Latent Aspect Mining via Exploring Sparsity and Intrinsic Information[*]

Yinqing Xu Tianyi Lin Wai Lam Zirui Zhou Hong Cheng Anthony Man-Cho So
Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong, Hong Kong
{yqxu,tylin,wlam,zrzhou,hcheng,manchoso}@se.cuhk.edu.hk

ABSTRACT

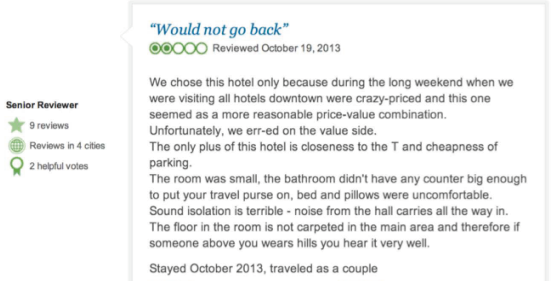
We investigate latent aspect mining problem that aims at automatically discovering aspect information from a collection of review texts in a domain in an unsupervised manner. One goal is to discover a set of aspects which are previously unknown for the domain, and predict the user's ratings on each aspect for each review. Another goal is to detect key terms for each aspect. Existing works on predicting aspect ratings fail to handle the aspect sparsity problem in the review texts leading to unreliable prediction. We propose a new generative model to tackle the latent aspect mining problem in an unsupervised manner. By considering the user and item side information of review texts, we introduce two latent variables, namely, user intrinsic aspect interest and item intrinsic aspect quality facilitating better modeling of aspect generation leading to improvement on the accuracy and reliability of predicted aspect ratings. Furthermore, we provide an analytical investigation on the Maximum A Posterior (MAP) optimization problem used in our proposed model and develop a new block coordinate gradient descent algorithm to efficiently solve the optimization with closed-form updating formulas. We also study its convergence analysis. Experimental results on the two real-world product review corpora demonstrate that our proposed model outperforms existing state-of-the-art models.

Categories and Subject Descriptors

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

Keywords

Topic Model, Sparse Coding, Aspect Mining



"Would not go back"
●●○○○ Reviewed October 19, 2013

Senior Reviewer
9 reviews
Reviews in 4 cities
2 helpful votes

We chose this hotel only because during the long weekend when we were visiting all hotels downtown were crazy-priced and this one seemed as a more reasonable price-value combination.
Unfortunately, we err-ed on the value side.
The only plus of this hotel is closeness to the T and cheapness of parking.
The room was small, the bathroom didn't have any counter big enough to put your travel purse on, bed and pillows were uncomfortable.
Sound isolation is terrible - noise from the hall carries all the way in.
The floor in the room is not carpeted in the main area and therefore if someone above you wears hills you hear it very well.

Stayed October 2013, traveled as a couple

Figure 1: A Sample Hotel Review

1. INTRODUCTION

There has been much research effort on extracting and mining information from review texts, such as sentiment analysis [24, 16], opinion summarization and identification [6, 3, 11]. But most of these models just target for the general sentiment analysis of review texts. In order to provide users more effective detailed insights of different reviews, it is necessary to detect more fine-grained information of the items. To address this task, aspect-based sentiment analysis has been conducted [22, 28, 7, 14] and it led to useful opinion summarization. Aspects are the common attributes or components of an item in a particular domain as exemplified by the aspects such as "Room", "Service" and "Location" of the hotel. On the E-commerce web sites such as Amazon and eBay, users usually write a piece of review text and provide an overall rating for the reviewed item. However, we do not know user's ratings on each aspect, i.e. aspect ratings.

The latent aspect mining task investigated in this paper takes as input a collection of review texts in a particular domain together with a numerical overall rating of each review. The goal is to discover a set of aspects and predict ratings on each aspect for each review in an unsupervised manner. Also, the key terms for each aspect are detected. Note that the aspects are previously unknown and only the number of aspects is required. Figure 1 illustrates a sample hotel review. Such kind of review consists of some text content and an overall rating (e.g. 2-star). Suppose that a collection of such reviews and ratings information is available, and the number of aspects is provided. The aim of latent aspect mining is to discover the aspects including "Room", "Value", "Location", etc. and predict user's ratings on each aspect for each review, e.g. 1-star for the Room aspect and 2-star for the Value aspect. Some key terms such as "standard", "twin" for the Room aspect, "remote", "accessible" for the Location aspect, etc. can also be detected. Recently, Wang et al. [26, 25] have proposed a model called Latent Aspect Rating Analysis Model (LARAM) that can tackle the latent

[*]The work described in this paper is substantially supported by grants from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project Code: CUHK413510) and the Direct Grant of the Faculty of Engineering, CUHK (Project Code: 4055034). This work is also affiliated with the CUHK MoE-Microsoft Key Laboratory of Human-centric Computing and Interface Technologies.

CIKM'14, November 03–07 2014, Shanghai, China.
Copyright 2014 ACM 978-1-4503-2598-1/14/11 ...$15.00.
http://dx.doi.org/10.1145/2661829.2662062.

aspect mining problem. They adopted the classical topic model Latent Dirichlet Allocation (LDA) [1] to model the generation of words in online reviews, and determine the aspect rating based on a rating regression component.

One limitation of probabilistic topic models such as LDA-based models is that they are ineffective when dealing with aspect sparsity in texts [31]. Aspect sparsity refers to the observation that the text content of most reviews only covers some aspects, rather than mentioning all aspects. In fact, it is quite common that real-world reviews exhibit aspect sparsity issue. For example, let us consider the hotel domain with a set of aspects such as Value, Room, Location, Cleanliness, Food, Service, etc. For a particular review, a user typically comments on some aspects and not necessarily all the aspects. Another example refers to a particular hotel which is famous for its delicious food. It is more likely that a typical review of this hotel contains comments on its the Food aspect while some other aspects such as Value and Room. are not mentioned especially for short reviews. The main obstacle for traditional probabilistic topic models such as LDA in LARAM mentioned above to handle aspect sparsity is that topic or aspect proportions are modeled as normalized distributions, namely, the sum of each aspect proportion should be one, so applying a sparsity inducing l_1-regularizer as in lasso [20] is not helpful. As a result, some of the aspect ratings (e.g ratings on the Room aspect and the Value aspect in the example above) predicted by the probabilistic topic model may not be reliable.

Recently, non-probabilistic sparse coding techniques, such as the Sparse Topical Coding (STC) model proposed by Zhu et al. [31], can tackle the above sparsity issue. STC does not require the aspect proportion be the normalized distribution, so it is able to employ a theoretically sound l_1-regularizer to control the aspect sparsity. However, one limitation is that it cannot be directly applied to tackle the latent aspect mining problem since we need to consider more latent variables such as aspect ratings. Incorporating these additional variables into the model may prohibit a closed-form updating formula, compromising computational efficiency especially for large data sets. Another issue of the STC model is that there is no convergence analysis reported for the block coordinate descent algorithm commonly used in Maximum A Posterior (MAP) estimation adopted by the model.

Another observation is that in practical situations, we can easily collect side information of the reviews such as user and item information. For example, it is easy to obtain all the reviews written by a particular user, or all the reviews associated with the same item. Such user and item information can be exploited to improve the latent aspect mining problem. Existing models for this problem do not explore such information.

