learning. To form the gold standard, two human judges were asked to segment each review sentence in the TEST and DEV sets in terms of aspect change (i.e. *environment, favorable policy, food, charge* and *service* aspects) and polarity change (i.e. positive and negative). That is, each multi-aspect or single-aspect and multi-polarity sentence was manually segmented into multiple single-aspect and single-polarity units, and a single-aspect and single-polarity sentence is considered as a single segment. The TEST and DEV sets were annotated separately to verify the inter-annotator agreement and to verify whether the task is well-defined. The inter-annotator agreement is 87.5%. For the disagreements between the judges, a third human judge acted as an adjudicator. The data statistics of the TEST set is shown in Table 1. The average number of segments per sentence is 1.54 in the TEST set.

Table 1. The numbers of three types of sentences in the TEST set.

Types of Sentences	Number of Sentences (%)
Multi-aspect or single-aspect and multi-polarity	317 (31.7%)
Single-aspect and single-polarity	427 (42.7%)
Others	256 (25.6%)
Total	1000 (100%)

5.2 Evaluation Criteria

The traditional *precision*, *recall* and *F1* metrics are used to evaluate the effectiveness of each automatic segmentation methods. However, the shortcoming of precision and recall for segmentation is that every inaccurately estimated segment boundary is penalized equally whether it is near or far from a true segment boundary. In the following experiments, we also adopt the *WindowDiff* metric [21] which has been widely used in text segmentation community. The WindowDiff metric is defined by [21]

$$WindowDiff(ref, hyp) = \frac{1}{N-k}\sum_{i=1}^{N-k}(\left|b(ref_i, ref_{i+k}) - b(hyp_i, hyp_{i+k})\right| > 0)$$

where *ref* and *hyp* represent the reference segmentation and a hypothesized segment. *b(i,j)* denotes the number of segmented boundaries between positions *i* and *j* in the sentence, and *N* denotes the number of sub-sentences in the sentence. The *k* value used in the WindowDiff metric is set to 3, that is, the average number of sub-sentences per segment in the gold standard.

5.3 Bootstrapping Learning Settings

In C-value method, we consider *bigrams*, *trigrams* and *4-grams* to be candidate multi-word terms. From unlabeled data we first extracted nouns, verbs, adjectives and adverbs, and used C-value method to select top-40,000 multi-word terms (ranked in the descending order of C-value score) to form the candidate ART set for bootstrapping. The *l* value of the limited context used in Equation (1) was empirically set to 5. In the bootstrapping-based ART learning algorithms, five seeds were initially provided for each aspect as follows:

Table 2. Initial aspect seed sets for bootstrapping

Aspect	*Seeds*
Environment	环境/*environment*, 豪华/*luxury*, 装修/*decoration*,嘈杂/*noisy*, 吵闹/*noisy*
Favorable Policy	打折/*discount*, 免费/ *free*, 赠券/*coupon* 优惠/*on sale*, 赠送/ *gift and rebate*
Food	食物/*food*, 口味/*taste*, 油腻/*oily,* 好吃/*delicious*, 正宗/*authentic*
Charge	价格/*price*, 贵/*expensive*, 便宜/*cheap*, 买单/*pay the bill*, 性价比/*cost performance*
Service	服务员/waiter, 体贴/*considerate*, 服务/service,周到/*good service* 热情/*friendly*

In the bootstrapping results, all learned ARTs for each aspect are ranked in the descending order of their importance degrees *score(.)*. To evaluate the effectiveness of each bootstrapping-based ART learning algorithm, we manually check top-500 learned ARTs for each aspect, and report the corresponding precision performances in the following Fig. 4.

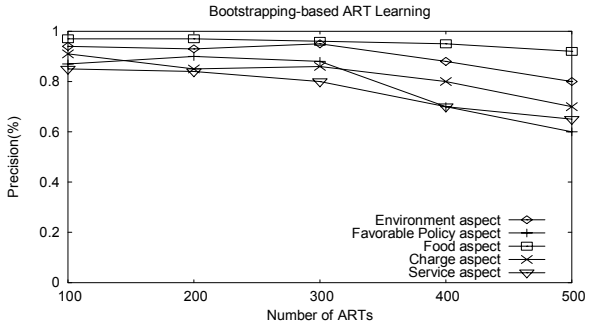

Figure 4. Precision performance of the top-500 ARTs for each aspect learned by bootstrapping methods.

