Latent Aspect Rating Analysis without Aspect Keyword Supervision

Hongning Wang, Yue Lu, ChengXiang Zhai
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana IL, 61801 USA
{wang296, yuelu2, czhai}@cs.uiuc.edu

ABSTRACT

Mining detailed opinions buried in the vast amount of review text data is an important, yet quite challenging task with widespread applications in multiple domains. *Latent Aspect Rating Analysis* (LARA) refers to the task of inferring both opinion ratings on topical aspects (e.g., location, service of a hotel) and the relative weights reviewers have placed on each aspect based on review content and the associated overall ratings. A major limitation of previous work on LARA is the assumption of pre-specified aspects by keywords. However, the aspect information is not always available, and it may be difficult to pre-define appropriate aspects without a good knowledge about what aspects are actually commented on in the reviews.

In this paper, we propose a unified generative model for LARA, which does not need pre-specified aspect keywords and simultaneously mines 1) latent topical aspects, 2) ratings on each identified aspect, and 3) weights placed on different aspects by a reviewer. Experiment results on two different review data sets demonstrate that the proposed model can effectively perform the *Latent Aspect Rating Analysis* task without the supervision of aspect keywords. Because of its generality, the proposed model can be applied to explore all kinds of opinionated text data containing overall sentiment judgments and support a wide range of interesting application tasks, such as aspect-based opinion summarization, personalized entity ranking and recommendation, and reviewer behavior analysis.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Text Mining

General Terms

Algorithms, Experimentation

Keywords

Aspect Identification, Latent Rating Analysis, Review Mining

KDD'11, August 21–24, 2011, San Diego, California, USA.

1. INTRODUCTION

In the era of Web 2.0, more and more people express their opinions on all kinds of entities, including products and services, which in turn help not only customers make informed decisions but also merchants improve their services. The rapid growth of such opinionated text data on the web, such as user reviews, raises interesting new challenges for text mining communities and leads to many studies on extracting information from reviews [4, 20, 17], sentiment analysis [19, 18, 11] and opinion summarization [6, 8, 14].

To help users efficiently and accurately digest a large number of online reviews about a particular entity (e.g., mp3 player), it is necessary to reveal detailed opinions on multiple topical aspects of the entity (e.g., battery life of mp3 player). To this end, recent work on opinion mining has attempted to perform fine-grained sentiment analysis: in most work (e.g., [21, 25, 6, 7]), the proposed algorithms are able to identify sentiment orientation or ratings on specific topical aspects, leading to useful detailed opinion summaries.

However, decomposing an overall rating into ratings on specific aspects is still not informative enough from a user's perspective. For example, a hotel with a five-star rating on the "*value*" aspect may actually be quite expensive by common standard if the reviewer emphasizes much more on the "*service*" of the hotel than the "*value*" (e.g., for a business trip), though it could also be indeed cheap if the reviewer really cares much more about the price than about other aspects of a hotel (e.g., for a casual vacation).

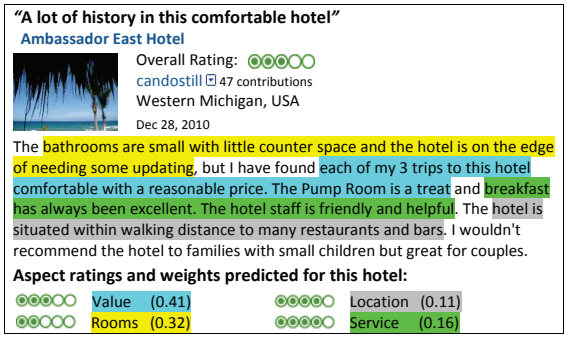


Figure 1: A sample output of LARA.

To further differentiate such different meanings of the same aspect rating, in our previous work [23], we went beyond aspect rating prediction to also infer the relative emphasis placed by a reviewer on different aspects, and introduced a

new opinion mining problem named *Latent Aspect Rating Analysis* (LARA). The task of LARA is to take as input a set of review text documents about an entity with overall ratings and generate as output 1) ratings on a set of predefined aspects of the entity, and 2) relative weights placed by a reviewer on each aspect when writing the review.

For example, Figure 1 shows a sample result of LARA on a hotel review where we see that LARA not only decomposes the overall rating into ratings on each of the four topical aspects (e.g., three stars on "*value*" and two stars on "*room*"), but also infers that the reviewer has placed a much higher weight on "*value*" than other aspects (thus the actual price of this hotel is likely indeed cheap). The inferred aspect weights of reviewers can be very useful. For example, to help a current user who also cares much about "*room*", we could selectively emphasize more on the reviews written by reviewers who have a similar taste to the current user (i.e., also placing a very high weight on "*room*") to recommend hotels, achieving personalized ranking of hotels. The inferred weights can also be used to analyze the rating behavior of reviewers with applications in business intelligence as shown in [23].

However, a major limitation of [23] is that the aspects must be given through keywords specified by users, restricting its usefulness in applications. Indeed, such supervision of specifying aspects with keywords requires manual work and is not always available. More importantly, in many cases, it is often unclear what are the aspects actually commented on in the reviews, thus it is very difficult, if not impossible, to pre-specify the aspects beforehand.

To address this limitation and enable LARA on arbitrary review data without explicit aspect keyword supervision, in this paper, we propose a unified generative Latent Aspect Rating Analysis Model (LARAM), which simultaneously identifies: 1) latent topical aspects, 2) ratings on each identified aspect, and 3) weights placed on different aspects by a reviewer. LARAM is a fully generative model for both the review text and the companion overall rating. Specifically, it is assumed that the text content describing a particular aspect is generated by sampling words from a topic model (i.e., a multinomial word distribution) corresponding to the latent aspect, the latent rating on an aspect is determined based on the words describing each aspect with latent sentiment polarities, and the overall rating is generated based on a weighted combination of aspect ratings where the (latent) weights reflect the relative emphasis on each aspect by the reviewer.

LARAM can be regarded as an extension of the Latent Rating Regression (LRR) model proposed in [23] to perform both aspect segmentation and aspect rating prediction in a unified framework (in contrast, the two tasks were performed separately in [23] with segmentation done based on user-provided keywords).

Experiment results on a hotel review data set crawled from TripAdvisor (`www.tripadvisor.com`) and a product review data set crawled from Amazon (`www.amazon.com`) show that the proposed LARAM can effectively perform Latent Aspect Rating Analysis task without requiring the supervision of aspect keywords. Since LARAM is a general framework, it can be applied to analyze any kind of review text data with overall sentiment judgment to discover opinionated latent topical aspects, decompose overall opinion into sentiment specific topical aspects, and infer relative weights placed by reviewers on different topical aspects; such detailed opinion analysis can support many interesting applications such as aspect-based opinion summarization, personalized opinion-based ranking of entities, and reviewer behavior analysis.