We investigate the latent aspect mining problem. The input data of this problem consists of a collection of reviews in a particular domain with a numerical overall rating associated with each review. One goal is to discover a set of aspects which are previously unknown for the domain, and predict the aspect ratings for each review. Another goal is to detect key terms for each aspect. We propose a new generative model that can tackle the latent aspect mining problem in an unsupervised manner. It is capable of alleviating the aspect sparsity issue when predicting aspect ratings. Our proposed model, known as Sparse Aspect Coding Model (SACM), is a new model employing l_1-regularizer to control the sparsity on the aspect proportions. In addition, we consider user and item side information of review texts. Such information can facilitate better modeling of aspect generation leading to improvement on the accuracy and reliability of predicted aspect ratings. Specifically, we introduce two notions, namely, *user intrinsic aspect interest* and *item intrinsic aspect quality*, which are modeled as latent variables in our model. User intrinsic aspect interest captures the intrinsic interest for each aspect of a particular user. Item intrinsic aspect quality represents the intrinsic quality for each aspect of a particular item. In addition to aspect rating prediction, our proposed model is able to detect key terms for each aspect.

We make use of MAP technique to find the solution. Instead of directly applying block coordinate descent algorithms as in STC, we first conduct analytical investigation on the MAP optimization problem and develop a new algorithm called *block coordinate gradient descent* algorithm with a closed-form formula to iteratively update the solution. We also study its convergence analysis. This new algorithm allows our model to process the text data efficiently.

Experimental results on two different real-world product review corpora demonstrate that our proposed model outperforms existing state-of-the-art models.

Our contributions in this paper can be summarized as follows:

- We propose a new model for tackling the latent aspect mining problem in an unsupervised manner. This model is capable of alleviating the aspect sparsity issue via a new modeling and derivation extended from the sparse topical coding method.

- We incorporate user and item side information of review texts via the design of two notions, namely, *user intrinsic aspect interest* and *item intrinsic aspect quality* which are modeled as latent variables.

- We conduct an analytical investigation on the MAP optimization problems used in our proposed model and propose a new block coordinate gradient descent algorithm to solve the MAP optimization with closed-form updating formulas. We also study its convergence analysis.

- We demonstrate the efficacy of user intrinsic aspect interest and item intrinsic aspect quality discovered from the model for supporting user and item characterization.

2. RELATED WORK

There have been much efforts on sentiment analysis for online reviews. One category is to determine whether a review is positive or negative [24, 17, 3]. Another category is to classify online reviews into multi-point sentiment scale [16]. However, all the above models above just conduct overall sentiment analysis and do not explore fine-grained aspects.

There have been some works on extracting aspect terms from review texts. Titov et al. [22] proposed a model called MG-LDA to automatically extract the ratable aspects. Mukherjee et al. [15] applied the user provided seed words of a few aspect categories to jointly extract and cluster aspect terms by a semi-supervised model. Chen et al. [2] exploited the prior domain knowledge to generate coherent aspects.

Some research efforts have been conducted on aspect-level sentimental opinion mining. Mei et al. [13] introduced sentiment in discovering the facets and also positive/negative opinions. Later, Titov and McDonald [21] extended their multi-grain topic model to extract aspect-specific topics. Lin et al. [10] proposed a sentiment/topic joint model called JST to extract the aspect and its corresponding sentiment polarity. However, it is still not informative enough to identify the sentiment orientation or predict ratings on each topical aspect of a particular item, especially for large review corpora. In [26], Wang et al. aimed at inferring the user's ratings and also relative weights on each aspect based on the review text and overall ratings. To tackle this problem, they have proposed two models. One model, known as Latent Rating Regression (LRR), models the overall rating by applying two-fold linear regression model for aspect rating and aspect weight, based on the user specified seed terms for each aspect. Their second model, known as Latent Aspect Rating Analysis Model (LARAM), is a unified generative model and it does not need to predefine the aspect seed terms. Nevertheless, the above models based on probabilistic topic models fail to handle the aspect sparsity issue.

The sparsity-enhanced models have been widely used in different applications. Yang et al. [29] extended the popular spatial pyramid matching model and proposed a linear SPM kernel based on SIFT sparse codes. Shashanka et al. [19] applied an entropic prior in Maximum A Posterior estimation to enforce sparsity based on the Probabilistic Latent Semantic Analysis. Zhu et al. [31] improved the traditional probabilistic models by incorporating the sparse coding idea to discover sparse latent representations for each document. Later, Zhu et al. [30] presented another model called Conditional Topical Coding which is enhanced by incorporating rich language features in text.

Recently, there have been some works on considering the user and item side information to conduct sentiment analysis for online reviews. Li et al. [9] explored the reviewer and product information to predict the overall rating of each review. Wang et al. [27] proposed a supervised topic model to label the prediction for each review with consideration of user and item information.

3. PROBLEM DEFINITION

We provide the problem definition for the latent aspect mining problem investigated in this paper. The input of the latent aspect mining problem consists of a collection of review texts in a particular domain. For each review text, it is also associated with a numerical overall rating. One goal is to discover the set of previously unknown aspects for the domain and predict the ratings on each aspect for each review. It only requires to specify the total number of aspects. Another goal is to detect key terms for each aspect.

Reviews are written by users to share opinions about their reviewed product items. For a particular domain, the input review corpus is represented as $\mathcal{D} = \{d_1, d_2, ..., d_{|D|}\}$. We use $\mathcal{U} = \{1, ..., U\}$ and $\mathcal{H} = \{1, ..., H\}$ to denote the user collection and item collection. Typically, we assume that the review $d \in \mathcal{D}$ is written by the user $u_d \in \mathcal{U}$ for the item $h_d \in \mathcal{H}$. Also, the overall rating, denoted by $Y_d \in \mathbb{R}_+$, is given by the user to express his overall satisfaction for the reviewed item. Normally, this rating value is a numerical integer value and it commonly ranges from 1 to 5 star.

An aspect represents the common attributes or components of the product item in a particular domain. For example, "Service", "Room" aspects for the hotel domain, "Flavor", "Location" for the restaurant domain, etc.. Let K be the total number of aspects in a particular domain. We use $\mathcal{A} = \{1, 2, ..., K\}$ to denote the set of aspects that has been commented in the review corpus. Each aspect is denoted by $k \in \mathcal{A}$. An aspect rating is the user's fine-grained rating on each aspect of the reviewed item, e.g. "3-star Service" and "5-star Room" for a hotel. For the review $d \in \mathcal{D}$, the aspect ratings are represented by a $K-$dimensional vector $Y_d^A \in \mathbb{R}_+^K$.

4. OVERVIEW AND BACKGROUND OF OUR MODEL

4.1 Overview of Our Model

We propose a new generative model that can tackle the latent aspect mining problem in an unsupervised manner. This model is capable of alleviating the aspect sparsity issue when predicting the aspect ratings. The aspect sparsity issue has been discussed in Section 1. Our proposed model, known as SACM, controls the sparsity of aspect proportions by means of l_1-regularizer, and generates the aspect ratings by considering user and item intrinsic information. We introduce two notions, namely, *user intrinsic aspect interest* and *item intrinsic aspect quality*, which are modeled as latent variables in our model. User intrinsic aspect interest denotes the intrinsic interest for each aspect of a particular user. This notion is different from the notion of aspect weight defined in [26]. Specifically, aspect weight represents the user's emphasis placed on each aspect when the user decides the overall rating. It varies with different review texts. User intrinsic aspect interest is not item dependent while aspect weight is item dependent. For example, consider a foodie user who has a great interest on the Food aspect in the hotel domain. This user's reviews will mainly comment on this aspect no matter for which hotel. Likewise, if a user has no interest on the Food aspect, his/her reviews do not likely mention this aspect. Item intrinsic aspect quality represents the intrinsic quality for each aspect of a particular item, and it is not user dependent. For example, for a five star hotel, the intrinsic quality for most of its aspects will be superior than that of lower star hotels except for the price.