Fig. 4 shows that the precision of the top-300 learned ARTs of each aspect is larger than 80%. For the top-500 ARTs, the bootstrapping algorithms work the best for the *food* aspect, and the worst for the *favorable policy* aspect. It is reasonable that customers generally comment on the *food* aspect of a specific restaurant more than on the *favorable policy* aspect. There are more food-related terms expressed in the restaurant reviews, compared to the favorable policy in general. From experimental results of the favorable policy aspect, we found that its most valuable ARTs have been learned before the 300th learning iteration, and many noisy terms were learned during the later learning iterations.

In the bootstrapping learning experiments, the learning process stops when 10000 ARTs for each aspect have been learned. We used the development data set to determine optimal numbers of ARTs of each aspect used for our two-stage segmentation model. Experimental results show our segmentation model achieves the best performance when using 2000 ARTs for each aspect.

6. RESULTS

In this experiment we constructed two baseline methods. The first baseline method is to segment a sentence in terms of comma, named *comma-based method*. That is, each sub-sentence is viewed as a single segment. The second baseline method is to simply consider the whole sentence as a single single-aspect segment, named *full-stop-based method*. In addition, we evaluate the effectiveness of three state-of-the-art linear text segmentation methods[9] such as *Dotplotting*, *C99* and *Fragkou method*, and our two-stage segmentation model as follows:

Table 3. Precision (P), Recall (R) and F1 performances of each method for aspect-based sentence segmentation. Each bold number denotes the best performance.

Methods	P	R	F1
Full-stop-based method	0.68	0.44	0.54
Comma-based method	0.17	0.38	0.24
Dotplotting	0.19	0.39	0.25
C99	0.65	0.44	0.53
Fragkou method	0.45	0.46	0.45
Two-stage segmentation model	**0.69**	**0.56**	**0.62**

Table 4. WindowDiff values of different methods for aspect-based sentence segmentation. The bold number denotes the best performance.

Methods	WindowDiff
Full-stop-based method	0.21
Comma-based method	0.72
Dotplotting	0.69
C99	0.21
Fragkou method	0.26
Two-stage segmentation model	**0.17**

Tables 3 and 4 show the effectiveness of each automatic method for aspect-based sentence segmentation. The smaller the WindowDiff value is, the better the segmentation performance is. Among these six automatic methods, the comma-based method is the worst. As mentioned in Table 1, 68.3% of review sentences in the TEST set are not multi-aspect or single-aspect and multi-polarity cases. The comma-based method often makes wrong segmentations on these sentences.

Our two-stage segmentation model achieves the best performance in terms of all evaluation metrics, followed by the C99 method. However, C99 obtains 44% recall performance that is lower than that of Fragkou method. There are two possible reasons why our segmentation method outperforms these state-of-the-art linear text segmentation methods. First, as mentioned above, for aspect-based sentence segmentation, one sentence cannot provide sufficient context for traditional linear text segmentation techniques to determine topic changes occurring in the input sentence. Second, our model explicitly utilizes aspect and polarity knowledge such as ARTs and an extended sentiment lexicon in the process of sentence segmentation while linear text segmentation methods do not consider any aspect and polarity information expressed in the sentence.

It surprises us that the dotplotting method obtains unsatisfactory performances, and its performance is very close to that of the comma-based method. The dotplotting method adopts local (i.e. between adjacent sub-sentences) similarity measures based on word repetitions [8]. From segmentation results, we find that many single-aspect and single-polarity sentences have been wrongly segmented by dotplotting. In most review sentences, there are few common words occurring in two adjacent sub-sentences. In such case, the dotplotting tends to split individual sentences at commas. Experimental results show that the dotplotting method fails to identify most true segments consisting of multiple consecutive sub-sentences.

In contrast with the dotplotting, the C99 and Fragkou method fail to segment most multi-aspect sentences due to lack of sufficient useful context to determine the topic change, and tend to take no action on most review sentences. Compared to the dotplotting, the C99 and Fragkou method work well in aspect-based sentence segmentation, because only 31.7% review sentences in the TEST set need to be segmented. That is the possible reason why the C99 and Fragkou method outperform the dotplotting and the comma-based method. For the same reason, compared to the comma-based method, the full-stop-based method obtains a good performance but lower than that of our two-stage segmentation model.

