Sentiment Extraction by Leveraging Aspect-Opinion Association Structure

Li Zhao, Minlie Huang, Jiashen Sun*, Hengliang Luo*, Xiankai Yang, Xiaoyan Zhu
State Key Lab. of Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Dept. of Computer Science and Technology, Tsinghua University, Beijing 100084, PR China
*Samsung R&D Institute China - Beijing
zhaoli19881113@126.com aihuang@tsinghua.edu.cn

ABSTRACT

Sentiment extraction aims to extract and group aspect and opinion words from online reviews. Previous works usually extract aspect and opinion words by leveraging association between a single pair of aspect and opinion word[5] [14] [9] [4][11], but the structure of aspect and opinion word clusters has not been fully exploited.

In this paper, we investigate the *aspect-opinion association structure*, and propose a "first clustering, then extracting" unsupervised model to leverage properties of the structure for sentiment extraction. For the clustering purpose, we formalise a novel concept *syntactic distribution consistency* as soft constraint in the framework of posterior regularization; for the extraction purpose, we extract aspect and opinion words based on cluster-cluster association. In comparison to traditional word-word association, we show that cluster-cluster association is a much stronger signal to distinguish aspect (opinion) words from non-aspect (non-opinion) words. Extensive experiments demonstrate the effectiveness of the proposed approach and the advantages against state-of-the-art baselines.

Categories and Subject Descriptors

I.2.7 [**Natural language processing—Text Analysis**]

Keywords

Sentiment Analysis; Opinion Mining; Sentiment Extraction; Information Extraction

1. INTRODUCTION

Sentiment extraction is the task of extracting and grouping aspect words and opinion words from online reviews. This task is important because, on top of extracted aspects and opinions, we can aggregate various opinions according to a product's aspects (or attributes), and provide much detailed, complete, and in-depth summaries of a large number of reviews. More specifically, aspect words refer to a product's or service's properties (or attributes) which people have expressed opinion upon, such as *battery, screen*, and so on. While opinion words are words that people use to express sentiment, such as *amazing*, and *interesting*. Since aspect/opinion is usually expressed by different synonymous aspect/opinion words, we need to group aspect/opinion words into clusters.

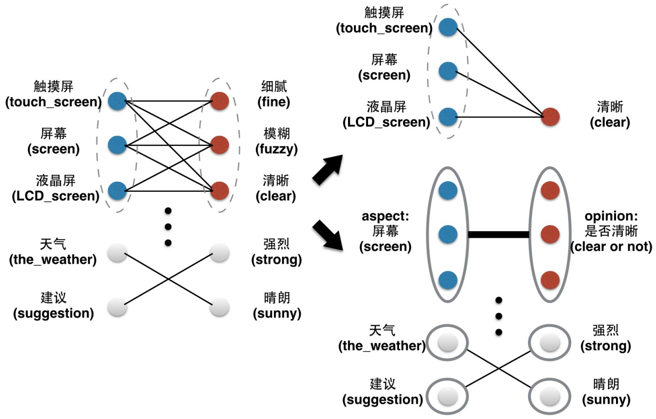


Figure 1: Aspect-opinion association structure. As shown in the figure, "touch-screen", "screen" and "LCD-screen" are synonymous aspect words, while "fine", "fuzzy" and "clear" are synonymous/antonymous opinion words. The rest words are non-aspect/non-opinion words. An edge represents the association between a pair of aspect and opinion word candidate. The picture shows that synonymous aspect words("touch-screen", "screen" and "LCD-screen") are modified by the same opinion word "clear", and vice versa; and that, the associations between aspect and opinion clusters are stronger and less prone to noises than word level associations.

Previous works for sentiment extraction focus on modeling the association between an aspect word and an opinion word [5] [14] [9] [4][11]. The central assumption lies in that, if a word is likely to be an opinion word, the words which has strong association with the word will be more possible to be aspect words, and vice versa. The word-word association has been modeled by nearest neighbor rule [5], manually designed dependency pattern[14], word alignment model [9], and statistical correlation such as likelihood ratio test[4]. Despite the success of previous works, they may discover some "False Opinion Relations"(false word-word as-

CIKM'15, October 19–23, 2015, Melbourne, Australia.
© 2015 ACM. ISBN 978-1-4503-3794-6/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2806416.2806525.

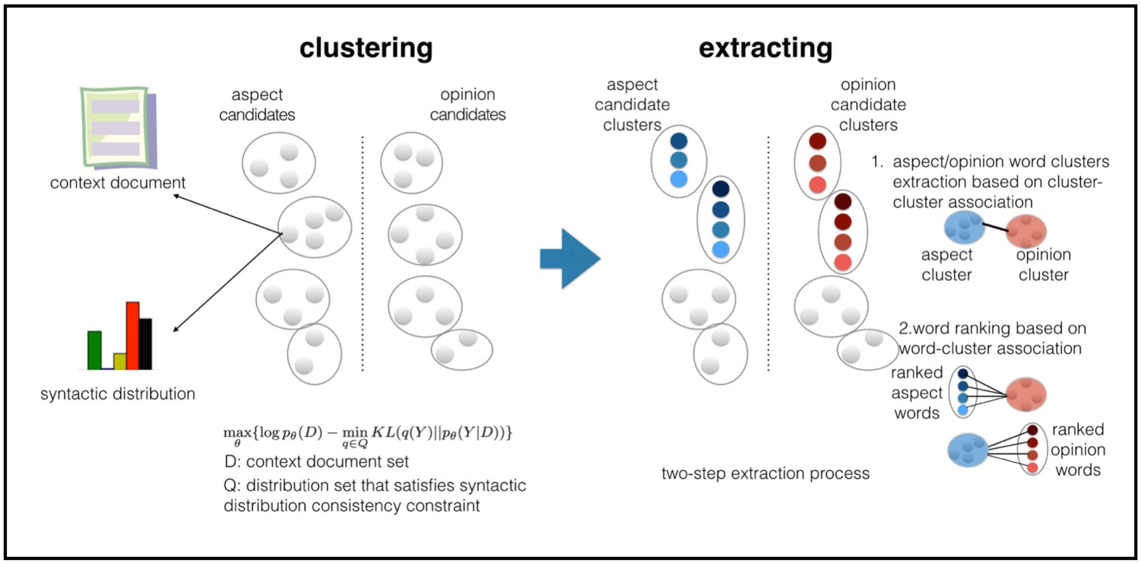

clustering
extracting
context document
syntactic distribution
aspect candidates
opinion candidates
aspect candidate clusters
opinion candidate clusters
1. aspect/opinion word clusters extraction based on cluster-cluster association
aspect cluster
opinion cluster
2.word ranking based on word-cluster association
ranked aspect words
ranked opinion words
$\max_{\theta}\{\log p_\theta(D) - \min_{q\in Q} KL(q(Y)||p_\theta(Y|D))\}$
D: context document set
Q: distribution set that satisfies syntactic distribution consistency constraint
two-step extraction process