2. RELATED WORK

The closest work to this study is our previous work [23], which introduced the problem of Latent Aspect Rating Analysis (LARA) and proposed a two-stage approach: in the first stage, keywords specified by users are used in a bootstrapping algorithm to identify the aspects and segment the review content; in the second stage, a generative Latent Rating Regression (LRR) model is applied to infer aspect ratings and weights in a review. However, this previous work requires a user to specify aspect keywords in advance, which requires manual work and is often very hard to achieve without knowing well about the opinions buried in the target review text. In this paper, we overcome this limitation, and propose a new generative model (i.e., LARAM) that can be regarded as an extension of LRR to perform aspect segmentation and aspect rating prediction simultaneously in a unified framework, thus enabling LARA without needing keyword specification from users.

The proposed LARAM is a hybrid generative model containing both aspect modeling and rating prediction, thus it is also related to the work of using topic modeling techniques to extract aspects and associated opinions. Mei et al. incorporated two additional sentiment language models into topic models to extract the facets and positive/negative opinions in weblogs [16]. Later, some work further introduced aspect-specific sentiment models in different ways, e.g., using supervision from sentiment priors [12, 9] or supervision from labeled sentences [24]. In [21], Titov and McDonald extended their multi-grain topic model [22] to discover topics that are representative of ratable aspects. Their regression module requires "ground truth" user ratings on the pre-defined aspects, which are not always available. In contrast, the proposed LARAM does not require aspect ratings from users and can decompose overall ratings into the ratings on the discovered aspects. More importantly, none of the work in this line is able to identify a reviewer's relative emphasis on different aspects, which is required for accurate interpretation of aspect ratings as we discussed in Section 1.

Other work on finer granularity analysis of opinions expressed in review text content is mostly to analyze the aspect-level opinion orientations [25, 6, 7, 20, 4, 15]. However, these methods can only identify the opinions associated with each individual aspect, but they cannot infer the reviewer's relative emphasis on different aspects, which the proposed LARAM is designed to achieve.

3. LATENT ASPECT RATING ANALYSIS

As a text mining problem, Latent Aspect Rating Analysis (LARA) [23] is to take as input a set of reviews of some interesting entities with companion overall ratings, and discover: 1) latent topical aspects that are commented on in the reviews; 2) ratings on each individual latent aspects; and 3) relative weights placed on different aspects by a reviewer when generating the overall rating.

Formally, the input can be represented as $D = \{d_1, d_2, \cdots, d_{|D|}\}$, a set of review text documents for a particular entity, where each review $d \in D$ is associated with a numerical overall

rating r. A review d is modeled as a bag of words $\mathbf{W}$ in a fixed vocabulary $V = \{v_1, v_2, \ldots, v_{|V|}\}$.

The desired output of LARA consists of the following three kinds of detailed information about opinions buried in the reviews: 1) k topical aspects that are frequently commented on in all the review data: $\{A_1, A_2, \ldots, A_k\}$, where each aspect A_i is characterized by a topic model, i.e., multinomial word distribution $p(w|A_i)$, where $w \in V$; 2) for each review d, a k dimensional aspect rating vector $\mathbf{s}$, where $\mathbf{s}_i$ is the predicted rating for aspect A_i in the review; and 3) for each review d, an aspect weight vector α, where α_i is the relative weight (emphasis) placed by the reviewer on aspect A_i when writing the review. In the following discussions, we will use "aspect" and "topic" interchangeably unless noted otherwise.

Note that a main difference between our definition of LARA here and that in our previous work [23] is that in the previous definition, all the aspects are explicitly specified with keywords provided by a user, thus while the aspect ratings are latent, the aspects are not really latent, whereas here, both aspect ratings and the aspects themselves are latent, which we need to infer from the data. Naturally, such a definition of the problem is more challenging, but it is also more general. If we can solve such a problem setup, it would enable a much wider scope of future applications.

Because the aspects are assumed to be latent, the previously proposed Latent Rating Regression (LRR) model cannot solve the new problem setup directly. Below, we present an extension of the LRR model that can simultaneously discover latent aspects, decompose overall ratings into aspect ratings, and infer weights on different aspects.

4. A GENERATIVE MODEL FOR LARA

A main challenge in solving LARA without the aspect keywords supervision is to properly associate the words with those meaningful aspects corresponding to the major opinions. To address this challenge, our basic idea is to use topic modeling techniques which provide a convenient way of segmenting the review contents by exploiting the word co-occurrence patterns in the data. The main technical contribution of this work is to incorporate topic modeling technique into the Latent Rating Regression (LRR) model proposed in [23] to obtain a more "complete" generative model called Latent Aspect Rating Analysis Model (LARAM), which can model the generation of both text data and the overall ratings in a unified framework. As a result, by fitting LARAM to the review data, we can identify latent topical aspects from the review contents as well as discover the latent ratings and weights for each aspect in a review.

The basic assumption in LARAM is that the latent aspects of a particular entity are characterized by a set of coherent topics, which are shared across different reviews discussing the same entity (e.g., "*service*" and "*location*" for a hotel). The topics can be used to identify aspect text segments which contribute to the observed overall ratings in each review via the latent aspect ratings and weights. Along this line, we propose a fully generative framework to capture such dependency between the review contents and overall sentiment ratings.

Based on the notations in Section 3, our generative assumption of a reviewer's rating behavior is as follows: to generate an opinionated review d, the reviewer would first decide the set of aspects $\{A_i\}$ she wants to comment on, and then for each aspect, the reviewer would choose the words with appropriate sentiment polarities to reflect her opinions on the aspects which are characterized by the aspect ratings $\mathbf{s}$. Finally the reviewer would assign an overall rating r based on a weighted sum of all the aspect ratings in her mind, where the weight α reflects the relative emphasis she has placed on each aspect.

According to this generative assumption, we define and combine two components: 1) an aspect modeling module based on statistical topic modeling is introduced to discover the topical aspects from the review contents, and 2) a rating analysis module similar to the Latent Rating Regression Model (LRR) used in [23], is employed to infer the latent aspect ratings and weights based on the aspect segmented review contents.