7. DISCUSSION

It is worth mentioning that our model can be applied to aspect-based English language sentence segmentation if one of the readily available English sentiment lexicons such as *SentiWordNet*[10] is used for polarity analysis.

From experimental results, we find that our two-stage segmentation model fails to handle the sentences containing multi-aspect sub-sentences in the first stage, because our model works on sub-sentence level, in which a sub-sentence is viewed as the basic unit for aspect-based sentence segmentation. Actually such case occurs very little in our collected restaurant reviews. To solve this problem, we think it is still worthwhile to further study our segmentation model on word-level in the future work.

To improve the performance of the lexicon-based method for polarity analysis in the restaurant domain, we manually extend the HowNet sentiment lexicon. Actually domain adaptation is very important for opinion analysis [22][16]. When labeled data is not provided, the lexicon-based method is a feasible choice. Since most readily available sentiment lexicons are general-purpose knowledge bases, it is worth studying how to automatically transfer a general-purpose sentiment lexicon to real domain applications to achieve better performance. In this case, there are at least three crucial issues to be considered. First, the same sentiment term might indicate different polarity in different domains. Second, difference in sentiment vocabularies across different domains should be considered. The third issue is how to assign a strength marker to each sentiment word.

[9] We used the package developed by Choi to implement Dotplotting and C99 algorithms (see www.lingware.co.uk/homepage/freddy.choi/index.htm), and implemented the Fragkou method in our experiments. Our implementation of Fragkou method achieves the same performance on Choi's evaluation text collection as reported in [10]

[10] http://sentiwordnet.isti.cnr.it/.

8. RELATED WORK

Aspect-based sentiment summarization [2-5] aims to produce a summary expressing the aggregated sentiment for each aspect and supporting textual evidences in the form of individual sentences. In the related area of opinion extraction from user reviews, some previous efforts have focused on the extraction of opinion topic [23][2] that is limited to extract the mentions of product names and their features. Kim and Hovy (2006) presented a technique for the extraction of opinion topic based on semantic frames, and provided a limited evaluation. However, in these previous efforts, there is little work on the issue of segmentation on multi-aspect sentences for aspect mention extraction or opinion topic extraction. In this paper, we address the issue of aspect-based sentence segmentation, and present a two-stage segmentation model to identify multiple single-aspect and single-polarity units in one sentence.

Titov and McDonald (2008) applied a multi-aspect sentiment model to learn aspect-related words for each aspect from unlabeled data. However, their method is limited to reviews with user provided aspect ratings. Some researchers [15][26][27] have applied bootstrapping method for learning subjective words or sentiment patterns. In our work, we consider both lexical words and multi-word terms as candidate aspect-related terms, and apply bootstrapping methods to learn aspect-related terms of each aspect from unlabeled reviews without aspect ratings. A new scoring method is applied to assign the importance score to each aspect-related term based on two factors, that is, rank degree and ambiguity degree.

Text segmentation is an important problem in information retrieval. Previous studies [8][9][10][6] generally focused on text segmentation at document-level instead of sentence-level. In this paper, some experiments have been designed to investigate the effectiveness of state-of-the-art linear text segmentation methods in the task of aspect-based sentence segmentation.

9. CONCLUSION AND FUTURE WORK

This paper addresses the issue of segmentation on multi-aspect sentences for aspect-based sentiment summarization, and proposes a two-stage segmentation framework to address the challenge of identifying multiple single-aspect and single-polarity units in one sentence. Our segmentation model deals with both issues of aspect change and polarity change occurring in the input sentence. In the implementation, aspect-related terms of each aspect are learned by bootstrapping methods from unlabeled data. These learned aspect-related terms are used for aspect identification. An extended sentiment lexicon is adopted for polarity analysis. Experiments on real restaurant reviews show that our two-stage segmentation model outperforms state-of-the-art linear text segmentation methods. In the future work, we are interested in sentiment lexicon domain adaptation for sentiment analysis, and applying our method for aspect-based sentiment summarization.

10. ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation of China (60873091).