Figure 2: The framework of our model. Our model consists of two components: clustering and extracting. Firstly, aspect/opinion word candidates are clustered based on both context information and syntactic distribution. Secondly, we extract aspect/opinion word clusters based on cluster-cluster association. For each extracted cluster, we further rank each word based on word-cluster association. Points with darker color represent words with higher confidence to be aspect/opinion words. Blue/red denotes aspect/opinion, respectively.

sociations) as discussed in [16]. For example, the phrase "autofocus camera" can be matched by a dependency pattern "Adj-amod-Noun", which is widely used in previous works[16][14]. But this phrase doesn't bear any sentiment orientation. The mined association between "autofocus" and "camera" is a false word-word association. Besides, we also find that previous works cannot fully employ the structure of the constructed graph, which is actually very useful for both extracting and clustering as we will show later.

Aspect-opinion association structure(Fig. 1) refers to the structure that the bipartite graph is particularly dense, nearly complete with regard to an associated pair of aspect and opinion. In other words, considering the subgraph which contains an associated pair of aspect and opinion word cluster, the bipartite subgraph is very dense and nearly complete. As shown in Fig.1, consider all the aspect words in aspect cluster "screen", and all the opinion words in opinion cluster "clear(or not)". Since most synonymous words can be used exchangeably, we observe that there is a very dense connection structure between words in aspect cluster "screen" and words in opinion cluster "clear(or not)".

From the perspective of sentiment extraction, *aspect-opinion association structure* preserves two properties: 1. Synonymous aspect words are likely to be modified by the same opinion word, and vice versa. 2. The associations between aspect and opinion clusters are stronger and less prone to noises than word-word association.

The first property is useful for clustering aspect/opinion word candidates. This property leads to our *syntactic distribution consistency* assumption, assuming that *synonymous aspect words are likely to be connected to the same opinion word through same dependency relation.* This assumption is further formulated as constraint to guide the clustering process. Our clustering model is especially useful for distinguishing words with similar contexts but different syntactic distributions. For example, "market" and "price" are somehow semantic-related and may have similar context. But they are modified by different opinion word candidates. With the help of syntactic distribution, we can assign them to different clusters. As a result, our clustering model can produce more pure clusters.

Inspired by the second property, we extract aspect and opinion by leveraging the association between an aspect candidate cluster and an opinion candidate cluster. Strongly associated cluster pairs are extracted as pairs of aspect and opinion. The extraction rule is illustrated as follows. As mentioned above, the resultant clusters can boost the association between an aspect word and an opinion word by exploiting cluster-cluster association. Now consider the non-aspect/non-opinion words, such as "suggestion" and "strong". Non-aspect/non-opinion words usually don't have that much synonymous words as aspect/opinion words do. Ideally, many non-aspect/non-opinion clusters would contain few words (as shown in Fig. 1). Thus the clustering process will not enhance the association of those cluster pairs. Therefore, the association between a pair of aspect and opinion is much stronger than that of non-aspect/non-opinion pairs.

In practice, the non-aspect/non-opinion clusters usually contains several words. But our assumption still holds well, and this is verified by our clustering results. The association between a pair of aspect cluster(with more than 10 aspect words) and opinion cluster(with more than 10 opinion words) is 6.8 times as much as the association between other cluster pairs on average.

To leverage the two properties of *aspect-opinion association structure*, we propose a "first clustering, then extracting" unsupervised model for sentiment extraction. Firstly, we cluster all aspect and opinion word candidates based on both context information and syntactic information. Secondly, we extract aspect and opinion by constructing a graph to model association between aspect and opinion candidate clusters.

Our contributions are listed as follows:

- We propose a "first clustering, then extracting" model for sentiment extraction by leveraging properties of *aspect-opinion association structure*.

- For the clustering component, we propose *syntactic distribution consistency* assumption and formulate this assumption as data-driven constraint. This constraint is further integrated into a context-based probabilistic model to guide the clustering process.

- For the extraction component, we propose a novel ranking algorithm by leveraging cluster-cluster association. Unlike word-word association, cluster-cluster association is a much stronger signal to distinguish aspect (opinion) words from non-aspect (non-opinion) words. As a result, our extraction method can promote the extraction performance.

The rest of this paper is organized as follows. We introduce the framework of our methodology in Section 2, along with the clustering model in Section 2.1, and the extraction model in Section 2.2. We present experiment results in Section 3. In Section 4, we survey related work. We summarize our work in Section 5.

2. METHODOLOGY

To extract and cluster aspect and opinion words from reviews, we propose a "first clustering, then extracting" framework, as shown in Fig. 2. Our model consists of two components as follows:

1) The Clustering component(Section 2.1) aims to cluster aspect/opinion candidates separately into word clusters. In this paper, we select nouns/adjectives to be aspect/opinion candidates. We cluster those candidates by leveraging both context information and syntactic information.

2) The Extracting component(Section 2.2) aims to extract aspect/opinion clusters from all candidate clusters. For each extracted cluster, we further rank each word of the cluster to find aspect or opinion words. We extract aspect/opinion clusters based on cluster-cluster association, and rank words in a cluster based on word-cluster association.

2.1 Clustering Based on Syntactic Distribution Consistency

In this section, we firstly present the *syntactic distribution consistency* assumption and show how to model it as constraint. Then, this constraint is integrated into a context-based probabilistic model(Multinomial Naive Bayes) under the framework of Posterior Regularization (PR).

We present our method only from the perspective of aspect word candidates. The process for opinion word candidates is similar. We use *ac* to represent an aspect word candidate, and *acc* to represent a golden aspect candidate cluster.

2.1.1 Syntactic Distribution Consistency

In this subsection, we describe *syntactic distribution consistency* formally, and show how to use it robustly on unbalanced data.

We propose a novel corpus-level syntactic representation for each aspect word candidate, by simply counting how many times the word is connected to each opinion word candidate through each direct dependency relation in the corpus. Taking aspect word candidate "screen" for example, its syntactic representation is

$$\begin{aligned} & Syn(ac = screen) \\ = & \big(Count(\overset{nsubj}{\longleftarrow} clear), Count(\overset{nsubj}{\longleftarrow} fine), \\ & ..., Count(\overset{rcmod}{\longrightarrow} expensive), ...\big)^T \\ = & \big(73, 31, ..., 0, ...\big)^T \end{aligned} \tag{1}$$

For an aspect candidate cluster, we simply add all syntactic vectors of the cluster's members together.