One characteristic of our proposed SACM is that the sparsity of aspect proportion in a particular review can be handled more effectively via the modeling of user intrinsic aspect interest and item intrinsic aspect quality. In general, it can be observed that if a particular aspect is not mentioned in a review, it is essentially due to two main reasons. The first reason is that the user has no interest on that aspect. The second reason is that the concerned aspect of a particular item is not so distinctive that such aspect is normally ignored when a user writes the review for that particular item. Another characteristic of SACM is that the aspect rating is modeled by a Gaussian distribution with the mean related to item intrinsic aspect quality, and the variance related to user intrinsic aspect interest. It can be observed that the aspect rating of a particular item from a large number of users should attain an average value determined by the item intrinsic aspect quality, and the variance of such aspect rating is related to the user intrinsic aspect interest. For example,

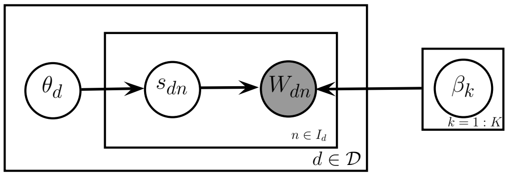


Figure 2: Sparse Topical Coding Model

in the hotel domain, when a foodie user, who is sensitive to the Food aspect, has written reviews for a number of different hotels, his ratings on the Food aspect of each hotel should exhibit some variations. The degree of sensitivity depends on his/her intrinsic interest. On the contrary, a user, who never cares about food, would exhibit much less variations in aspect ratings on the Food aspect in this user's reviews. Thus, the variance of aspect rating is related to user intrinsic aspect interest.

In addition to aspect rating prediction, our proposed model can also detect the key terms for each aspect. The learned (aspect) dictionary within our proposed model contains terms that are associated with each aspect together with the association strength.

4.2 Sparse Topical Coding Background

Recently, Zhu et al. [31] have proposed a Sparse Topical Coding (STC) model for discovering the hidden topic representations from a collection of documents. Unlike traditional topic models, it can directly control the sparsity of the inferred representations by sparsity-inducing regularizers such as l_1 regularizer. Figure 2 depicts the graphical model of STC. Each circle represents a variable. The shaded circles represent the observed variables and non-shaded circles are the hidden variables to be inferred. The inner rectangle plate denotes the replication for the word in each document, and the outer rectangle plate is the replication for a document. The arrows capture the dependency among the variables. STC models the observed text words in each document by latent variables including word code s, document code θ and the dictionary β.

For a particular document $d \in \mathcal{D}$, the document code $\theta_d \in \mathbb{R}_+^K$ is a K-dimensional vector, where the component θ_{dk} represents the document's association strength regarding the topic k. For example, the sample document code (0.5, 2.3, 1.4, 3.4, 0.0) indicates that this document mainly focuses on the second, third and fourth topic but hardly mention the first and fifth topic. Different from the topic distribution in traditional probabilistic topic models, the sum of each component of the document code does not require to be one, so l_1 regularizer can be applied to enforce sparsity for the document code, i.e. some components of the document code equal zero. Similarly, the word code s_{dn} is also a K-dimensional vector, and the kth component s_{dnk} captures the association strength on the topic k for the word n in the document d. The sum of the component in the word code does not need to be one as well. Hence, this word code for each word is also different from the topic assignment. In traditional probabilistic topic models, topic assignment just assigns each word to one of the predefined topics while word code can let each word belong to multiple topics with varying degrees. $\beta \in \mathbb{R}_+^{K \times N}$ is a dictionary[1] with K bases and the vocabulary size of N. It is a global matrix and document independent. Each row $\beta_{k\cdot}$ represents an topical basis with a unigram distribution over the vocabulary V. In other words, $\beta_{k\cdot}$ belongs to a (N-1)-simplex. Essentially, the document d is projected to a semantic space spanned by the topical bases in the dictionary β.

Assume that $V = \{1, ..., N\}$ is the vocabulary with N words. We model each document $d \in \mathcal{D}$ as a vector $(w_{d1}, ..., w_{dn_d})^T$, where $n_d = |I_d|$. I_d is the index set of the appearing words in the document d and w_{dn}, where $n \in I_d$, denotes the number of occurrences, namely the word count, of the word n in the document d. The basic generative process for the words in the document $d \in \mathcal{D}$ is as follows: we first sample the document code from the prior $p(\theta_d)$, and sample the word code s_{dn} from $p(s_{dn}|\theta_d)$ for each observed word n, where n is the word index in vocabulary. Finally, we sample the observed word count w_{dn} from a distribution with $s_{dn}^T \beta_{\cdot n}$ as the mean, where $\beta_{\cdot n}$ represents the n-th column of β. The joint distribution is defined as:

$$p(\theta_d, s_d, \{w_{dn}\}_{n \in I_d} | \beta) = p(\theta_d) \sum_{n \in I_d} p(s_{dn}|\theta_d) p(w_{dn}|s_{dn}, \beta) \quad (1)$$

Normally, the word count in each document is assumed to be sampled from a Poisson distribution. For the sparsity of θ_d and s_{dn}, the document code is induced by the Laplace prior, and the word code is drawn from the supergaussian. The specific formulations are shown in Section 5.1.

STC employs the Maximum A Posterior (MAP) estimation method to infer these set of hidden variables. We represent the collection of document code and word code as $\mathbf{\Theta}$ and $\mathbf{S}$, respectively, i.e. $\mathbf{\Theta} = \{\theta_d\}_{d \in \mathcal{D}}$, $\mathbf{S} = \{s_{dn}\}_{d \in \mathcal{D}, n \in I_d}$. The hidden variable set can be represented as $\Omega = \{\mathbf{\Theta}, \mathbf{S}, \beta\}$. The observed data is the text words $\{w_{dn}\}_{d \in \mathcal{D}, n \in I_d}$. The goal is to infer hidden variable set Ω conditioned on the observed data. The MAP objective function of STC can be formulated as follows:

$$\hat{\Omega}_{MAP} = \arg\max_{\Omega} p(\Omega | \{w_{dn}\}_{d \in \mathcal{D}, n \in I_d}) \quad (2)$$

The block coordinate descent algorithm is usually employed to solve the objective function above.

5. OUR PROPOSED MODEL - SACM

5.1 Model Description

As mentioned in Section 4, our proposed model, known as Sparse Aspect Coding Model (SACM), incorporates two latent variables, namely, user intrinsic aspect interest t_u and item intrinsic aspect quality q_h when modeling the observed review text and overall rating. User intrinsic aspect interest t_u for the user $u \in \mathcal{U}$ represents this user's intrinsic interest for each aspect. Item intrinsic aspect quality q_h denotes the intrinsic quality of the item $h \in \mathcal{H}$ for each aspect, which is user independent. More description for these two notions can be found in Section 4. The generative process is as follows: One would first choose the subset of all aspects for giving comments and decide the text proportion for describing each aspect based on the user intrinsic aspect interest t_u and item intrinsic aspect quality q_h. Then, some terms including opinionated words would be selected to form the review content. The details of the generation process of a

[1] Note that β is called a dictionary in the sparse coding area historically. It is different from the concept of dictionary in the IR community. To avoid confusion, we call it "aspect dictionary" instead of "dictionary" in this paper.

word will be described below. Next, the sentimental orientation for each aspect characterized by the aspect rating is determined. Finally, the observed overall rating given by this user will be based on the weighted sum of aspect ratings.

The graphical model of SACM is depicted in Figure 3. The outer rectangle plate represents the replication for a review. The inner rectangle plate captures each word in each review. There are two components in this model. The first component shown on the lower left is related to the review text content component including θ_d, s_{dn} and w_{dn}. The second component shown on the upper right is related to the rating mining component.