11. REFERENCES

[1] Pang B. and Lee L. 2008. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, Vol. 2, Nos. 1-2(2008) 1-135

[2] Hu M. and Liu B. 2004. Mining and summarizing customer reviews. In Proceedings of the 2004 ACM SIGKDD international conference on knowledge discovery and data mining, pp168-177.

[3] Carenini G., Ng R., and Pauls A. 2006. Multi-document summarization of evaluative text. In Proceedings of 11st Conference of the European Chapter of the Association for Computational Linguistics, pp305-312

[4] Zhuang L., Jing F. and Zhu X. Y. 2006. Movie review mining and summarization. In Proceedings of the 15th ACM international conference on information and knowledge management, pp43-50

[5] Feiguina O. and Lapalme G. 2007. Query-based summarization of customer reviews. In Proceedings of Canadian Conference on AI 2007, pp.452-463

[6] Hearst, M.A. 1997. TextTiling: segmenting text into multi-paragraph subtopic passages. Computational Linguistics, 23(1) p33-64

[7] Church K. W. 1993. Chart Align: A program for aligning parallel texts at the character level. In Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics, pp1-8

[8] Reynar J.C. 1994. An automatic method of finding topic boundaries. In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, pp331-333

[9] Choi F. Y. Y. 2000. Advances in domain independent linear text segmentation. In Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics, pp26-33

[10] Fragkou P., Petridis V., and Kehagias Ath. 2004. A dynamic programming algorithm for liner text segmentation. Journal of Intelligent Information System, 23(2):179-197.

[11] Yarowsky D. 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics, pp.189-196.

[12] Riloff E. and Jones R. 1999. Learning dictionaries for information extraction by multi-Level bootstrapping, Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99)

[13] Hatzivassiloglou, V. and McKeown K. 1997. Predicting the semantic orientation of adjectives. In Proc. of ACL-EACL 97

[14] Pang, B. Lee L., and Vaithyanathan S. 2002. Thumbs up? Sentiment classification using machine learning techniques. In proc. of EMNLP02

[15] Riloff, E., Wiebe J., and Wilson T. 2003. Learning subjective nouns using extraction pattern Bootstrapping. In Prof. of CoNLL-03

[16] Dave K., Lawrence S., and Pennock D. M. 2003. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In Proceedings of WWW, pp519-528

[17] Riloff E., Patwardhan S., and Wiebe J. 2006. Feature subsumption for opinion analysis. In Proceedings of EMNLP06, pp.440-448

[18] Frantzi K., Ananiadou S., Mima H. 2000. Automatic recognition of multi-word terms: the C-value/NC-value method. International Journal on Digital Libraries, p115-130

[19] Dong Z. and Dong Q. 2006. Hownet and the computation of meaning. World Scientific Publishing Co., Inc.

[20] Wan X. 2008. Using bilingual knowledge and ensemble techniques for unsupervised Chinese sentiment analysis. In Proceedings of EMNLP08, pp553-561

[21] Pevzner L. and Hearst M. A. 2002. A critique and improvement of an evaluation metric for text segmentation. Computational Linguistics, 28(1): 19-35

[22] Aue A. and Gamon M. 2005. Customizing sentiment classifiers to new domains: a case study. In Proceedings of recent advances in natural language processing (RANLP).

[23] Popescu A. M. and Etzioni O. 2005. Extracting product features and opinions from reviews. In Proceedings of the conference on empirical methods in natural language processing.

[24] Kim S. and Hovy E. 2006. Extracting opinions, opinion holders, and topics expressed in online news media text. In Proceedings of ACL/Coling Workshop on Sentiment and Subjectivity in Text, pp1-8

[25] Titov I. and McDonald R. 2008. A joint model of text and aspect ratings for sentiment summarization. In Proceedings of ACL-08, pp308-316.

[26] Wang B. and Wang H. 2007. Bootstrapping both Product Properties and Opinion Words from Chinese Reviews with Cross-Training. IEEE/WIC/ ACM International Conference on Web Intelligence (WI'07), pp.259-262

[27] Zagibalov T., and Carroll J. 2008. Unsupervised Classification of Sentiment and Objectivity in Chinese Text. In Proceedings of Proceedings of the Third International Joint Conference on Natural Language Processing, pp304-311