Given an ideal cluster acc_i, let us consider its syntactic representation. In a large review dataset, aspect words in cluster acc_i could be modified by different opinion words through different dependency relations many times. Each time can be viewed as a trial, and the syntactic vector of candidate cluster acc_i follows a certain multinominal distribution $\vec{p_i}$, which is the syntactic distribution of acc_i.

$$Syn(acc_i) \sim Multinomial(n_i, \vec{p_i}) \tag{2}$$

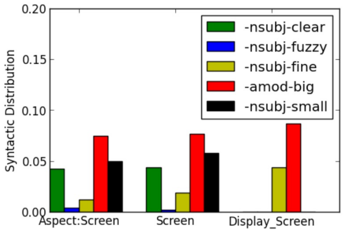
where n_i is calculated by adding all elements in vector $Syn(acc_i)$ together.

Based on the syntactic representation, we further assume that syntactic vector of candidate word ac_j follows the same distribution that its cluster follows. This leads to our *syntactic distribution consistency* assumption:

- ***Syntactic Distribution Consistency*** : The syntactic distribution of *aspect word candidate* is the same as that of the corresponding *aspect candidate cluster*. Formally, for any aspect word candidate $ac_j \in$ aspect candidate cluster acc_i, $Syn(ac_j) \sim Multinomial(n_j, \vec{p_i})$.

n_j is calculated by adding all elements in vector $Syn(ac_j)$.This assumption is verified on our dataset.

However, the frequencies of aspect word candidates could vary from very small to very large, making the estimate of these distributions unstable. As shown in Fig 3 [1], the estimated syntactic distribution of frequent word is reliable, and our assumption holds well. But for less frequent words, their estimated distributions are different from that of their clusters due to insufficient observations.



	Count
Aspect:Screen	3,267
Screen	1,670
Display-Screen	23

Figure 3: Estimated syntactic distribution of Aspect:Screen and its aspect word "Screen" and "Display-Screen".

[1] We calculate the syntactic distribution on the laptop domain. For simplicity, we only show probability on 5 dependency-opinion features. A frequent aspect candidate could be connected to tens of opinion candidates in total.

We employ *Pearson's chi-squared test* to judge whether this difference between the expected distribution and observed candidate syntactic distribution is due to sampling variation, or differs significantly. The chi-square statistic is a summary measure of how well the observed frequencies of categorical data match the frequencies that would be expected under certain multinomial distribution. In our case, we want to measure how well $Syn(ac_j)$ match $\vec{p_i}$, which is the syntactic distribution of acc_i. According to *Pearson's chi-squared test*, we have the following equation.

$$\chi^2(acc_i, ac_j) = \sum_{k=1}^{k=dim(\vec{p_i})} \frac{(n_j \vec{p_i}^{(k)} - Syn(ac_j)^{(k)})^2}{n_j \vec{p_i}^{(k)}} \le \chi^2_{dim(\vec{p_i})-1,\alpha} \tag{3}$$

$dim(\vec{p_i})$ represents the dimension of vector $\vec{p_i}$, and α corresponds to a certain confidence level. We look up the chi-square distribution table to find the right threshold corresponding to a certain confidence level(such as 95%, $\alpha = 0.05$) with the freedom of $dim(p_i) - 1$. If aspect candidate ac_j belongs to aspect candidate cluster acc_i, then we are confident that the above equation holds. In other words, if the above equation doesn't hold, we are confident that $ac_j \notin acc_i$ according to our assumption.

We introduce an indicator variable z_{ij} to represent whether aspect candidate ac_j belongs to aspect candidate cluster acc_i, as follows:

$$z_{ij} = \begin{cases} 1 & ;\text{if } ac_j \in acc_i \\ 0 & ;\text{otherwise} \end{cases} \tag{4}$$

This leads to our *syntactic distribution consistency(SDC)* constraint function.

$$z_{ij}\chi^2(acc_i, ac_j) \le \chi^2_{dim(\vec{p_i})-1,\alpha} \tag{5}$$

Thus, we can leverage syntactic information for both frequent words and less frequent words in a very robust way. Only when we have enough observations for ac_j and the reliable estimated syntactic distribution of ac_j is really different from that of acc_i, we apply $ac_j \notin acc_i$.

2.1.2 Syntactic Distribution Consistency Regularized Multinomial Naive Bayes (SDC-MNB)

In this section, we present our probabilistic model which employs both context information and syntactic distribution.

First of all, we extract a context document d to represent each candidate, by collecting the preceding and following t words of a candidate in each review. We use D to represent the document collection for all aspect word candidates. Assuming that the documents in D are independent and identically distributed, the probability of generating D is then given by:

$$p_\theta(D) = \prod_{j=1}^{|D|} p_\theta(d_j) = \prod_{j=1}^{|D|} \sum_{y_j} p_\theta(d_j, y_j) \tag{6}$$

where y_j is a latent variable indicating the cluster label for aspect word candidate ac_j, and θ is the model parameter.

In our problem, we are actually more interested in the posterior distribution over cluster label, i.e., $p_\theta(y_j|d_j)$. Once the learned parameter θ is obtained, we can get our clustering result from $p_\theta(y_j|d_j)$, by assigning cluster label acc_i with the largest posterior to aspect word candidate ac_j. We can also enforce SDC-constraint in expectation(on posterior p_θ). We use $q(Y)$ to denote the valid posterior distribution that satisfy our SDC-constraint, and Q to denote the valid posterior distribution space, as follows:

$$Q = \{q(Y) : E_q[z_{ij}\chi^2(acc_i, ac_j)] \le \chi^2_{dim(\vec{p_i})-1,\alpha}, \forall i, j\}. \tag{7}$$

Since posterior plays such an important role in joining the context model and SDC-constraint, we formulate our problem in the framework of Posterior Regularization (PR). PR is an efficient framework to inject constraints on the posteriors of latent variables. Instead of restricting p_θ directly, which might not be feasible, PR penalizes the distance of p_θ to the constraint set Q. The posterior-regularized objective is termed as follows:

$$\max_\theta\{\log p_\theta(D) - \min_{q\in Q} KL(q(Y)||p_\theta(Y|D))\} \tag{8}$$

By trading off the data likelihood of the observed context documents (as defined in the first term), and the KL divergence of the posteriors($p_\theta(Y|D)$) to the valid posterior subspace(Q) defined by SDC-constraint (as defined in the second term), the objective encourages model with both desired posterior distribution and data likelihood. In essence, the model attempts to maximize data likelihood of context subject (softly) to SDC-constraint.