We first describe the review text content component which uses a variant of STC mentioned in Section 4.2 to generate the observed words. For a particular review $d \in \mathcal{D}$ written by the user $u_d \in \mathcal{U}$ for the item $h_d \in \mathcal{H}$, the document code θ_d is modeled as the Hadamard product between the user intrinsic aspect interest t_{u_d} and the item intrinsic aspect quality q_{h_d} instead of Laplace prior. Precisely, the kth element of the document code θ_{dk} represents the association strength on the aspect k. Also, the more the word occurrence over the kth aspect, the higher the value of θ_{dk} is. Specifically, the dominated aspect proportions in a review mainly depend on the corresponding t_{u_d} and q_{h_d}. For instance, in the hotel domain, a user who likes delicious food will have high $t_{u_d k}$ where the aspect k is the Food aspect. This user likely provides opinions on food in detail in his/her reviews leading to a high value of θ_{dk}. Additionally, a hotel possessing distinctive environment, i.e. q_{hk} is high where k is the Environment aspect, is likely to draw attention from users by its environment. Thus, it tends to attract some comments on this aspect. As a result, the corresponding θ_{dk} also has a high value. The above examples show us that both t_{u_d} and q_{h_d} contributes to θ_d. Based on the above motivation, we use Eq. (3) below to generate the aspect proportion, which is modeled by the document code θ_d for review d,

$$\theta_d = t_{u_d} \circ q_{h_d} \tag{3}$$

where the operator $\circ$ is the Hadamard product, which is defined as the entry-wise product between the vector t_{u_d} and the vector q_{h_d}.

It is reasonable that the user intrinsic aspect interest t_u, $u \in \mathcal{U}$ is drawn from the Laplace prior, i.e. $p(t_u) \propto exp(-\lambda \|t_u\|_1)$. Specifically, a user usually will not be interested in all possible aspects of a particular item. Then, we use the STC model to generate the observed review text. After obtaining the document code θ_d, we sample the word code s_{dn} from $p(s_{dn}|\theta_d)$ for each observed word n, where n is the word index in vocabulary, and sample the observed word count w_{dn} from a distribution with $s_{dn}^T \beta_{\cdot n}$ as the mean, where $\beta_{\cdot n}$ represents the n-th column of β. Unlike the multinomial distribution adopted in traditional probabilistic topic models, for the sparsity of word code, s_{dn} is drawn from the super-gaussian as shown below. The l_1-norm within them tends to find sparse codes.

$$p(s_{dn}|\theta_d) \propto exp(-\gamma \|s_{dn} - \theta_d\|_2^2 - \rho \|s_{dn}\|_1) \tag{4}$$

Then, the word count in each document is sampled from the Poisson distribution $p(w_{dn}|s_{dn}, \beta) = Poiss(w_{dn}; s_{dn}^T \beta_{\cdot n})$.

In the rating mining component, we define the aspect weight represents the user's relative weight placed on each aspect when the user decides the overall rating for a particular review. For the review d, we assume that aspect weight $\eta_d \in \mathbb{R}_{++}^K$ is generated by the document code θ_d, which denotes the aspect strength in each aspect. After normalization, we have each element of η_d as follows:

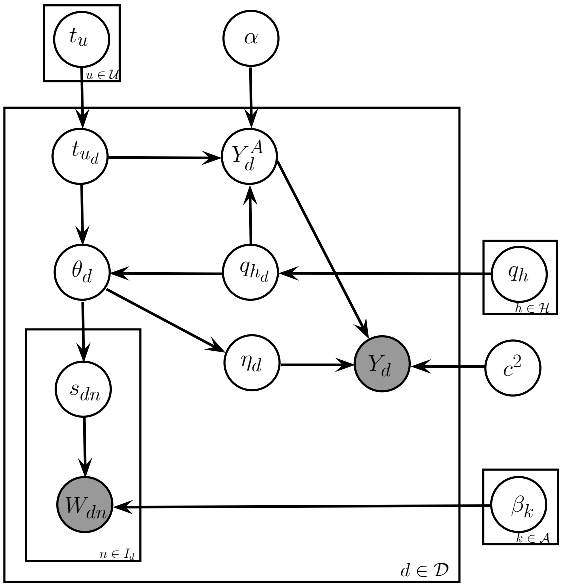


Figure 3: Our Proposed Model - SACM

$$\eta_{dk} = \frac{exp(\theta_{dk})}{\sum_j exp(\theta_{dj})} \tag{5}$$

For a review d written by the user u_d for the item h_d, we assume that the k-th element of the aspect rating Y_{dk}^A is drawn from a Gaussian distribution. The mean and variance are assumed to be $q_{h_d k}$ and $\alpha^2 t_{u_d k}^2$ respectively where α is a positive scaler.

$$Y_{dk}^A \sim N(q_{h_d k}, \alpha^2 t_{u_d k}^2) \tag{6}$$

Consequently, the ratings on the kth aspect Y_{dk}^A from all reviews for a particular item h_d should attain the average value determined by the intrinsic aspect quality q_{h_d} of this item h_d. For a particular user u, the variance for his/her aspect ratings should be related to this user's intrinsic aspect interest t_u. For example, in the hotel domain, a foodie person is likely to write more about the Food aspect in the reviews, and this user would be more sensitive about the variation of the Food aspect in different hotels. Thus, he would give ratings on the Food aspect with higher variance. Another example is that a thrifty person would be more sensitive to the Price aspect and tends to provide a wider range of ratings for the Price aspect for different hotels. But for other aspects, this user does not care much and the ratings on them would exhibit much less variance.

Finally, as the generative process mentioned above, we assume that the overall rating Y_d of the review d is drawn from a Gaussian distribution. The weighted sum of aspect ratings $\eta_d^T Y_d^A$ is the mean and c^2 is a fixed variance, i.e. $Y_d \sim N(\eta_d^T Y_d^A, c^2)$.

Since the user intrinsic aspect interest is modeled by a Laplace prior, we employ the Maximum A Posterior (MAP) to estimate all the latent variables in this model. Let $\mathbf{T}$ and $\mathbf{Q}$ be the collection of user intrinsic aspect interest and item intrinsic aspect quality respectively, i.e. $\mathbf{T} = \{t_u\}_{u \in \mathcal{U}}$, $\mathbf{Q} = \{q_h\}_{h \in \mathcal{H}}$, and we represent the collection of word codes and aspect ratings as $\mathbf{S} = \{s_{dn}\}_{d \in D, n \in I_d}$ and $\mathbf{Y} = \{Y_d^A\}_{d \in \mathcal{D}}$, respectively. Our goal is to infer the latent variable set Ω where $\Omega = \{\mathbf{Y}, \mathbf{S}, \mathbf{T}, \mathbf{Q}, \beta, \alpha\}$. The objective function is the negative logarithm of the posterior $p(\Omega | \{w_{dn}, Y_d\}_{d \in \mathcal{D}, n \in I_d})$. Combining (3) to (6), and the review text content component, the optimization problem based on MAP estimation is given as follows:

$$
\begin{aligned}
\min_{\Omega} \quad & \sum_u \lambda \|t_u\|_1 + \sum_d \sum_{n \in I_d} (\gamma \|s_{dn} - \theta_d\|_2^2 + \rho \|s_{dn}\|_1) \\
& - \sum_d \sum_{n \in I_d} [(w_{dn} \log(s_{dn}^T \beta_{\cdot n})) - s_{dn}^T \beta_{\cdot n}] \\
& + \sum_d \frac{1}{2c^2} (Y_d - \sum_k \eta_{dk} Y_{dk}^A)^2 + \sum_d \sum_k [\log(\alpha t_{u_d k}) \\
& + \frac{1}{2\alpha^2 t_{u_d k}^2} (Y_{dk}^A - q_{h_d k})^2] \\
\text{s.t.} \quad & t_u \geq 0, q_h \geq 0, s_{dn} \geq 0, \eta_{dk} = \frac{exp(\theta_{dk})}{\sum_j exp(\theta_{dj})} \\
& \theta_d = t_{u_d} \circ q_{h_d}, \beta_k \in \mathcal{S}^{(N-1)}, \alpha > 0, \forall d, n \in I_d, \forall k
\end{aligned}
\tag{7}
$$

where $\mathcal{S}^{(N-1)}$ represents the (N-1)-simplex.