In the aspect modeling part, we assume an aspect A_i is characterized by a multinomial word distribution $Mul(\epsilon_i)$ over the vocabulary V. The proportion of aspects θ being discussed in each review d is drawn from a Dirichlet distribution $Dir(\gamma)$, where γ postulates the prior distribution of aspects in the whole corpus. Then, a review is treated as a mixture over the latent aspects, and the joint probability of observed word contents $\mathbf{W}$, latent aspect assignments $\{z_n\}_{n=1}^{|d|}$ and aspect proportion θ is defined as follows:

$$p(\mathbf{W}, \mathbf{z}, \theta|\gamma, \epsilon) = p(\theta|\gamma) \prod_{n=1}^{|d|} p(w_n|z_n, \epsilon) p(z_n|\theta) \quad (1)$$

where z_n is an indicator variable representing the latent aspect from which the word w_n is drawn.

In the rating analysis part, aspect rating $\mathbf{s}_i$ is assumed to be determined by the aggregated sentiment over the text segments discussing aspect A_i:

$$\mathbf{s}_i = \sum_{n=1}^{|d|} \beta_{ij} \mathbf{\Delta}[w_n = v_j, z_n = i] \quad (2)$$

where $\beta_{ij} \in \Re$ represents the word j's sentiment polarity on aspect A_i, and $\mathbf{\Delta}[w_n = v_j, z_n = i]$ is an indicator function representing the nth word in review d, which is the jth entry in vocabulary V, is discussing aspect A_i. When we have each aspect's rating, the overall rating r is assumed to be drawn from a Normal distribution with fixed variance δ^2 and mean value as the weighted sum of aspect ratings, i.e., $\alpha^T \mathbf{s} = \sum_{i=1}^{k} \alpha_i \mathbf{s}_i$. Plugging in $\mathbf{s}_i$ defined in Eq (2), we have:

$$r \sim N(\sum_{i=1}^{k} \alpha_i \sum_{n=1}^{|d|} \beta_{ij} \mathbf{\Delta}[w_n = v_j, z_n = i], \delta^2) \quad (3)$$

In order to further capture the diversity of different reviewers' preferences over different aspects and the dependency among the aspects, we employ a multivariate Normal distribution as the prior for aspect weight α, i.e., $\alpha \sim N(\mu, \mathbf{\Sigma})$. As a whole, the probability of observing the overall rating r and aspect weight α given the aspect segments in review d is defined as:

$$p(r, \alpha|\mathbf{W}, \mu, \mathbf{\Sigma}, \delta^2) \quad (4)$$
$$= p(\alpha|\mu, \mathbf{\Sigma}) p(r|\sum_{i=1}^{k} \alpha_i \sum_{n=1}^{|d|} \beta_{ij} \mathbf{\Delta}[w_n = v_j, z_n = i], \delta^2)$$

We can find from the above description of the two components that the dependency between review text content

$\mathbf{W}$ and the latent aspect ratings $\mathbf{s}$ is bridged by the aspect assignment $\{z_n\}_{n=1}^{|d|}$. More specifically, we introduce the latent aspect assignments to associate the appropriate word descriptions with the corresponding aspect, so that the rating regression model can exploit such association to infer the latent aspect ratings/weights. Compared with the previous two-stage approach in [23], where a bootstrapping algorithm is used to generate a hard association, in the unified model, such correspondence is determined by the probabilistic aspect assignments.

Combining the two components, the joint probability of the observed text content $\mathbf{W}$ and overall rating r in a given review d can thus be defined as:

$$\begin{aligned} & P(r, \mathbf{W}|\epsilon, \gamma, \beta, \mu, \mathbf{\Sigma}, \delta^2) \\ =& P(\mathbf{W}|\epsilon, \gamma) \times P(r|\mathbf{W}, \beta, \mu, \mathbf{\Sigma}, \delta^2) \\ =& \iint \prod_{n=1}^{|d|} \sum_{z_n=1}^{k} p(w_n|z_n)p(z_n|\theta)p(\theta|\gamma)d\theta \times \\ & p(r|\sum_{i=1}^{k} \alpha_i \sum_{n=1}^{|d|} \beta_{ij} \mathbf{\Delta}[w_n = v_j, z_n = i], \delta^2)p(\alpha|\mu, \Sigma)d\alpha \end{aligned} \tag{5}$$

where we denote $\mathbf{\Theta} = (\epsilon, \gamma, \beta, \mu, \mathbf{\Sigma}, \delta^2)$ as the set of *corpus-level* model parameters, which do not depend on individual reviews. A graphical model illustration of the unified LARAM is shown in Figure 2.

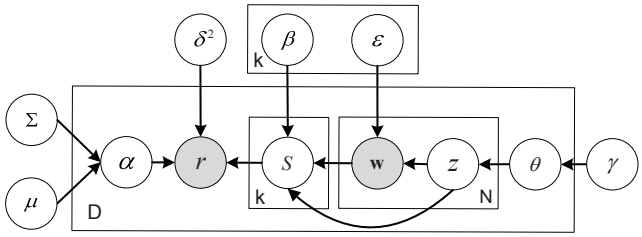


Figure 2: Graphical model representation of LARAM. The outer box represents reviews, while the inner box represents the composition of latent aspect rating regression and word generation within a review.

As illustrated in Figure 2, the word usage patterns embedded in the review text collection are captured by the aspect modeling part; the latent aspect assignments $\{z_n\}_{n=1}^{|d|}$ associate such word clusters with the corresponding aspect, and based on such probabilistic aspect segmentations, the latent aspect ratings and weights are discovered accordingly. We want to emphasize that the aim of the proposed LARAM is not to predict the overall ratings; instead, we are more interested in leveraging the observed general sentiment polarities from overall ratings to identify the opinions in each detailed latent aspect.

We should also note the proposed LARAM is not a trivial replacement of previous keyword-based bootstrapping method in [23] with topic models. Instead, it naturally bridges the gap between the aspect segmentation and sentiment analysis procedure in the previously used two-stage approach. There are two important factors distinguishing the proposed LARAM from the previous LRR model. First, in the current model, the latent aspect rating is a random variable, or more specifically, sum of a set of random variables (because each word's aspect assignment is a probability distribution over all the aspects), while in LRR model, it is treated as a fixed response variable of a rating regression module based on the given aspect segments. This difference enables us to model the uncertainty from the aspect segmentation part as well. Second, the latent rating analysis part provides informative feedback for the aspect segmentation part in the new model. Intuitively, in LARAM, we should assign the word w_j to a proper aspect A_i, in which the word's sentiment orientation β_{ij} is the most consistent with the identified orientation $\mathbf{s}_i$, or in other words, to be consistent with other words in its segment. In the previous LRR model, such segmentation is fixed once the set of aspect keywords is determined.