Multinomial Naive Bayes.

In spirit to [19], we use Multinomial Naive Bayes (MNB) to model the context document. Let $w_{d_j,k}$ denotes the k^{th} word in document d_j, where each word is from the vocabulary $V = \{w_1, w_2, ..., w_{|V|}\}$. For each aspect phrase f_j, the probability of its latent aspect cluster label being acc_i and generating context document d_j is

$$p_\theta(d_j, y_j = acc_i) = p(acc_i) \prod_{k=1}^{|d_j|} p(w_{d_j,k}|acc_i) \tag{9}$$

where $p(acc_i)$ and $p(w_{d_j,k}|acc_i)$ are parameters of this model. Each word $w_{d_j,k}$ is conditionally independent of all other words given the cluster label acc_i.

The optimization algorithm for our model is an EM-like algorithm, which can be easily implemented as described in [3]. So we omit the algorithm here. After each E-step, the syntactic distribution of each cluster is updated by adding and normalizing all syntactic vectors of the cluster's candidates together according to the current posterior distribution q.

2.2 Extraction Based on Cluster-Cluster Association

Our extraction model consists of two steps: 1. We extract aspect cluster and opinion cluster based on cluster-cluster association. 2. Since clustering results may not be perfect and each cluster may still contain some noisy words, we propose a word ranking algorithm based on word-cluster association. At the end of this section, we also present an in-depth comparison between our model and models based on word-word association.

2.2.1 Association-based Cluster Extraction

As mentioned earlier, the clustering process leads to dramatic boosting on the association between associated pairs of aspect and opinion, and little or no boosting on the association of other cluster pairs. This leads to our following assumption,

Assumption: If a pair of clusters have strong association, the two clusters are likely to be a pair of aspect cluster and opinion cluster.

In spirit to [4], we adopt the Likelihood Ratio Test (LRT) statistics to measure cluster-cluster association. LRT computes a contingency table of aspect candidate cluster AC_i and opinion candidate cluster OC_j, derived from corpus statistics, as given in Table 1. $k_1(AC_i, OC_j)$ is the number of times that words in AC_i and words in OC_j is connected by an direct dependency relation. $k_2(AC_i, \overline{OC_j})$ is the number of dependency relation connected pair that contains words in AC_i but not words in OC_j. $k_3(\overline{AC_i}, OC_j)$ is the number of dependency relation connected pair that contains words in OC_j but not words in AC_i. $k_4(\overline{AC_i}, \overline{OC_j})$ is the number of dependency relation connected pair that contains neither words in AC_i nor words in OC_j. Note that our purpose here is to measure how strongly pair-wise clusters are associated with each other given the corpus statistics, rather than performing an actual statistics test.

Corpus statistics	AC_i	$\overline{AC_i}$
OC_j	$k_1(AC_i, OC_j)$	$k_2(AC_i, \overline{OC_j})$
$\overline{OC_j}$	$k_3(\overline{AC_i}, OC_j)$	$k_4(\overline{AC_i}, \overline{OC_j})$

Table 1: Contingency table for aspect candidate cluster AC_i and opinion candidate cluster OC_j

Based on the corpus statistics shown in Table 1, the LRT model captures the statistical association between aspect candidate cluster AC_i and opinion candidate cluster OC_j by employing the following function:

$$\begin{aligned} LRT(AC_i, OC_j) =& f(k_1, k_2, k_3, k_4) \\ =& 2[logL(p1, k1, n1) + logL(p2, k2, n2) \\ & - logL(p, k1, n1) - logL(p, k2, n2)] \end{aligned} \quad (10)$$

where,

$$\begin{aligned} & L(p, k, n) = p^k(1-p)^{n-k}; n_1 = k_1 + k_3; \\ & n_2 = k_2 + k_4; p_1 = k_1/n_1; p_2 = k_2/n_2; \\ & p = (k_1 + k_2)/(n_1 + n_2;) \end{aligned}$$

The larger the quantity $LRT(AC_i, OC_j)$ is, the stronger the statistical association between aspect candidate cluster AC_i and opinion candidate cluster OC_j. We rank all cluster pairs by the LRT score, and top ranked pairs are extracted as aspect clusters and opinion clusters.

2.2.2 Association-based Word Ranking

After cluster level extraction, we get pairs such as "Aspect:Screen" - "Opinion:Clear(or not)". However, there might be some noisy words in the "Aspect:Screen" cluster or "Opinion:Clear(or not)" cluster. For each word in the "Aspect:Screen" cluster, the stronger the association of the word to the "Opinion:Clear(or not)" cluster, the more likely the word is an aspect word in "Aspect:Screen".

This motivates our word ranking method. For an extracted aspect/opinion cluster, we select its corresponding opinion/aspect cluster with the strongest association, and calculate the association between each word with the corresponding cluster. The higher the association score is, the word is more likely to be the aspect/opinion word of this cluster. The word-cluster association is also calculated by LRT, based on the corpus statistics shown in Table 2 [2]

Corpus statistics	C_i	$\overline{C_i}$
T_j	$k_1(C_i, T_j)$	$k_2(C_i, \overline{T_j})$
$\overline{T_j}$	$k_3(\overline{C_i}, T_j)$	$k_4(\overline{C_i}, \overline{T_j})$

Table 2: Contingency table for term T_j and cluster C_i

First of all, for each selected aspect/opinion cluster, we selected the opinion/aspect cluster with the largest LRT score as the corresponding paired cluster. Then, for each word T_j in the aspect/opinion cluster, we calculate the LRT score $LRT(T_j, C_i)$ between the word and its cluster's corresponding paired cluster C_i. Finally, each word is ranked by the LRT score. The higher the LRT score is, the word is more likely to be an aspect/opinion word of the cluster.

2.2.3 Key Difference to Word-word Association based Models

As mentioned above, existing models based on word level association may discover some "False Opinion Relations"(false word-word associations) as discussed in [16]. We will show that our cluster level associations are more robust and less prone to noises than word level associations.

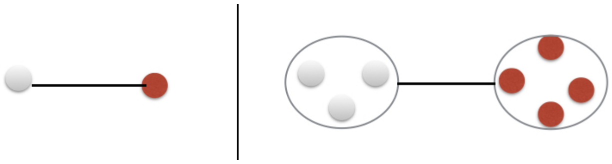


Figure 4: Left is a false association where a non-aspect word *na* connects an opinion word *o*. Right is the cluster-cluster association between the cluster of *na* and that of *o*.

