World Wide Web (2014) 17:1321–1342
DOI 10.1007/s11280-013-0239-z

Users' interest grouping from online reviews based on topic frequency and order

Jianfeng Si · Qing Li · Tieyun Qian · Xiaotie Deng

Received: 8 July 2012 / Revised: 4 June 2013 /
Accepted: 2 July 2013 / Published online: 27 July 2013

Abstract Large volume of online review data can reveal consumers' major interests on domain product, which attracts great research interests from the academic community. Most of the existing works focus on the problems of review summarization, aspect identification or opinion mining from an item's point of view such as the quality or popularity of products. Considering the fact that users who generate those review texts draw different attentions to product aspects with respect to their own interests, in this article, we aim to learn K users' interest groups indicated by their review writings. Such K interest groups' identification can facilitate better understanding of major and potential consumers' concerns which are crucial for applications like product improvement on customer-oriented design or diverse marketing strategies. Instead of using a traditional text clustering approach, we treat the groupId/clusterId as a hidden variable and use a permutation-based structural topic model called KMM. Through this model, we infer K interest groups' distribution by discovering not only the frequency of product aspects (Topic Frequency), but also the occurrence

This paper is an extended version of our previous conference paper [32]

J. Si (✉) · Q. Li
Department of Computer Science, City University of Hong Kong, Hong Kong, China
e-mail: thankjeff@gmail.com

Q. Li
e-mail: itqli@cityu.edu.hk

T. Qian
State Key Laboratory of Software Engineering, Wuhan University, Wuhan, China
e-mail: qty@whu.edu.cn

X. Deng
AIMS Lab, Department of Computer Science, Shanghai Jiaotong University, Shanghai, China
e-mail: deng-xt@cs.sjtu.edu.cn

priority of respective aspects (Topic Order). They jointly present an informative summarization on the raw review corpus. Our experiment on several real-world review datasets demonstrates a competitive solution.

Keywords Review analysis · Structural topic modeling · Interest grouping

1 Introduction

With the development of Web2.0 techniques, people get involved into various online communities. Simultaneously, the user-generated data is of great value and need to be deeply mined. Especially in the E-commerce sites such as the www.amazon.com, large volume of online reviews are written by experienced customers or domain experts. Such reviews of products are commercially valuable because they are one kind of feedback and can well reflect the users' experiences or preferences on the products. But it is very hard to do manual summarization on such large datasets. There are a lot of existing works related to review analysis, yet most of these focus on the task of product recommendation to new users. In our research, we study and investigate the reviews' text writing styles in order to identify certain number of users' interest groups, each of which shares similar interests in domain products. For example, Table 1 present two interest groups $\{C_0, C_1\}$, it demonstrates the learning result of *KMM* and we will detail it in Section 3. Such interest groups can help manufacturers better understand consumers' main concerns for further customer-oriented product design, and can also serve as guidance to the user-targeting marketing strategy.

Considering the various kinds of consumers' needs, a user-targeting marketing strategy succeeds from its market understanding. Take the Digital Festival which is an annual campus promotion on notebooks in Hong Kong as an example, basically there would be five different categories for users to choose, namely: *Basic, Portable, Performance, High-Performance, Multi-Media*. As shown in Figure 1, it reveals five distinct consumer concerns. For instance, those people targeting at *High-Performance* would typically comment using words like "CPU, speed, performance, RAM, etc.", while those caring for *Portable* would comment using words like "size, weight, battery, etc.". The former users could be of "software engineer" while the latter would more likely to be of "business man". As this example shows, aspect interests reflected by text writings actually can identify groups with separative taste of interests.