It would be also interesting to compare LARAM with sLDA introduced in [2]. sLDA is a topic modeling based regression method, which aims at modeling the labeled data in the projected latent topic space. One obvious difference between LARAM and sLDA is that sLDA is designed to predict the overall response for the given input; it can not discover the latent aspect ratings, since it assumes the observed overall response is directly generated from the weighted combination of the latent topic factors. Our LARAM further assumes that the overall rating is determined based on a set of latent ratings over different topical aspects, naturally modeling the latent aspect ratings.

4.1 Posterior Inference

Given the model parameters $\mathbf{\Theta} = (\epsilon, \gamma, \beta, \mu, \mathbf{\Sigma}, \delta^2)$, the key step of applying the proposed LARAM is to infer the posterior distribution of the latent aspect assignment $\{z_n\}_{n=1}^{|d|}$ and aspect weight α in a given review d. Once this is done, the aspect ratings can be easily derived from Eq (2). In this work, we use the variational inference method [10] due to its computational efficiency. The key idea behind variational approximation is to optimize the free parameters of a distribution over the latent variables so that the approximated distribution is close to the true posterior by Kullback-Leibler divergence.

More specifically, we introduce a family of factorized variational distribution for the hidden variables (z, θ, α) in each review d,

$$q(\mathbf{z}, \theta, \alpha|\phi, \eta, \lambda, \sigma^2) = q(\theta|\eta) \prod_n q(z_n|\phi_n)q(\alpha|\lambda, \sigma^2) \tag{6}$$

where the aspect assignment z for each word in d is specified by a k-dimensional multinomial distribution $Mul(\phi)$, aspect proportion θ is governed by a k-dimensional Dirichlet distribution $Dir(\eta)$, and aspect weight α is determined by a k-dimensional multivariate Normal distribution $N(\lambda, \sigma^2)$ with a diagonal covariance matrix to simplify the computation.

Based on the introduced variational distribution, the lower bound of the log-likelihood in review d can be approximated as follows:

$$\begin{aligned} & \log p(r, \mathbf{W}|\epsilon, \gamma, \beta, \mu, \mathbf{\Sigma}, \delta^2) \\ =& \log p(\mathbf{W}|\epsilon, \gamma) + \log p(r|\mathbf{W}, \beta, \mu, \mathbf{\Sigma}, \delta^2) \\ \geq& E_q[\log p(\mathbf{z}, \theta, \mathbf{W}|\epsilon, \gamma)] - E_q[\log q(\mathbf{z}, \theta|\phi, \eta)] \\ +& E_q\big[\log p(r, \alpha, \mathbf{z}|\mathbf{W}, \beta, \mu, \mathbf{\Sigma}, \delta^2)\big] - E_q[\log q(\alpha, \mathbf{z}|\epsilon, \gamma, \phi, \eta)] \end{aligned} \tag{7}$$

where the bound is derived from the Jensen's inequality for convex function. In the following discussions, we will use

$\mathcal{L}_d(\phi, \eta, \lambda, \sigma^2)$ to represent the lower bound defined by a set of variational parameters $(\phi, \eta, \lambda, \sigma^2)$ in review d.

The explanation for this lower bound is quite intuitive: the first part of $\mathcal{L}_d(\phi, \eta, \lambda, \sigma^2)$ represents the objective of aspect modeling module, which aims at finding the optimal aspect assignments $\{z_n\}_{n=1}^{|d|}$ and the corresponding aspect proportion θ for the observed review contents; the second part explains the objective of rating analysis module, which tunes both $\{z_n\}_{n=1}^{|d|}$ and α to fit the overall ratings. These two parts are not independently separated but connected by the common aspect assignments $\{z_n\}_{n=1}^{|d|}$. Both parts attempt to allocate proper aspect assignments to better accommodate the observed text contents and overall rating, respectively.

Details for calculating the expectation of the first part can be found in [3]. The expectation of the complete-data log-likelihood function for the rating analysis module under the variational distribution is derived as:

$$
\begin{aligned}
&E_q\big[\log p(r, \alpha, \mathbf{z}|\mathbf{W}, \beta, \mu, \mathbf{\Sigma}, \delta^2)\big] \\
=&-\frac{(\lambda^T \bar{\mathbf{s}} - r)^2}{2\delta^2} - \frac{1}{2\delta^2}\sum_{i=1}^{k}\Big\{(\lambda_i^2 + \sigma_i^2)Var[\mathbf{s}_i] + \sigma_i^2 \bar{\mathbf{s}_i}^2\Big\} \\
&-\frac{1}{2}(\lambda - \mu)^T\Sigma^{-1}(\lambda - \mu) - \frac{1}{2}Tr\big(diag(\sigma^2)\Sigma^{-1}\big) - \frac{1}{2}\log\delta^2 - \frac{1}{2}\log|\Sigma|
\end{aligned}
$$

where $\bar{\mathbf{s}_i} = \sum_{n=1}^{|d|}\beta_{ij}w_n^j\phi_{ni}$, $Var[\mathbf{s}_i] = \sum_{n=1}^{|d|}(\beta_{ij}w_n^j)^2\phi_{ni}(1-\phi_{ni})$ (w_n^j is a short for the indicator function $\mathbf{\Delta}[w_n = j]$).

Once the expectations in $\mathcal{L}_d(\phi, \eta, \lambda, \sigma^2)$ are analytically determined, an iterative fixed-point method is employed to find the set of variational parameters $(\phi, \eta, \lambda, \sigma^2)$ in order to maximize the lower bound of the original log-likelihood, which in turn would minimize the KL divergency between the variational posterior and true posterior in each review. In particular, we compute the derivative of $\mathcal{L}_d(\phi, \eta, \lambda, \sigma^2)$ with respect to the variational parameters accordingly, and use them to find the optimal setting for each variational parameter.