As shown in Fig. 4, there is a false association that connects a non-aspect word *na* and an opinion word *o*. This non-aspect word *na* may be extracted as "aspect word" by word-word association based models. In our model, we examine the association between the cluster of the non-aspect word *na* and the cluster of the opinion word *o*. The association between this two clusters will be much weaker than that between a real pair of aspect and opinion cluster. Thus this pair of clusters will not be extracted by our cluster-cluster association based method. In practice, our method is also better at filtering non-aspect/non-opinion words.

Another important difference is that we not only extract but also cluster aspect/opinion words, while existing models that leverage word-word association only extract aspect/opinion words. For instance, for the aspect "picture", the model extracts opinion clusters "clear(or not)", "bright(or not)" and "good(or not)" as shown in Table 5 in experiment.

3. EXPERIMENTS

In this section, we evaluate our approach on reviews from various domains. To assess the effectiveness of our approach, we conduct three experiments on the task of aspect extraction, clustering, aspect and opinion word extraction.

[2] The corpus statistics is calculated in a similar way as in Table 1.

- ***Aspect Extraction***: Aspect extraction refers to the task of extracting and clustering aspect words simultaneously. We compare our model with existing knowledge-based topic models.
- ***Clustering***: Aspect/opinion word candidate clustering is a key component in our model. We compare our unsupervised, syntactic distribution consistency inspired clustering model with other unsupervised clustering models.
- ***Aspect and Opinion Word Extraction***: Aspect and opinion word extraction aims at extracting opinion targets (e.g., *battery*) and opinion words (e.g., *amazing*). In this task, we don't care about which cluster each word belongs to. Two state-of-art methods are selected as baselines.

Since there is no prior work on clustering opinion words, the result of opinion word clustering is qualitatively evaluated in this paper.

3.1 Datasets and Evaluation Metrics

We evaluate our method on a large Chinese review corpus, as used in [21]. This dataset contains reviews from four domains: *Camera*, *Cellphone*, *Laptop*, and *MP3/MP4*. The statistics of the corpus are present in Table 3.

We create a gold standard for aspect/opinion word extraction and clustering.[3] Since the dataset is very large, it is impossible to go through every review and label each word manually. Therefore, we select all nouns with frequency larger than 3 (5 for the cellphone domain) as aspect candidates, and all adjectives as opinion candidates. We go through every candidate word, and read sampled reviews which contain this candidate to judge whether it's an aspect/opinion word or not. Since it is impossible to discover all the aspect/opinion words, it is infeasible to evaluate the recall measure.

	Camera	Cellphone	Laptop	MP3/MP4
#Products	449	694	702	329
#Reviews	101,235	579,402	102,439	129,471
#Aspect Candidates	3,686	6,360	4,373	3,724
#Aspect Words	894	1,321	993	930
#Aspect	27	26	25	27
#Opinion Candidates	1,614	2,235	1,657	1,617
#Opinion Words	806	880	791	792
#Opinion	28	28	28	28

Table 3: Statistics of the review corpus. # denotes the size.

Evaluation Metrics: Similar to previous work [2] [13] we adopt precision@k (or p@k), where k is the rank position to compare different approaches. As just mentioned, recall is not applicable in our settings.

3.2 Evaluation on Aspect Extraction

Aspect extraction refers to the task of extracting and clustering aspect words simultaneously. Existing models for aspect extraction are all topic-model based methods[1][2][13].

In this section, we compare our model with three topic-model based aspect extraction methods: Latent Dirichlet allocation(**LDA**), Seeded Aspect and Sentiment model(**SAS**)[13] and topic modeling with Automatically generated Must-links and Cannot-links(**AMC**)[1]. Those baselines represent three types of topic models: unsupervised, semi-supervised, self-learning.

- ***LDA***: We implement LDA as suggested in [2]. Each sentence is treated as a document.
- ***SAS***: SAS is implemented as suggested in [13]. For SAS, we randomly sample 15 topics as labeled topics, and provide 5 seeds for each labeled topic.
- ***AMC***: AMC is a self-learning topic model, which learns knowledge from other similar domains. This method requires many similar domains to work well. Since our dataset only contains 4 domains, we randomly split data in each domain into 5 pieces, and each piece is treated as a "domain" in AMC. Each sentence is treated as a document. We use the code provided by [4][1].

Settings:

The parameter of all topic models are set to $\alpha = 1$, $\beta = 0.1$. For LDA and AMC, each sentence is treated as a document.

Setting the number of topics/aspects is often tricky. We follow the approach in [13] and assume that we already know the gold aspect number.

Note that the topic models used in our experiments are also with only candidate words but not with all words without filtering. The filtering process always promotes the performance of these topic models. We believe this configuration makes it more fair to compare our method with topic model baselines.

For our method, we set the cluster number to be 100, and the extracted aspect number to be the gold aspect number. For clustering, we set the window size to be 4 to extract contexts, and set α to be 0.05 to get the chi-squared threshold. Since our method depends on the random initiation, we use the average result of 5 runs as the final result.

3.2.1 *Quantitative Results*

We follow the evaluation methods in [2]. For each topic (or cluster in our setting), we judge it as a "good topic" if the top 15 words contain at least 5 synonymous aspect words. If so, we also assign the corresponding aspect to this topic.

For each good topic, we evaluate whether the top-ranked words belong to the corresponding aspect of the topic in terms of precision@k (or p@k).

	Camera	Cellphone	Laptop	MP3/MP4
LDA	3.8/27	4.0/26	6.8/25	4.8/27
AMC	8.6/27	11.4/26	13.8/25	11.8/27
SAS	6.0/27	6.0/26	15.0/25	9.0/27
Ours	**16.8**/27	**15.6**/26	**15.4**/25	**18.0**/27

Table 4: Comparison with topic model baselines : each cell denotes the number of good topics/ all extracted topics.

Experiment results are shown in Table 4 and Fig.5. We can see that our approach can extract more good topics with better precision than all baselines on all domains. The merits of our approach may come from the following facts:

- We have an explicit extraction process and filter many non-aspect words. While in knowledge based topic models, they attempt to rank aspect words higher and generate less noisy topics. But since all non-aspect

[3] We will release our gold standard later.