5.2 Aspect Rating Prediction and Term Detection

We utilize the dictionary β to detect the key terms for each aspect. For a particular aspect k, each row $\beta_{k \cdot}$ represents the association strength of each term for the aspect k. We can rank the terms based on their association strength and treat the top terms as the representative key terms for the aspect k. In contrast with the STC model, our proposed model incorporates the overall ratings associated with each review text as input, so our learned aspect dictionary can be more informative.

5.3 Optimization Technique—Block Coordinate Gradient Descent

We investigate the optimization technique for finding the solution for MAP estimation in (7). Note the such problem could be written in the following form,

$$
\begin{aligned}
\min \quad & f(\mathbf{Y}, \mathbf{S}, \mathbf{T}, \mathbf{Q}, \beta, \alpha) + \lambda \|\mathbf{T}\|_1 + \rho \|\mathbf{S}\|_1 \\
\text{s.t.} \quad & \mathbf{T} \geq \mathbf{0}, \mathbf{Q} \geq \mathbf{0}, \mathbf{S} \geq \mathbf{0}, \alpha > 0, \beta_k \in \mathcal{S}^{(N-1)}, \forall k
\end{aligned}
\tag{8}
$$

where $\|\mathbf{T}\|_1 = \sum_u \|t_u\|_1$, $\|\mathbf{S}\|_1 = \sum_d \sum_{n \in I_d} \|s_{dn}\|_1$ and $f(\mathbf{Y}, \mathbf{S}, \mathbf{T}, \mathbf{Q}, \beta, \alpha)$ denotes the function that unifies all the other terms in the objective of problem (7).

A popular approach for solving optimization problem of form (8) is the block coordinate descent (BCD) method. At each iteration of BCD, a single block (subset) of the whole set of variables is chosen to be optimized while fixing the remaining variables, as used in STC [31]. However, our model is more complex such that for each subproblem of BCD, we are unable to find a closed-form solution. In other words, in our model, solving each subproblem of BCD would be of high computational cost. To remedy this issue, we introduce the *block coordinate gradient descent* (BCGD) method.

Like BCD, BCGD is also an iterative algorithm starting with a specified initial point $x^0 = (\mathbf{Y}^0, \mathbf{S}^0, \mathbf{T}^0, \mathbf{Q}^0, \beta^0, \alpha^0)$. At each iteration of BCGD, it first chooses an block B to be updated (in (8), $B \in \{\mathbf{Y}, \mathbf{S}, \mathbf{T}, \mathbf{Q}, \beta, \alpha\}$). Then it calculates a descent direction at the current point $x = (\mathbf{Y}, \mathbf{S}, \mathbf{T}, \mathbf{Q}, \beta, \alpha)$ with respect to the block B, denoted by $\mathbf{d}(x; B)$. After the descent direction $\mathbf{d}(x; B)$ is obtained, we update the variable by $x^{new} = x + \alpha_B \mathbf{d}(x; B)$. Here α_B is the step size that could be determined by various searching rule, e.g. Armijo rule. Now the remaining question of BCGD is how to calculate the descent direction $\mathbf{d}(x; B)$. Mathematically, it is given as follows:

$$
\begin{aligned}
\mathbf{d}(x; B) := \quad & \arg\min \quad \nabla f(x)^T \mathbf{d} + \tfrac{1}{2} \|\mathbf{d}\|_2^2 + r(x + \mathbf{d}) \\
& \quad\ \text{s.t.} \quad\ \mathbf{d} + x \in \mathcal{F}, \mathbf{d}_j = 0, \quad \forall j \notin B.
\end{aligned}
\tag{9}
$$

where $r(x) = \lambda \|\mathbf{T}\|_1 + \rho \|\mathbf{S}\|_1$ and $\mathcal{F}$ denotes the whole feasible region in (8). Though it seems complicated, the following proposition ensures that for problem (8), the descent direction $\mathbf{d}(x; B)$ admits closed-form solutions for $B \in \{\mathbf{Y}, \mathbf{S}, \mathbf{T}, \mathbf{Q}, \alpha\}$.

Proposition 1. *Suppose v and x are given vectors in $\mathbb{R}^n$, then the optimal solution of the following optimization problem*

$$
\begin{aligned}
\min \quad & v^T \mathbf{d} + \tfrac{1}{2} \|\mathbf{d}\|_2^2 + \mu \|x + \mathbf{d}\|_1 \\
s.t. \quad & \mathbf{d}_j + x_j \geq 0, \quad j = 1, \ldots, n.
\end{aligned}
\tag{10}
$$

is given by

$$
\mathbf{d}_j^* = \max\{-x_j, -v_j - \mu\}, j = 1, \ldots, n. \tag{11}
$$

Let $\nabla_B f$ denote the partial derivative of function f with respect to block B. Then it is obvious that the descent directions $\mathbf{d}(x; \mathbf{T}), \mathbf{d}(x; \mathbf{S})$ are obtained by solving optimization problem of form (10) with $v = \nabla_{\mathbf{T}} f$, $\mu = \lambda$ and $v = \nabla_{\mathbf{S}} f$, $\mu = \rho$, respectively. Moreover, the optimization for descent directions $\mathbf{d}(x; \mathbf{Y})$, $\mathbf{d}(x; \mathbf{Q})$ and $\mathbf{d}(x; \alpha)$ are all of form (10) with $v = \nabla_Y f, v = \nabla_{\mathbf{Q}} f, v = \nabla_\alpha f$ respectively and $\mu = 0$. Thus by Proposition 1, the updating scheme of block $B \in \{\mathbf{Y}, \mathbf{S}, \mathbf{T}, \mathbf{Q}, \alpha\}$ are all with simple implementation and of low computational cost.

For the aspect dictionary block β, since its feasible region is a simplex, we could not hope for a closed-form of its update. Instead, we apply the projected gradient descent method for solving (9) and use a linear algorithm [5] to perform the projection to the simplex.

Thus, we summarize our Block Coordinate Gradient Descent for solving (7) in Algorithm 1. Our goal is to solve each latent variables including $\mathbf{Y}$, $\mathbf{S}$, $\mathbf{T}$, $\mathbf{Q}$, β, and α separately assuming that the other variables are fixed in an alternate manner.

Moreover, the convergence of BCGD has been extensively studied in the optimization community. Specifically, for optimization problems with the property that all non-smooth parts are of a block-separable structure, such as (8), both the objective value and the iterates generated by Algorithm 1 are guaranteed to converge to a critical point. We summarize the result in the following theorem and its proof could be found in [23].

Theorem 1. *Suppose $\{x^k\}$ is the sequence generated by Algorithm 1, and the step sizes are chosen by the Armijo rule bounded away from 0. Then the value of the objective function is nonincreasing and every cluster point of $\{x^k\}$ is a stationary point of Problem* (8).