In the aspect modeling part, the variational parameter η can be easily estimated as

$$\hat{\eta}_i = \gamma_i + \sum_{n=1}^{|d|}\phi_{ni} \tag{8}$$

Due to the involvement of rating analysis part, it is hard for us to get a closed-form solution for ϕ as in [3]. Therefore, we appeal to the gradient-based optimization procedure to obtain the optimal solution:

$$
\begin{aligned}
\hat{\phi}_n = \arg\max_{\phi_n} &\sum_{i=1}^{k} w_n^j\phi_{ni}\Big[\psi(\eta_i) - \psi(\sum_{j=1}^{k}\eta_j) + w_n^j\log\epsilon_{ij} - \log\phi_{ni}\Big] \\
&- \frac{1}{2\delta^2}(\lambda^T\bar{\mathbf{s}} - r)^2 - \frac{1}{2\delta^2}\sum_{i=1}^{k}\Big[(\lambda_i^2 + \sigma_i^2)Var[\mathbf{s}_i] + \sigma_i^2\bar{\mathbf{s}_i}^2\Big]
\end{aligned} \tag{9}
$$

s.t. $\forall$ i, $0 \le \phi_{ni} \le 1$ and $\sum_{i=1}^{k}\phi_{ni} = 1$

In the rating analysis part, to make it comparable across different aspects within the same review, we need to force the aspect weight α to satisfy the constraint that $\forall$ i, $0 \le \alpha_i \le 1$ and $\sum_{i=1}^{k}\alpha_i = 1$. This would postulate additional constraints on the optimization procedures for $\mathcal{L}_d(\phi, \eta, \lambda, \sigma^2)$ with respect to variational parameter λ. We use posterior constrained expectation maximization [5] to address this problem. Gradient-based searching algorithm is utilized to find the optimal solution:

$$
\begin{aligned}
\hat{\lambda} = \arg\min_{\lambda}\Big\{&\frac{1}{2\delta^2}\Big[\sum_{i=1}^{k}\lambda_i^2 Var[\mathbf{s}_i] + (\lambda^T\bar{\mathbf{s}} - r)^2\Big] + \\
&\frac{1}{2}(\lambda - \mu)^T\Sigma^{-1}(\lambda - \mu)\Big\}
\end{aligned} \tag{10}
$$

s.t. $\forall$ i, $0 \le \lambda_i \le 1$ and $\sum_i^k \lambda_i = 1$.

Finally, σ^2 could be easily calculated as,

$$\sigma_i^2 = \frac{\delta^2}{Var[\mathbf{s}_i] + \bar{\mathbf{s}}_i^2 + \delta^2\Sigma_{ii}^{-1}} \tag{11}$$

The interaction between the aspect modeling module and latent rating analysis module is now clearly stated in the above inference procedures: an optimal aspect assignments $\{z_n\}_{n=1}^{|d|}$ should not only fit the observed review contents as much as possible, but also minimize the overall rating prediction error and the variance of each aspect rating prediction; in other words, the rating prediction part can be considered as a regularization factor for the aspect assignments. Meanwhile, the rating analysis part should also consider the variance of each inferred aspect rating, i.e., we should not put too much emphasis on any single aspect, which is highly uncertain.

4.2 Model Estimation

In the previous section, we have discussed how to infer the latent aspect assignments $\{z_n\}_{n=1}^{|d|}$ and aspect weight α in each review d when given the model $\mathbf{\Theta} = (\epsilon, \gamma, \beta, \mu, \mathbf{\Sigma}, \delta^2)$. In this section, we discuss how to estimate these *corpus-level* parameters using the Expectation Maximization (EM) algorithm by maximizing the expectation of observing review contents and the overall ratings in a given review document collection.

As defined in Eq (7), we can approximate the log-likelihood in each individual review d by the introduced variational distribution $q(z, \theta, \alpha|\phi, \eta, \lambda, \sigma^2)$. Since the variational inference is carried out independently for each review in the collection, the log-likelihood over the whole collection D is simply a summation over the lower bound of each review:

$$\mathcal{L}(D) = \sum_{d\in D}\mathcal{L}_d(\phi, \eta, \lambda, \sigma^2) \tag{12}$$

Following similar procedures used in Section 4.1, we maximize Eq (12) by finding the optimal *corpus-level* model parameters $\mathbf{\Theta} = (\epsilon, \gamma, \beta, \mu, \mathbf{\Sigma}, \delta^2)$.

The detailed procedures for updating the parameters in the aspect modeling part is the same as derived in [3], so we only list them here:

$$\epsilon_{ij} \propto \sum_{d}^{D}\sum_{n}^{N_d}\phi_{dni}w_{dn}^j \tag{13}$$

$$\frac{\partial\mathcal{L}(\gamma)}{\partial\gamma_i} = D\Big[\psi(\sum_j\gamma_j) - \psi(\gamma_i)\Big] + \sum_{d}^{D}\Big[\psi(\eta_{di}) - \psi(\sum_j\eta_{dj})\Big] \tag{14}$$

$$\frac{\partial^2\mathcal{L}(\gamma)}{\partial\gamma_i\partial\gamma_j} = D\big[\psi'(\sum_j\gamma_j) - \sigma(i,j)\psi'(\gamma_i)\big] \tag{15}$$

The updating equations for the aspect weight prior $(\mu, \mathbf{\Sigma})$, and overall rating prediction variance δ^2 in the rating analysis part can be easily obtained from their sufficient statistics

accordingly:

$$\hat{\mu} = \frac{1}{D}\sum_{d=1}^{D}\lambda_d \tag{16}$$

$$\hat{\Sigma} = \frac{1}{D}\sum_{d=1}^{D}\left[(\lambda_d - \hat{\mu})(\lambda_d - \hat{\mu})^T + diag(\sigma_d^2)\right] \tag{17}$$

$$\hat{\delta}^2 = \frac{1}{D}\sum_{d=1}^{D}\left\{(r_d - \lambda_d^T\bar{\mathbf{s}}_d)^2 + \sum_{i=1}^{k}\left[(\lambda_{di}^2 + \sigma_{di}^2)Var[\mathbf{s}_{di}] + \sigma_{di}^2\bar{\mathbf{s}}_{di}^2\right]\right\} \tag{18}$$

However a closed-form updating formula for the term weight matrix β is hard to write out:

$$\hat{\beta} = \arg\min_{\beta}\sum_{d}^{D}\left\{(\lambda_d^T\bar{\mathbf{s}}_d - r_d)^2 + \sum_{i=1}^{k}\left[(\lambda_{di}^2 + \sigma_i^2)Var[\mathbf{s}_{di}] + \sigma_{di}^2\bar{\mathbf{s}}_{di}^2\right]\right\} \tag{19}$$

We apply the gradient-based optimization technique to find the optimal solution of β with the following gradients:

$$\frac{\partial\mathcal{L}(\beta_i)}{\partial\beta_{ij}} = \sum_{d}^{D}\Big[(\lambda_d^T\bar{\mathbf{s}}_d - r_d)\lambda_{di} + \sigma_{di}^2\bar{\mathbf{s}}_{di} + (\lambda_{di}^2 + \sigma_{reviewi}^2)\beta_{ij}w_{dn}^j(1-\phi_{dni})\Big]\phi_{dni}w_{dn}^j$$

From the above equation, we can find another evidence of the interaction between aspect modeling module and rating analysis module: an optimal word sentiment polarity setting should not only ensure the sentiment orientation expressed in the review content consist with the observed overall sentiment judgment, but also reduce the uncertainty on each latent aspect's opinion prediction, which would help the model allocate the aspect assignment for each word more accurately.