[4] https://github.com/czyuan/AMC.git

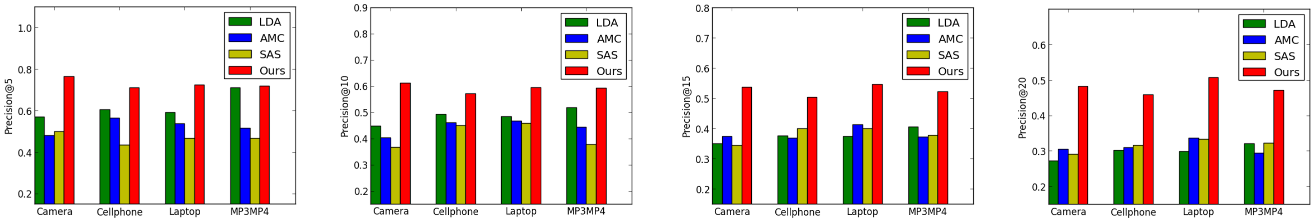


Figure 5: Comparison with topic model baselines : p@k of good topics across different domains. k = 5,10,15,20.

Our Method							
Aspect	Opinion			Aspect	Opinion		
Picture	Clear	Bright	Good	Battery	Good	Long-lasting	
画面 (picture)	清晰 (clear)	艳丽 (bright)	好 (good)	时间 (time)	好 (good)	耐用 (durable)	
图像 (picture)	细腻 (fine)	亮丽 (bright)	差 (bad)	寿命 (life)	差 (bad)	持久 (long-lasting)	
画质 (picture)	清楚 (clear)	逼真 (lifelike)	棒 (good)	周期 (period)	棒 (good)	强悍 (powerful)	
图象 (image)	柔和 (soft)	饱满 (full)	烂 (suck)	待机 (standby)	烂 (suck)	强劲 (powerful)	
影像 (picture)	顺畅 (smooth)	淡 (light)	完美 (perfect)	续航 (life)	完美 (perfect)	受用 (powerful)	
字迹 (writing)	满 (full)	锐利 (sharp)	一般 (ordinary)	电力 (electricity)	一般 (ordinary)	长久 (long)	
色彩 (color)	过瘾 (fun)	鲜明 (bright)	差劲 (poor)	使用 (usage)	差劲 (poor)	猛 (powerful)	
图画 (picture)	快活 (happy)	鲜活 (bright)	理想 (ideal)	用量 (amount)	理想 (ideal)	凶猛 (powerful)	
画片 (picture)	壮观 (great)	鲜艳 (bright)	糟糕 (bad)	研究 (study)	糟糕 (bad)	充沛 (abundant)	
图案 (picture)	生动 (vivid)	黯淡 (dim)	可靠 (reliable)	供电 (power)	可靠 (reliable)	弱 (weak)	
SAS				**AMC**		**LDA**	
Picture		Battery		Battery		Battery	
Aspect	Opinion	Aspect	Opinion	Aspect	Opinion	Aspect	Opinion
画面 (picture)	大 (big)	时间 (time)	长 (long)	待机 (standby)	短 (short)	时间 (time)	长 (long)
显示 (display)	清晰 (clear)	小时 (hour)	耐用 (durable)	时间 (time)	长 (long)	电池 (battery)	耐用 (durable)
效果 (effect)	好 (good)	待机 (standby)	好 (good)	小时 (hour)	耐用 (durable)	小时 (hour)	短 (short)
视频 (video)	不错 (good)	使用 (usage)	不错 (good)	电池 (battery)	坏 (bad)	待机 (standby)	强 (strong)
色彩 (color)	高 (high)	能力 (ability)	短 (short)	续航 (life)	粘 (sticky)	续航 (life)	低 (low)
感觉 (feel)	小 (small)	充电 (charge)	强 (strong)	能力 (ability)	最大 (biggest)	能力 (ability)	清楚 (clear)
清晰度 (sharpness)	低 (low)	电量 (power)	行 (ok)	充电 (charge)	好看 (good-looking)	电量 (power)	太多 (too much)
图像 (image)	清楚 (clear)	播放 (broadcast)	久 (long)	使用 (usage)	次 (bad)	充电 (charge)	实际 (actual)
角度 (angle)	差 (suck)	电影 (movie)	次 (bad)	CPU (CPU)	可惜 (pity)	颜色 (color)	不错 (good)
电影 (movie)	细腻 (fine)	容量 (capacity)	差 (bad)	温度 (temparater)	耐磨 (wear-resisting)	选择 (choice)	麻烦 (troublesome)

Table 5: Qualitative results on aspect extraction and opinion extraction in MP3/MP4 domain.

Aspect	Camera			Cellphone			Laptop			MP3/MP4		
	DP	LRT	Ours	DP	LRT	Ours	DP	LRT	Ours	DP	LRT	Ours
P@100	0.830	0.810	**0.872**	0.780	0.710	**0.846**	0.820	0.810	**0.858**	0.820	0.810	**0.899**
P@200	0.695	0.695	**0.764**	0.655	0.645	**0.677**	0.710	0.685	**0.767**	0.715	0.685	**0.777**
P@300	0.620	0.607	**0.649**	0.592	0.583	**0.593**	0.646	0.633	**0.651**	0.626	0.603	**0.691**
P@400	0.580	0.560	**0.595**	**0.550**	0.543	0.529	**0.595**	0.585	0.584	0.588	0.575	**0.615**
P@500	0.532	0.516	**0.550**	0.496	0.492	**0.499**	**0.552**	0.542	0.543	0.556	0.550	**0.588**
Opinion	Camera			Cellphone			Laptop			MP3/MP4		
	DP	LRT	Ours	DP	LRT	Ours	DP	LRT	Ours	DP	LRT	Ours
P@100	0.900	0.890	**0.968**	0.890	0.880	**0.906**	0.890	0.880	**0.926**	0.870	0.880	**0.929**
P@200	0.835	0.780	**0.909**	0.815	0.810	**0.905**	0.805	0.785	**0.869**	0.830	0.805	**0.881**
P@300	0.796	0.770	**0.855**	0.783	0.760	**0.873**	0.760	0.740	**0.817**	0.786	0.766	**0.841**
P@400	0.770	0.757	**0.788**	0.767	0.735	**0.833**	0.747	0.740	**0.774**	0.770	0.737	**0.788**
P@500	0.748	0.726	**0.758**	0.756	0.716	**0.792**	**0.740**	0.712	0.728	0.746	0.730	**0.749**

Table 6: Comparison on aspect and opinion word extraction across different domains.

words are still in their ranking list, it's hard to low-rank frequent non-aspect words.

- We leverage both context information and syntactic information to generate aspect word clusters. While in knowledge based topic models, they only use context information. As discussed above, our method can distinguish words with similar contexts but different syntactic distributions, leading to better clustering performance.
- In knowledge based topic models, there are topics which contain different aspects. Knowledge-based topic models are good at extracting coarse-grained aspects, while our model is good at extracting fine-grained aspects.
- SAS and AMC are better than LDA, which demonstrates the effectiveness of human-provided knowledge and self-learning.