6. EXPERIMENT

6.1 Data Sets

Algorithm 1 Algorithm for Our Proposed Model SACM

```
Input: A collection of reviews $\mathcal{D} = \{d_1, d_2, ..., d_{|D|}\}$. For each review $d \in \mathcal{D}$, the overall ratings $Y_d$, the corresponding user $u$, and the item $h$.
Output: $Y_d^A, s_{dn}, t_u, q_h, \beta, \alpha, \forall d, u, h$
Initialize $x^0 = ({Y_d^A}^0, s_{dn}^0, t_u^0, q_h^0, \beta^0, \alpha^0), n \in I_d, \forall d, u, h$ .
repeat
    for $d = 1$ to $|D|$ do
        Optimize over $Y_d^A$: solve the gradient $\nabla_{Y_d^A} f$ to obtain $\mathbf{d}(x; Y_d^A)$. Choose a step size $\alpha_{Y_d^A}$ to set ${Y_d^A}^{new} = Y_d^A + \alpha_{Y_d^A} \mathbf{d}(x; Y_d^A)$ and update $x$
        for $n = 1$ to $|I_d|$ do
            Optimize over $s_{dn}$: solve the gradient $\nabla_{s_{dn}} f$ to obtain $\mathbf{d}(x; s_{dn})$. Choose a step size $\alpha_{s_{dn}}$ to set $s_{dn}^{new} = s_{dn} + \alpha_{s_{dn}} \mathbf{d}(x; s_{dn})$ and update $x$
        end for
    end for
    for $u = 1$ to $|U|$ do
        Optimize over $t_u$: solve the gradient $\nabla_{t_u} f$ to obtain $\mathbf{d}(x; t_u)$. Choose a step size $\alpha_{t_u}$ to set $t_u^{new} = t_u + \alpha_{t_u} \mathbf{d}(x; t_u)$ and update $x$
    end for
    for $h = 1$ to $|H|$ do
        Optimize over $q_h$: solve the gradient $\nabla_{q_h} f$ to obtain $\mathbf{d}(x; q_h)$. Choose a step size $\alpha_{q_h}$ to set $q_h^{new} = q_h + \alpha_{q_h} \mathbf{d}(x; q_h)$ and update $x$
    end for
    Optimize over $\beta$: solve the gradient $\nabla_\beta f$ to obtain $\mathbf{d}(x; \beta)$. Choose a step size $\alpha_\beta$ to set $\beta^{new} = \beta + \alpha_\beta \mathbf{d}(x; \beta)$ after being projected to a simplex and update $x$
    Optimize over $\alpha$: set the gradient $\nabla_\alpha f$ to zero, we update $\alpha$ by $\alpha = \sqrt{\sum_d \sum_k (\frac{Y_{dk}^A - q_{h_d k}}{t_{u_d k}})^2 \frac{1}{|D||K|}}$ and then update $x$
until certain convergence criterion is met
```

We carry out some experiments on two review corpora. One is the beer review corpus from a beer-rating web site *BeerAdvocate*[2], which has been used in [12]. Another is the hotel review corpus crawled from *TripAdvisor*[3], and originally used in [26] and [25]. In the beer corpus, for each review, in addition to review texts, ratings are given on 4 aspects including Appearance, Aroma, Palate, and Taste. Furthermore, there is an overall rating for each review. All ratings range from 1 to 5 stars. In the hotel corpus, users are allowed to rate hotels on 7 predefined aspects in each review: Value, Room, Location, Cleanliness, Check In/Front Desk, Service, and Business Service, as well as an overall rating. All ratings range from 1 to 5 stars. In some reviews, there are several aspects not being rated and they are represented by "-1" instead of 1 to 5 stars. We call such kind of aspect rating as a **non-existent aspect rating**, and its corresponding aspect is **non-existent aspect**. Very often, a review text may contain only some and not necessarily all aspects. This issue is known as aspect sparsity as mentioned in Section 1. Some previous works such as [25] filter out such kind of reviews in their experiments since their models cannot handle aspect sparsity. In contrast, we retain such kind of reviews without removing them and form two data sets in our experiments. They are called "Beer" and "Hotel" data set. Table 1 depicts the statistics of these data sets. The Sparse Ratio is defined as the fraction of non-existent aspect ratings.

$$SparseRatio = \frac{\sum_d g_d}{D \times K} \tag{12}$$

where g_d denotes the number of non-existent aspect ratings in the review d. D and K are the number of reviews and the number of predefined aspects, respectively.

By controlling the weight of l_1 regularizer, our model can also be applicable for data sets without non-existent aspect ratings. Therefore, in order to further investigate the efficacy of our model, we also prepare two additional data sets deriving from the beer and hotel corpora without aspect sparsity by removing reviews containing non-existent aspect ratings similar to some previous works such as [26, 25]. These additional data sets are called "Beer-nonsparse" and "Hotel-nonsparse" as depicted in Table 1.

Data Set	#Item	#User	#Review	Sparse Ratio
Beer	6,469	14,993	302,399	0.273
Hotel	1,850	79,189	91,224	0.442
Beer-nonsparse	3,743	7,781	81,787	0.0
Hotel-nonsparse	1,850	52,882	58,513	0.0

Table 1: Statistics of data sets

6.2 Experimental Setup

We perform pre-processing on these data sets including: (1) removing the punctuations, stop words from a standard stop word list as in [8], and the terms whose count frequency is less than 5; (2) converting the words into lower cases; (3) stemming each word to its root form using Porter Stemmer [18].

We carry out the experiment on predicting the aspect ratings for each review to conduct quantitative evaluation. The numerical aspect ratings can be used as the ground-truth for the task of aspect rating prediction. Note that if a certain aspect rating exists but its corresponding aspect has not been mentioned in the review text content, then the rating cannot been regarded as a valid ground-truth information for the evaluation of aspect rating prediction. To ensure the validity of the ground-truth aspect ratings, we employ the Aspect Segmentation algorithm in [26] to segment each review. Aspect ratings which are not supported by the text content segments will be treated as non-existent aspect ratings. Besides, in order to align with the predefined aspects, we use a set of seed words for each aspect (e.g. "friend" and "concierg" for the Service aspect) in the beer and hotel domain as a prior to guide the text content component in our model, which has been conducted similarly in [25].

We initialize each word code s by the prior seed words, and uniformly initialize θ, t, q and β. The aspect ratings Y_d^A are initialized by its corresponding overall rating Y_d. For the parameter setting, we manually set $\alpha = 1.0$, $c = 0.1$, $\rho = 5e^{-4}$, and search for the most appropriate λ and γ both in the range of $[0.1, 1.0]$. The number of aspects for the Beer and Beer-nonsparse data sets is fixed as 4 while we fixed the number of aspects to 7 for the Hotel and Hotel-nonsparse data sets.

Our quantitative experiments are conducted in three trials. In the first trial, all the models will be evaluated on the "Beer" and "Hotel" data sets. The Beer data set is much larger than the Hotel data set. In the second trial, we evaluate all the models on the "Beer-nonsparse" and "Hotel-nonsparse" data sets. Note that our proposed model can

[2] http://beeradvocate.com/

[3] http://www.tripadvisor.com/

be easily configured to generate numerical aspect ratings without non-existent aspect rating by means of controlling the weight of l_1-regularizer. In the third trial, we examine the performance of non-existent aspect identification for our model. Finally, we also perform some qualitative experiment on user and item characterization.

6.3 Metrics

Similar to previous works such as [25], we make use of several metrics to measure the performance of our proposed model and all the comparing methods. Specifically, we use three groups of metrics to conduct quantitative evaluation. The first group of metrics evaluate the performance on all the reviews based on the aspect, including: (1) Mean Square Error (MSE) between the predicted aspect ratings and the ground-truth aspect ratings. It can evaluate the prediction accuracy. For the data set involving the aspect sparsity, we need to adjust the MSE metric as follows. Suppose that we successfully predict the non-existent aspect rating meaning that both the predicted aspect rating and the ground-truth aspect rating are "non-existent", MSE will be zero. On the other hand, if a model fails to detect "non-existent" aspect rating, the MSE will be penalized by our specified constant. In our experiment, this constant is 1.0. Note that if there is no non-existent aspect rating in the review data set, then this MSE is exactly the same as the standard MSE. (2) Pearson correlation of all the reviews (ρ_a). For an individual review, this metric can evaluate the performance on preserving the relative order of aspect ratings; (3) Percentage of failing to detect the best and worst aspect within reviews (Mis_a); (4) nDCG of aspect ranking in all the reviews ($nDCG_a$). For each review, we regard the ground-truth aspect ratings as the graded relevance to calculate the nDCG. Each of the first group of metrics is calculated by the average value over all the reviews. The second group of metrics evaluate the performance based on items, including: (1) Pearson correlation across all the items (ρ_h) measures the performance on maintaining the ranking order of aspect ratings for all items. Based on all the reviews commenting on each item, we calculate the average predicted aspect ratings and the ground-truth aspect ratings for each item to calculate ρ_h; (2) Mean Average Precision (MAP_h@10) measures the ranking accuracy for items. If each aspect is a query, after ranking by the ground-truth aspect ratings, we regard the top 10% of the items as the relevant answers. MAP_h@10 evaluates whether we are able to preserve their top ranking positions if using the predicted aspect rating to rank them. Each of the second group of metrics is calculated by the average value over all the items. For Beer and Hotel data set, when we calculate metrics except MSE, each non-existent predicted or ground-truth aspect ratings will be replaced by the mean determined by the existing aspect ratings in the same review.