All the parameters are first randomly initialized to obtain $\Theta_{(0)}$ (subscript indicates the iteration step) and then the following EM algorithm is applied to iteratively update and improve the parameters by alternatively executing the *E-step* and *M-step* in each iteration until the log-likelihood defined in Eq (12) converges.

E-Step: For each review d in the corpus, infer aspect weight α and topic assignments $\{z_n\}_{n=1}^{|d|}$ based on the current parameter $\Theta_{(t)}$ by using Eq (8) to Eq (11) and compute aspect rating $\mathbf{s}$ by Eq (2).

M-Step: Given the sufficient statistics collected from each review in ***E-Step***, find the updated model parameters $\Theta_{(t+1)}$ by using Eq (13) to Eq (19).

5. EXPERIMENT RESULTS

In this section, we first describe the review data sets we used for evaluation purpose, and then discuss both qualitative and quantitative experiment results.

5.1 Data Sets and Preprocessing

We include two review data sets in our experiments: a hotel review data set originally used in [23], and an MP3 player review data set crawled from `www.amazon.com`. In the hotel data, in addition to the overall ratings, reviewers are also asked to provide ratings on 7 pre-defined aspects in each review: *value, room, location, cleanliness, check in/front desk, service, business service* ranging from 1 star to 5 stars. This can serve as the ground-truth for quantitative evaluation of both aspect identification and latent aspect rating prediction. In the MP3 data set, there is only one overall rating in each review, ranging from 1 star to 5 stars. Both data sets are available at `http://timan.cs.uiuc.edu/downloads.html`.

Table 1: Statistics of data sets

	#Item	#Review	#Reviewer	Avg Len	Rating
Hotel	2,232	37,181	34,187	96.5	3.92±1.23
MP3	686	16,680	15,004	87.3	3.76±1.41

Table 2: Topical aspects learned on MP3 reviews

Low Overall Ratings			High Overall Ratings		
unit	jack	service	files	player	vision
usb	headphone	charge	format	music	video
battery	warranty	problem	included	download	player
charger	replacement	support	easy	headphones	quality
reset	problem	hours	convert	button	great
time	player	months	mp3	set	product
hours	back	weeks	videos	hours	sound
work	months	back	file	buds	radio
thing	buy	customer	wall	volume	accessory
wall	amazon	time	hours	ear	fm

We first perform simple pre-processing on these two data sets: 1) remove the reviews with any missing aspect rating or document length less than 50 words (to keep the content coverage of all possible aspects); 2) convert all the words into lower cases; and 3) removing punctuations, stop words defined in [1], and the terms occurring in less than 10 reviews in the collection. After the pre-processing, we have 37,181 hotel reviews and 16,680 MP3 reviews; the detailed statistics are listed in Table 1.

5.2 Aspect Identification

Automatic Adaptation of Aspects: In Amazon reviews, the reviewers are only asked to give an overall rating, so they would have more freedom, or less guidance to write the comments. In this case, it is *very difficult* to pre-specify aspects in keywords. Here, we will qualitatively demonstrate that our unified model, LARAM, can automatically identify meaningful aspects based on the data characteristics. We separate the reviews into two subsets, one with low overall ratings (at most 3 stars) and the other with high overall ratings (at least 4 stars), and run LARAM to extract 20 aspects on each subset. It is expected that users usually comment on different aspects in positive reviews and negative reviews. In Table 2, we show the top 10 words of the highest generation probability for the three aspects with the highest aspect weight prior μ which can be considered as contributing most to the overall ratings. We can see that LARAM automatically adapts to such different data characteristics: in the negative reviews, the most complained aspects are about warranty and service while positive reviews emphasize the good product features such as flexible file format and great video quality.

Next, we quantitatively compare our model with existing methods on the quality of identified topical aspects.

Algorithms for Comparison: Since we depend on the topic modeling techniques to discover the aspects, we compare LARAM with two different topic models: unsupervised **LDA** model and supervised **sLDA** model. LDA behaves similarly as our aspect modeling module, but it can only fit the word co-occurrence patterns in the review content. sLDA extends LDA by adding a regression module to model the observed overall response, so that it uses the same input as our model. However, since sLDA and LARAM employ

Table 3: KL divergence between the align aspects

	LDA	sLDA	LARAM
7 topics	**5.675**	14.878	5.827
14 topics	8.819	19.074	**8.356**
21 topics	12.745	22.411	**11.167**

different generation assumption for the overall response, it would be interesting to compare these two models.

Measure: In the hotel data set, since TripAdvisor asks reviewers to rate the predefined 7 aspects, it is reasonable to assume those are the major aspects most reviewers comment about. Thus, we use the full set of keywords (in total 309 words) generated by the bootstrapping method used in [23] as a prior to train a LDA model on this data set, and treat the learned topics as the "ground-truth" aspect descriptions. Then we train all the three models *without* any supervision of aspect keywords, find the optimal alignment between the learned topics with "ground-truth" aspects by Kuhn-Munkres algorithm [13] and quantitatively measure the quality of the identified aspects using KL divergence:

$$D(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

where p(x) is the "ground-truth" word distribution, and q(x) is the word distribution learned by a given model.

Result Analysis: We train these three models with 7, 14 and 21 topics (since we already know there are 7 aspects) on the hotel data set separately and list the results in Table 3. From the results, we can find that compared with the unsupervised LDA model, the aspects identified by LARAM are closer to the ground-truth aspects when we have more topics (i.e., smaller KL divergence); sLDA's performance is the worst, even though it has additional information from the overall ratings. From the comparison we can see that the inferred aspect ratings from LARAM help the aspect model to better allocate the words across different aspects, which would not be easily distinguished solely from the co-occurrence patterns (LARAM versus LDA). The sLDA model assumes that the topics (aspects) directly characterize the overall ratings, rather than the words' sentiment polarities in the specific aspects as we assumed in LARAM, so it prefers rating sentiment sensitive words than those general content words. As a result, the word distribution under each topic learned by sLDA is quite different from the keyword specified aspects. Besides, we can observe that when we use more topics, KL divergence gets larger. The reason is that with more topics, all the models will then have more freedom to distinguish the aspects in a finer granularity, which is not covered by the predefined aspect keywords.