3.2.2 *Qualitative Results*

Table 5 shows two example aspects ("picture" and "battery") and its top 10 words produced by each methods in MP3/MP4 domain. From Table 5, we can see that our method discovers more correct and meaningful aspect words at the top positions. While AMC and LDA fail to discover fine-grained aspect "picture".

Compared with topic-model based methods, our method can extract aspect cluster and opinion cluster simultaneously. As a result, our methods can capture the many-to-many association between aspect clusters and opinion clusters. For example, opinion "good(or not)" is related to both aspect "picture" and aspect "battery" in our extraction result. Our method can also extract and distinguish fine-grained opinion cluster. For aspect "picture", we extract three related fine-grained opinion clusters: "clear(or not)", "bright(or not)" and "good(or not)". While topic model based methods can only extract one mixed opinion topic for each aspect topic.

We believe that extracting fine-grained opinion clusters has potential value for generating fine-grained opinion summary. For example, we may want to know more detailed opinions about "picture", i.e., whether it's "clear(or not)", "bright(or not)", or generic opinions such as "good(or not)".

We can also see that some opinion cluster contains both synonymous and antonymous opinion words. Separating positive opinion words and negative opinion words is beyond the scope of this paper.

3.3 Evaluation on Clustering

In this section, we want to evaluate the effectiveness of leveraging syntactic distribution consistency in the clustering component. We conduct an extrinsic experiment that evaluate the final aspect extraction performance, by changing the clustering component. We compare our clustering model with two simple clustering methods: **Kmeans** and **MNB-EM**.

- ***Kmeans***: Kmeans is the most popular clustering algorithm, and is also used as baseline in previous work[21]. Here we use the context distributional similarity (cosine similarity) as the similarity measure.
- ***MNB-EM***: MNB-EM is our clustering model without the *syntactic distribution consistency* constraint. Compared with MNB-EM, we can see clearly the effectiveness of leveraging syntactic distribution consistency.

The result is shown in Fig.6. We can see that our SDC-MNB clustering model can extract more good topics with better or comparable precision than baselines, which demonstrates the merits of syntactic distribution consistency in the clustering component. Besides, our extraction model with kmeans is comparable to topic model baselines, which demonstrates the effectiveness of our "first clustering, then extracting" framework.

	Camera	Cellphone	Laptop	MP3/MP4
Kmeans	8.4/27	10.4/26	9.4/25	10.6/27
MNB-EM	14.0/27	14.2/26	11.2/25	17.8/27
SDC-MNB	**16.8**/27	**15.6**/26	**15.4**/25	**18.0**/27

Table 7: Comparison with clustering baselines : each cell denotes the number of good topics/ all extracted topics.

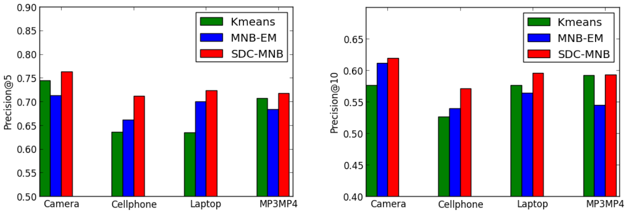


Figure 6: Comparison with clustering baselines : p@k of good topics across different domains. k = 5,10.

3.4 Evaluation on Aspect and Opinion Word Extraction

In this section, we evaluate our method on the aspect and opinion word extraction task. In this task, we don't care about which cluster each word belongs to. An extracted word is correct, as long as it's an aspect/opinion word. We choose precision@k as evaluation measure. We compare our method with two strong baselines: **DP** and **LRT**.

- ***DP***: DP uses syntax-based patterns to capture word-word associations in sentences, and then uses a bootstrapping process to extract aspect/opinion words [14].
- ***LRT***: LRT captures associations using co-occurrence statistics. They employ these statistics to extract opinion words/targets in a bootstrapping framework[4].

Extracted aspect/opinion words are ranked by frequency in DP and LRT.

The result is shown in Table 6. We can see that our method outperforms two baselines for most cases. This result demonstrates that, based on candidate clustering result, our simple ranking method works well for aspect and opinion word extraction task.

We also find that our method works especially well for smaller k in term of p@k. As k increases, the performance of our method drops only slightly. The reason is as follows: In the first step of extraction, we extract aspect/opinion cluster. If the extracted cluster number is small, then the extracted clusters may not cover all aspect/opinion words. So our method doesn't perform well for aspect/opinion word extraction in terms of p@k for larger k. However, since we don't care about the aspect number in this task, a larger cluster number could solve this problem.

4. RELATED WORK

Sentiment extraction is an important task for aspect-level sentiment analysis. Previous works can be divided into two categories: sentence level extraction and corpus level extraction.

For sentence-level extraction, previous methods mainly aimed to identify all opinion target/word mentions in sentences by supervised learning[15] [12][7] [17]. They regarded it as a sequence labeling task, where several classical models were used. Jin and Ho applied a lexicalized HMM model to learn patterns to extract aspects and opinion expressions[6]. Li et al. integrated two CRF variations, i.e., Skip-CRF and Tree-CRF, to extract aspects and also opinions[7].

This paper falls into corpus level extraction, and aims to generate an aspect/opinion cluster list rather than to identify mentions in sentence. Previous corpus level extraction methods can be divided into two categories: knowledge-based topic models that leverage word-word co-occurrence information implicitly, and graph-based models that model the association between aspect and opinion candidates explicitly.

Knowledge-based topic models performed aspect term extraction and clustering simultaneously by leveraging word-word co-occurrence information implicitly. To generate coherent topics, several knowledge-based topic models, have been proposed. Mukherjee and Liu proposed Seeded Aspect and Sentiment model(SAS) by using human-generated seeds to learn coherent topics[13]. Chen et al. proposed MC-LDA to leverage both must-sets and cannot-sets[2]. Chen et al. also proposed AMC to learning knowledge from other domain[1].

Our method is similar to topic models in terms of discovering aspect and opinion based on word co-occurrence. The key difference is that topic models can't distinguish aspect words and background(or non-aspect) words based on word association in an unsupervised manner, but our model can. Topic models such as ME-LDA[22] includes a word classification module, which classifies words into three categories: aspect, opinion, and background. But they need heavy annotation. Topic models such as SAS work in a weakly supervised manner. But they treat all words as either aspect or opinion words. Since not all words are aspect words or opinion words, the learned topic must contains some background words. If a background word is frequent, it's usually hard to filter it. While our method can filter frequent non-aspect words based on cluster-cluster association.