The third group of metrics are used to evaluate the performance on the non-existent aspect identification, including: (1) Precision is the fraction of predicted non-existent aspects that are predicted correctly. (2) Recall is the fraction of non-existent aspects having being predicted correctly. (3) F1 score is the harmonic mean of precision and recall.

Note that for MSE and Mis_a, the lower the value is, the better the performance is. For the remaining metrics, the higher the value is, the better the performance is.

6.4 Aspect Rating Prediction

	Beer			Hotel		
	LRR	LARAM	SACM	LRR	LARAM	SACM
MSE	1.954	1.119	**0.916**	1.871	1.228	**0.878**
ρ_a	0.076	0.130	**0.231**	0.101	0.177	**0.260**
Mis_a	0.491	0.425	**0.377**	0.471	0.426	**0.413**
$nDCG_a$	0.902	0.924	**0.939**	0.811	0.848	**0.852**
ρ_h	0.497	0.607	**0.694**	0.590	0.613	**0.669**
MAP_h@10	0.407	0.472	**0.531**	0.381	0.392	**0.421**

Table 2: Aspect rating prediction performance on the data sets with aspect sparsity. For MSE and Mis_a, the lower the value is, the better the performance is. For other metrics, the higher the value is, the better the performance is.

	Beer			Hotel		
	STC	MedSTC	SACM	STC	MedSTC	SACM
Precision	0.184	0.248	**0.317**	0.489	0.429	**0.496**
Recall	0.223	0.471	**0.566**	0.315	0.669	**0.702**
F1 score	0.203	0.325	**0.406**	0.383	0.523	**0.581**

Table 3: Non-existent aspect identification performance on the data sets with aspect sparsity. For all the metrics, the higher the value is, the better the performance is.

We conduct the aspect rating prediction of our model SACM comparing with LRR [26] and LARAM [25], which are two state-of-the-art models to do latent aspect mining problem. Since LRR model needs to apply topic models to identify aspects, in order to conduct fair comparision, sLDA [4] model which is able to consider the overall rating is employed to identify aspects for each review in LRR.

The aspect rating prediction performance of different models on the data sets with aspect sparsity is illustrated in Table 2, where we highlight the best performance for each metric. In general, for both Beer and Hotel data sets, our proposed model SACM outperforms two comparing methods in all measures. In the first group of metrics, MSE denotes that SACM can achieve better prediction accuracy. ρ_a, Mis_a and $nDCG_a$ are the aspect-based ranking metrics. The results show that SACM is able to better preserve the relative order of the aspect ratings within a review. In other words, our model can better answer the questions such as "What is this user's favourite aspect?" and "Does this user prefer the Service than the Room of this hotel?". In addition, ρ_a is relatively low for all the methods because our predicted aspect ratings are real values while the ground truth aspect ratings are all integers, leading to an over-penalty for the ρ_a metric. Instead, $nDCG_a$ is able to alleviate this bias and handles the integer tie cases well. In the second group of metrics, ρ_h and MAP_h@10 indicate that the performance of LRR and LARAM on the ranking of items is inferior in comparision with that of SACM.

Table 4 depicts the result of the second trial experiment on "Beer-nonsparse" and "Hotel-nonsparse" data sets. It can be observed that our proposed model still outperforms the LRR and the LARAM in all measures.

6.5 Non-existent Aspect Identification

For the data sets "Beer" and "Hotel" with aspect sparsity, our model SACM is capable of identifying the non-existent aspect indicated by the user intrinsic aspect interest t. To evaluate the identification performance, we compare SACM with two different methods: Sparse Topical Coding (STC) and its supervised version MedSTC. These existing models only conduct non-existent aspect identification but cannot predict aspect ratings. STC can take a collection of reviews as input and identify the non-existent aspects of each review by the low association strength in the corresponding document code θ. MedSTC improves the STC model by taking advantage of the overall rating associated with each review, and identifies the non-existent aspects similar with STC. Note that we assume if the association strength or user intrinsic aspect interest value of a certain aspect is less than 0.005, then we regard the aspect as a non-existent aspect.

With the same parameters setting mentioned in Section 6.2, we show the non-existent aspect identification performance measured by precision, recall and F1 score in Table 3. We can observe that SACM shows superior performance on non-existent aspect identification than all the comparing methods, which is benefit from considering the user and item information into the modeling of aspect ratings and review texts. STC achieves a poor performance because of ignoring the valuable overall rating associated with each review. Besides, it can be observed that all the methods perform better on the Beer data set than that on the Hotel data set. It is mainly due to the reason that the fraction of non-existent aspect ratings (i.e. Sparse Ratio) of the Beer data set is less than that of Hotel data set implying that the users in *BeerAdvocate* are more willing to share detailed experience with others.

6.6 Qualitative Results

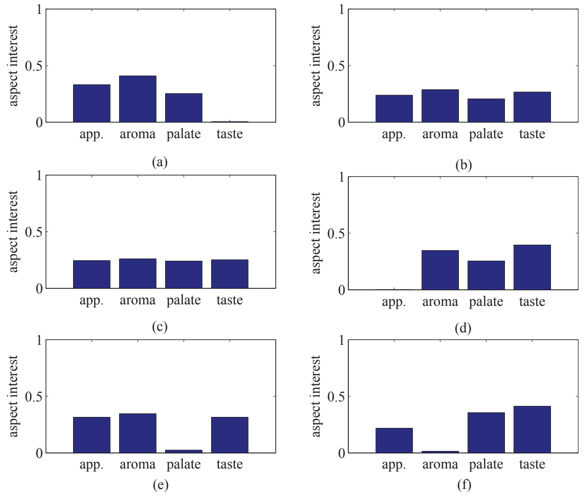
Table 5 shows the detected key term lists of some aspects, including Room, Cleanliness, Check In/Front Desk (i.e. CI/FD), Service, and Business service (i.e. BS), for the Hotel data set by SACM. It can be observed that each key term list can express the basic idea of its corresponding aspect. For example, “single”, “door”, “room”, “standard”, and “window” appear quite common in the reviews providing comments on the Room aspect. These terms are quite indicative to the Room aspect.

	Beer-nonsparse			Hotel-nonsparse		
	LRR	LARAM	SACM	LRR	LARAM	SACM
MSE	1.547	1.209	**1.112**	1.813	1.037	**0.936**
ρ_a	0.103	0.110	**0.206**	0.088	0.170	**0.179**
Mis_a	0.497	0.427	**0.398**	0.431	0.397	**0.371**
$nDCG_a$	0.881	0.917	**0.937**	0.869	0.892	**0.896**
ρ_h	0.573	0.645	**0.727**	0.543	0.746	**0.814**
MAP_h@10	0.331	0.340	**0.403**	0.479	0.488	**0.554**

Table 4: Aspect rating prediction performance on the data sets without aspect sparstiy. For MSE and Mis_a, the lower the value is, the better the performance is. For other metrics, the higher the value is, the better the performance is.