5.3 Aspect Rating Prediction

Although LARAM is designed to infer ratings on any discovered topical aspects, we can only quantitatively evaluate the predicted ratings on the collection where we have ground-truth aspect ratings from the users. In this experiment, we use the hotel review data set as the testing corpus. In order to ensure our discovered aspects are aligned with the pre-defined aspects, we use the full set of keywords as prior to guide the aspect modeling part.

Algorithms for Comparison: As an alternative method to the proposed unified method **LARAM**, we can take a two-stage approach: apply topic models (e.g., LDA or sLDA) or bootstrapping to identify the aspect segments and then apply LRR to predict aspect ratings. Therefore we include three methods for the comparison purpose, i.e., **LDA+LRR**, **sLDA+LRR**, and **Bootstrap+LRR**.

Table 4: Aspect rating prediction performance on reviews

	LDA+LRR	sLDA+LRR	LARAM
MSE	2.130	2.360	**1.234**
ρ_{aspect}	0.080	0.079	**0.228**
Mis_{aspect}	0.439	0.439	**0.387**
$nDCG_{aspect}$	0.860	0.886	**0.901**
ρ_{hotel}	0.558	0.450	**0.622**
MAP_{hotel}@10	0.427	**0.437**	0.436

Measures: We quantitatively evaluate the algorithms using six different measures, including: (1) Mean Square Error (MSE_{aspect}) of the predicted aspect ratings compared with the ground-truth aspect ratings; (2) Pearson correlation inside reviews (ρ_{aspect}) measures how well the predicted aspect ratings can preserve the relative order of aspects within a review given by their ground-truth ratings; (3) percentage of mis-ordered aspects inside reviews (Mis_{aspect}) measures the cases when the predicted aspect ratings confuse the best and worst aspects within reviews (if they are different as in ground-truth); (4) nDCG of aspect ranking inside reviews ($nDCG_{aspect}$) evaluates the model's ranking accuracy of aspects inside reviews, where the ground truth aspect ratings are used as the graded relevance in the measure; (5) Pearson correlation across hotels (ρ_{hotel}) measures how well the predicted aspect ratings (average over the predicted aspect ratings of all the reviews commenting on this hotel) can preserve the relative order of hotels by their ground-truth ratings; and (6) Mean Average Precision (MAP_{hotel}@10) evaluates the model's ranking accuracy of hotels. We treat each aspect as a query, the top 10% of hotels ranked by the ground-truth aspect ratings as the relevant answers, and test whether we would be able to rank these top 10 hotels on the top, if we use the predicted aspect ratings to rank them. Those measurements can be categorized into two different groups: aspect-level evaluation, including the first four metrics, which are averaged over all the reviews, and hotel-level evaluation, including the last two metrics, which are averaged over all the aspects.

Result Analysis (1): Since we are only interested in predicting the latent aspect ratings, we used all the data for both training and testing. We report the performance of running different models on individual reviews in Table 4, where we highlight the best performance in each metric. We did not include Bootstrap+LRR in this table, because only a small subset of reviews can be tagged with all 7 aspects using the bootstrapping method, but the second-stage LRR requires all reviews to be tagged with 7 aspects as input.

In general, LARAM outperforms other methods in all measures except that its MAP_{hotel}@10 performance is basically the same as sLDA+LRR. The top part of the table shows MSE, which directly measures the difference between the predicted aspect ratings and ground truth ratings. The middle part shows the performance of ranking different

aspects inside reviews. The absolute values of ρ_{aspect} are generally low because this measure is over-penalizing all numerical regression methods which produce ratings in real value while ground-truth ratings are all integers. This bias is eliminated in $nDCG_{aspect}$ which also measures the aspect ranking accuracy but handles the integer tie cases well. Mis_{aspect} only looks at the most confident cases where reviewers show explicit preference of one aspect to another. All three measures show superior performance of the proposed unified model in aspect ranking inside reviews which can answer questions like "Do the reviewer like the *location* better than *cleanliness* of hotel XXX?". Finally, the bottom part of the table demonstrates the model's ranking capability of hotels based on the predicted aspect ratings. Since we average the aspect ratings from individual reviews to get aspect ratings for the same hotel, which are in real value, there is no bias in the ρ_{hotel} measure.

Table 5: Aspect rating prediction performance on h-reviews

	Bootstrap+LRR	LARAM	LARAM+LRR
MSE	1.617	1.589	**0.947**
ρ_{aspect}	0.322	0.197	**0.445**
Mis_{aspect}	0.298	0.318	**0.239**
$nDCG_{aspect}$	0.889	0.905	**0.947**
ρ_{hotel}	0.697	**0.767**	0.764
MAP_{hotel}@10	0.599	**0.627**	0.590

Result Analysis (2): In order to compare with Bootstrap+LRR, we perform another set of experiments where we concatenate all the reviews commenting on the same hotel together as a new review (we call it "h-review") and average the overall/aspect ratings over them as the ground-truth ratings for this hotel.

The results are shown in Table 5. We observe mixed results when comparing Bootstrap+LRR with our unified LARAM model: LARAM is worse in the middle part which measures the aspect ranking performance within reviews while better in MSE_{aspect} (top part) and hotel ranking (bottom part). To explain this behavior, we analyzed the results from aspect modeling component of LARAM and found that the bag-of-words assumption affected the aspect tagging results of general sentiment words a lot[1]. For example, if a word like "nice" appears in a review, LARAM can hardly associate it with the correct aspect with regard to the local context, or mistakenly assign it to a fixed aspect only because they co-occur more often. And the situation gets even worse when we concatenate different reviews together. In contrast, by considering the adjacency of words or sentence boundaries, we will know that "nice" should contribute to the aspect "*location*" if they are in the same sentence. To test this hypothesis, we first use the topics learned using LARAM to annotate sentences and then apply LRR on the tagged sentences. We call this method LARAM+LRR and show its performance on the last column in Table 5, where the highlighted numbers are the best results on each measure. We can see that by tagging each whole sentence with one aspect, LARAM+LRR provides the best performance in almost all measures. It also validates that the topical aspects discovery by LARAM are effective.