Graph-based models extracted aspect and opinion terms by modeling association between aspect and opinion candidates explicitly. Those models usually started from seeds and propagate label on a graph. The word-word association has been studied by many researchers. Hu and Liu exploited nearest neighbor rules to mine association among words[5]. Qiu et al. designed syntactic patterns to perform this task[14]. Liu et al. employed word alignment model to capture word-word association rather than syntactic parsing[9][8][10]. Hai et al. explore robust statistical correlation in a bootstrapping framework[4].

Our method is similar to graph-based models in terms of employing association strength for extraction. However, our method extract aspect and opinion by leveraging cluster-cluster association, while graph-based models employ word-word association. We found that the bond between aspect and opinion clusters are stronger and less prone to noises than word level associations.

Another line of work focused on clustering aspect terms based on context information. Several weakly supervised models has been proposed. Zhai et al. proposed an EM-based semi-supervised learning method to group aspect expressions into user-specified aspects[18]. They employed lexical knowledge to provide a better initialization for EM. In Zhai et al.[19], an EM-based unsupervised version was proposed. The so-called L-EM model first generated softly labeled data by grouping feature expressions that share words in common, and then merged the groups by lexical similarity. Zhai et al.[20] proposed a LDA-based method that incorporates must-link and cannot-link constraints. Zhao et al. proposed an EM-based unsupervised method by leveraging data-driven sentiment consistency constraint[21].

Compared with previous work, our model has two advantages: 1. Our model can extract aspect and opinion based on cluster-cluster association. 2. Our model is totally unsupervised, doesn't need any labeled data, human-generated knowledge, or seeds.

5. CONCLUSION

In this paper, we propose a novel sentiment extraction method by leveraging *aspect-opinion association structure*. First, we cluster all aspect candidates and opinion candidates by our *syntactic distribution consistency* inspired model. Second, we extract aspect and opinion word clusters based on cluster-cluster association. Compared with existing topic-model based aspect extraction methods, our method performs an explicit extraction process and filter many non-aspect/non-opinion candidates. Compared with existing graph-based methods that extract aspect words and opinion words, our method can extract aspect and opinion at cluster level. Experiments show that our approach outperforms state-of-the-art baselines remarkably.

6. ACKNOWLEDGEMENT

This work was partly supported by the National Basic Research Program (973 Program) under grant No.2012CB316301 /2013CB329403, the National Science Foundation of China under grant No.61272227/61332007, and the Beijing Higher Education Young Elite Teacher Project. The work was also supported by Tsinghua University Beijing Samsung Telecom R&D Center Joint Laboratory for Intelligent Media Computing.

7. REFERENCES

[1] Z. Chen and B. Liu. Mining topics in documents: Standing on the shoulders of big data. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1116–1125, New York, NY, USA, 2014. ACM.

[2] Z. Chen, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh. Exploiting domain knowledge in aspect extraction. In *EMNLP*, pages 1655–1667. ACL, 2013.

[3] J. V. Graca, L. Inesc-id, K. Ganchev, B. Taskar, J. V. Graça, L. F. Inesc-id, K. Ganchev, and B. Taskar. Expectation maximization and posterior constraints. In *In Advances in NIPS*, pages 569–576, 2007.

[4] Z. Hai, K. Chang, and G. Cong. One seed to find them all: Mining opinion features via association. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 255–264, New York, NY, USA, 2012. ACM.

[5] M. Hu and B. Liu. Mining opinion features in customer reviews. In *Proceedings of the 19th National Conference on Artifical Intelligence*, AAAI'04, pages 755–760. AAAI Press, 2004.

[6] W. Jin and H. H. Ho. A novel lexicalized hmm-based learning framework for web opinion miningnote from acm: A joint acm conference committee has determined that the authors of this article violated acm's publication policy on simultaneous submissions. therefore acm has shut off access to this paper. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 465–472, New York, NY, USA, 2009. ACM.

[7] F. Li, C. Han, M. Huang, X. Zhu, Y.-J. Xia, S. Zhang, and H. Yu. Structure-aware review mining and summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 653–661, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[8] K. Liu, L. Xu, Y. Liu, and J. Zhao. Opinion target extraction using partially-supervised word alignment model. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pages 2134–2140. AAAI Press, 2013.

[9] K. Liu, L. Xu, and J. Zhao. Opinion target extraction using word-based translation model. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 1346–1356, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[10] K. Liu, L. Xu, and J. Zhao. Syntactic patterns versus word alignment: Extracting opinion targets from online reviews. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1754–1763, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[11] K. Liu, L. Xu, and J. Zhao. Extracting opinion targets and opinion words from online reviews with graph co-ranking. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 314–324, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[12] T. Ma and X. Wan. Opinion target extraction in chinese news comments. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 782–790, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[13] A. Mukherjee and B. Liu. Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 339–348, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[14] G. Qiu, B. Liu, J. Bu, and C. Chen. Opinion word expansion and target extraction through double propagation. *Comput. Linguist.*, 37(1):9–27, Mar. 2011.

[15] Y. Wu, Q. Zhang, X. Huang, and L. Wu. Phrase dependency parsing for opinion mining. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3*, EMNLP '09, pages 1533–1541, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[16] L. Xu, K. Liu, S. Lai, Y. Chen, and J. Zhao. Mining opinion words and opinion targets in a two-stage framework. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 1764–1773, 2013.

[17] B. Yang and C. Cardie. Joint inference for fine-grained opinion extraction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1640–1649, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[18] Z. Zhai, B. Liu, H. Xu, and P. Jia. Grouping product features using semi-supervised learning with soft-constraints. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 1272–1280, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[19] Z. Zhai, B. Liu, H. Xu, and P. Jia. Clustering product features for opinion mining. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 347–354, New York, NY, USA, 2011. ACM.

[20] Z. Zhai, B. Liu, H. Xu, and P. Jia. Constrained lda for grouping product features in opinion mining. In *Proceedings of the 15th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining - Volume Part I*, PAKDD'11, pages 448–459, Berlin, Heidelberg, 2011. Springer-Verlag.

[21] L. Zhao, M. Huang, H. Chen, J. Cheng, and X. Zhu. Clustering aspect-related phrases by leveraging sentiment distribution consistency. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1614–1623, Doha, Qatar, October 2014. Association for Computational Linguistics.

[22] W. X. Zhao, J. Jiang, H. Yan, and X. Li. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 56–65, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.