Room	Cleanliness	CI/FD	Service	BS
hotel	clean	staff	very	internet
very	pool	time	stay	include
excellent	resort	service	service	definite
city	recommend	only	breakfast	feel
enjoy	bathroom	restaurant	bed	coffe
single	floor	book	friend	tv
everything	price	check	help	easy
door	look	arrive	food	top
quite	little	before	offer	choice
room	comfortable	morning	dure	expensive
shower	free	rate	left	extrem
star	trip	wait	manage	reason
suite	desk	size	nothing	stop
family	lot	another	hot	home
husband	review	pay	since	provide
standard	front	money	tip	decor
window	street	reservation	english	modern
air	because	told	concierg	square
light	water	charge	run	plenty
smell	travel	person	care	spacious
noisy	close	available	impress	convenient
inn	alway	outside	probablity	central
toilet	week	reception	please	pleasant
waikiki	bad	extra	wall	distance
suggest	noisy	early	sit	direct

Table 5: The detected key terms of some aspects in the Hotel data set



(a)
(b)
(c)
(d)
(e)
(f)

Figure 4: Result of user characterization. “Appearance” is denoted by “app.”

7. USER AND ITEM CHARACTERIZATION

As we discussed before, the output user intrinsic aspect interest t_u and item intrinsic aspect quality q_h in the SACM can be used to characterize different types of users and items. Specifically, we apply the k-means clustering on the user intrinsic aspect interest t_u for all the users $u \in \mathcal{U}$ on the Beer data set and perform the same procedures for item intrinsic aspect quality $q_h, h \in \mathcal{H}$.

7.1 User Characterization

For the clustering of user intrinsic aspect interest, we specify the number of cluster as 6, and the average normalized user intrinsic aspect interest of each cluster is depicted in Figure 4. There are six types of users. Users in type (b) and type (c) represent the groups who have no obvious preference for different aspects of beers. When writing the reviews, these types of users always write detailed experience for each aspect. But other four types of users, namely, (a), (d), (e) and (f) have their own taste. For example, users in type (d) appear to be hardly interested in the Appearance of beer. When this type of user wants to buy a bottle of beer, he/she would not care about the appearance of the beers no matter how beautiful the appearance design is. On the other hand, from the perspective of beer merchants, they can provide personalized beer sales strategy for different types of users based on the result of user characterization.

7.2 Item Characterization

For the clustering of item intrinsic aspect quality, we specify the number of cluster as 5, and we show the average item intrinsic aspect quality of each cluster in Figure 5. It can be observed that the top item group possesses relatively better average aspect quality on each aspect than that of lower

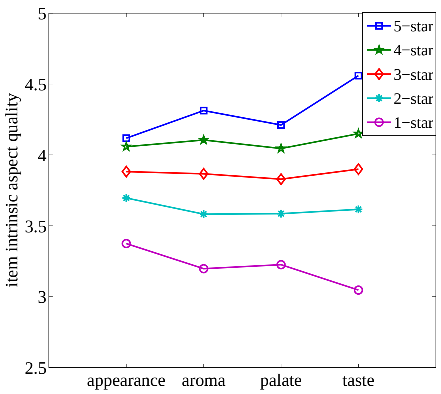


Figure 5: Result of item characterization

item group. We name them "5-star" item to "1-star" item. Users can make use of the result of item characterization to know the difference between different items at the aspect level, and choose the most appropriate item based on their own aspect interest. For example, based on Figure 5, when a user in type (a), who has no interest in the Taste aspect of beer, wants to buy a bottle of beer, he/she is able to see the main advantage of "5-star" beer due to its Taste, and the quality of other aspects has little difference with "4-star" beer. Hence, he would buy the "4-star" beer for saving money. On the other hand, merchants can know the reasons why the items they sold are inferior than other items. For example, based on Figure 5, a "4-star" beer seller can find that its main weakness is the Taste aspect in contrast with "5-star" beers.

8. CONCLUSION

We propose a generative model to tackle the latent aspect mining problem. Our proposed model SACM can handle the aspect sparsity when predict the aspect ratings from a review text corpus. SACM applies l_1-regularizer to control the sparsity on the aspect proportion and also takes user and item intrinsic information into consideration. Moreover, we conduct the analytical investigation for the Maximum A Posterior (MAP) problem used in our proposed model and develop a new block coordinate gradient descent algorithm to effectively find the solution with closed-form updating formulas. Our experimental results on two real-world review corpora demonstrate that our proposed model SACM outperforms the state-of-the-art models.

9. REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

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

[3] K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *WWW*, pages 519–528, 2003.

[4] D.Blei and J.McAuliffe. Supervised topic models. In *NIPS*, volume 7, pages 121–128, 2007.

[5] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the l 1-ball for learning in high dimensions. In *ICML*, pages 272–279, 2008.

[6] M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD*, pages 168–177, 2004.

[7] M. Hu and B. Liu. Mining opinion features in customer reviews. In *AAAI*, volume 4, pages 755–760, 2004.

[8] S. Lacoste-Julien, F. Sha, and M. I. Jordan. Disclda: Discriminative learning for dimensionality reduction and classification. In *NIPS*, pages 897–904, 2008.

[9] F. Li, N. Liu, H. Jin, K. Zhao, Q. Yang, and X. Zhu. Incorporating reviewer and product information for review rating prediction. In *IJCAI*, pages 1820–1825, 2011.

[10] C. Lin and Y. He. Joint sentiment/topic model for sentiment analysis. In *CIKM*, pages 375–384, 2009.

[11] Y. Lu and C. Zhai. Opinion integration through semi-supervised topic modeling. In *WWW*, pages 121–130, 2008.

[12] J. McAuley, J. Leskovec, and D. Jurafsky. Learning attitudes and attributes from multi-aspect reviews. In *ICDM*, pages 1020–1025, 2012.

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

[14] S. Moghaddam and M. Ester. The flda model for aspect-based opinion mining: addressing the cold start problem. In *Proceedings of the international conference on WWW*, pages 909–918, 2013.

[15] A. Mukherjee and B. Liu. Aspect extraction through semi-supervised modeling. In *ACL*, pages 339–348, 2012.

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

[17] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *EMNLP*, pages 79–86, 2002.

[18] M. F. Porter. An algorithm for suffix stripping. *Program: electronic library and information systems*, 14(3):130–137, 1980.

[19] M. Shashanka, B. Raj, and P. Smaragdis. Sparse overcomplete latent variable decomposition of counts data. In *NIPS*, pages 1313–1320, 2007.

[20] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society.*, pages 267–288, 1996.

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

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

[23] P. Tseng and S. Yun. A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117(1-2):387–423, 2009.

[24] P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *ACL*, pages 417–424, 2002.

[25] H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis without aspect keyword supervision. In *KDD*, pages 618–626, 2011.

[26] H. Wang and Y. Lu C. Zhai. Latent aspect rating analysis on review text data: a rating regression approach. In *KDD*, pages 783–792, 2010.

[27] S. Wang, F. Li, and M. Zhang. Supervised topic model with consideration of user and item. In *AAAI*, 2013.

[28] L. Xu, K. Liu, S. Lai, Y. Chen, and J. Zhao. Walk and learn: a two-stage approach for opinion words and opinion targets co-extraction. In *WWW*, pages 95–96, 2013.

[29] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, pages 1794–1801, 2009.

[30] J. Zhu, N. Lao, N. Chen, and E. P. Xing. Conditional topical coding: an efficient topic model conditioned on rich features. In *KDD*, pages 475–483, 2011.

[31] J. Zhu and E. P. Xing. Sparse topical coding. In *UAI*, pages 831–838, 2011.