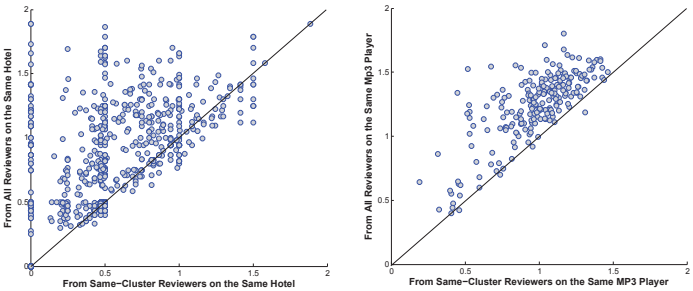
[1]A sample annotation result can be found in `http://sifaka.cs.uiuc.edu/~wang296/Data/annotation.html`

5.4 Aspect Weights Prediction

As we discussed before, a significant advantage of our unified model over existing joint models of aspect and sentiment is that we can infer the relative wights that reviewers have placed on different aspects, i.e., α. Unfortunately, it is infeasible to evaluate the inferred aspect weights directly because we cannot obtain the ground-truth weights from the reviewers. As a result, we choose to evaluate the weights indirectly through clustering the reviewers based on their weights on different aspects.

More specifically, we apply k-means on the aggregated aspect weights of each reviewer over all her reviews to get 10 clusters, where reviewers in the same cluster are expected to share similar taste if the clustering based on the inferred weights is meaningful. In other words, we assume reviewers who share similar aspect preference tend to give the same entity (hotel/MP3 player) similar overall ratings. We test this hypothesis by looking at two sets of standard deviations of ground-truth overall ratings: the first set is for reviewers in the same cluster who also reviewed the same entity while the second control set is for all the reviewers who reviewed the same entity. To make the comparison stable and accurate, we filter out the entities with less than 10 reviews and get 884 hotels with 23,870 reviews and 222 MP3 players with 11,012 reviews in the testing set.

In Figure 3, we show the scatter plot of the two sets of standard deviations. It is clear that most standard deviations of the first set (reviewers from the same cluster on the same entity) are smaller than those of the second control set (all the reviewers on the same entity). Furthermore, two-sample Kolmogorov-Smirnov (KS) test of the samples in the two sets of standard deviations indicates that the difference between the two groups is statistically significant with p-value of e^{-10} for both data sets. This means that reviewers clustered together based on similar aspect weights tend to give more similar overall ratings to the same entity. It also indirectly suggests that the aspect weights learned by our unified model capture the taste of different reviewers.



(a) Hotel (b) MP3

Figure 3: Scatter plot of standard deviations of overall ratings

6. CONCLUSIONS

In this paper, we propose a unified generative Latent Aspect Rating Analysis Model (LARAM) which can explore review text data with companion overall ratings to simultaneously discover: 1) latent topical aspects, 2) latent ratings on each identified aspect, and 3) latent weights placed on different aspects by a reviewer. It is a fully generative model

for both the review text and the overall rating, and enables LARA task on arbitrary review data without needing aspect keyword supervision. Our empirical experiments on a hotel review data set and an MP3 player review data set show that the proposed LARAM can effectively solve the problem of LARA, including automatically identifying meaningful topical aspects, inferring interesting differences in aspect ratings within reviews, and modeling users' preferences with the inferred relative emphasis on different aspects. Such detailed analysis of opinions at the level of topical aspects enabled by LARAM can support multiple application tasks, including aspect opinion summarization, ranking of entities based on aspect ratings, and analysis of reviewers rating behavior.

As we have observed in the detailed experiment results, the bag-of-words assumption seriously hampers the model's aspect segmentation capability, which provides inaccurate segments for the later rating analysis part. For our future work, we are interested in restricting the naive bag-of-words assumption by adding sentence boundary and proximity information into the model for better associating general sentiment words with the appropriate aspect, which will lead to better aspect rating prediction performance. Also, it would be interesting to study how to alleviate the data sparseness problem because reviewers usually do not comment on all the aspects for a given entity. In this situation, borrowing some ideas from the collaborative filtering problem would be beneficial.

7. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant Numbers IIS-0713581 and CNS 1028381, and by an AFOSR MURI Grant FA9550-08-1-0265.

8. REFERENCES

[1] Onix text retrieval toolkit stopword list. `http://www.lextek.com/manuals/onix/stopwords1.html`.

[2] D. Blei and J. McAuliffe. Supervised topic models. *Advances in Neural Information Processing Systems*, 20:121–128, 2008.

[3] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[4] X. Ding, B. Liu, and L. Zhang. Entity discovery and assignment for opinion mining applications. In *Proceedings of the 15th KDD*, pages 1125–1134. ACM, 2009.

[5] J. Graça, K. Ganchev, and B. Taskar. Expectation maximization and posterior constraints. In *Advances in Neural Information Processing Systems*, volume 20. MIT Press, 2007.

[6] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proceedings of the 10th KDD*, pages 168–177. ACM, 2004.

[7] M. Hu and B. Liu. Mining opinion features in customer reviews. In *AAAI*, pages 755–760. AAAI Press / The MIT Press, 2004.

[8] N. Jindal and B. Liu. Identifying comparative sentences in text documents. In *Proceedings of 29th SIGIR*, 2006.

[9] Y. Jo and A. H. Oh. Aspect and sentiment unification model for online review analysis. In *Proceedings of the fourth ACM international conference on Web search and data mining*, 2011.

[10] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[11] S. Kim and E. Hovy. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, pages 1367–1373. Association for Computational Linguistics, 2004.

[12] C. Lin and Y. He. Joint sentiment/topic model for sentiment analysis. In *Proceeding of the 18th CIKM*, pages 375–384, New York, NY, USA, 2009. ACM.

[13] L. Lovász and M. Plummer. *Matching theory*. Elsevier Science Ltd, 1986.

[14] Y. Lu and C. Zhai. Opinion integration through semi-supervised topic modeling. In *Proceeding of the 17th WWW*, pages 121–130. ACM, 2008.

[15] Y. Lu, C. Zhai, and N. Sundaresan. Rated aspect summarization of short comments. In *Proceedings of the 18th WWW*, pages 131–140, 2009.

[16] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th WWW*, pages 171–180. ACM, 2007.

[17] S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima. Mining product reputations on the web. In *Proceeding of the 8th KDD*, pages 341–349, 2002.

[18] B. Pang and L. Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd ACL*, pages 115–124, 2005.

[19] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 79–86, 2002.

[20] A.-M. Popescu and O. Etzioni. Extracting product features and opinions from reviews. In *HLT '05*, pages 339–346, Morristown, NJ, USA, 2005.

[21] I. Titov and R. T. McDonald. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of the 46th ACL*, pages 308–316, 2008.

[22] I. Titov and R. T. McDonald. Modeling online reviews with multi-grain topic models. In *Proceeding of the 17th WWW*, pages 111–120, 2008.

[23] H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis on review text data: a rating regression approach. In *Proceedings of the 16th KDD*, pages 783–792. ACM, 2010.

[24] 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, 2010.

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