Compared to regular documents, the online reviews are written in a much more free style. Figure 2 contains three hotel reviews: the left two reviews talk about the staff service mostly while the right one pays more attention to the hotel location

Table 1 Example of two interest groups with specific topic frequency and order

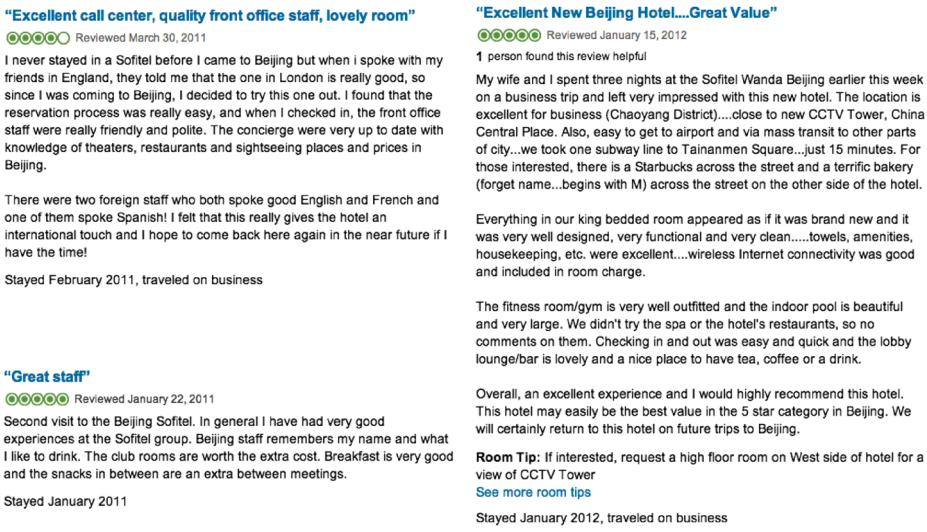
C_0		C_1	
Ordering	Frequency (%)	Ordering	Frequency (%)
Room Condition	60	Food	15
Staff Service	35	Staff Service	50
Food	5	Room Condition	35

Figure 1 Notebooks on sell from Digital Festival 2011 of City University of Hong Kong. http://www.hknotebook.com/

and room condition. Obviously, they stress differently with their own concerns. So it is hard to derive a unique global topic structure among review corpus. Instead, we identify K topic structures each of which shares similar aspect interests. The choice of K depends on what granularity the groups need to be; e.g., we may set $K = 5$ in the notebook example.

Chen et al. [9] proposed a structural topic model with the assumption of "one centroid ordering constraint" for learning document structure. We make use of their work, and further extend their latent topic model to solve the interest grouping problem by introducing a new hidden variable k to indicate the group to which documents belong. We then learn the topic structure w.r.t. K topic frequencies and K topic centroid orderings, and finally output the K interest groups.

In this article, we extend a permutation-based structural topic model for the task of unsupervised learning of interest groups, each with a specific Topic Frequency



"Excellent call center, quality front office staff, lovely room"

Reviewed March 30, 2011

I never stayed in a Sofitel before I came to Beijing but when i spoke with my friends in England, they told me that the one in London is really good, so since I was coming to Beijing, I decided to try this one out. I found that the reservation process was really easy, and when I checked in, the front office staff were really friendly and polite. The concierge were very up to date with knowledge of theaters, restaurants and sightseeing places and prices in Beijing.

There were two foreign staff who both spoke good English and French and one of them spoke Spanish! I felt that this really gives the hotel an international touch and I hope to come back here again in the near future if I have the time!

Stayed February 2011, traveled on business

"Great staff"

Reviewed January 22, 2011

Second visit to the Beijing Sofitel. In general I have had very good experiences at the Sofitel group. Beijing staff remembers my name and what I like to drink. The club rooms are worth the extra cost. Breakfast is very good and the snacks in between are an extra between meetings.

Stayed January 2011

"Excellent New Beijing Hotel....Great Value"

Reviewed January 15, 2012

1 person found this review helpful

My wife and I spent three nights at the Sofitel Wanda Beijing earlier this week on a business trip and left very impressed with this new hotel. The location is excellent for business (Chaoyang District)....close to new CCTV Tower, China Central Place. Also, easy to get to airport and via mass transit to other parts of city...we took one subway line to Tainanmen Square...just 15 minutes. For those interested, there is a Starbucks across the street and a terrific bakery (forget name...begins with M) across the street on the other side of the hotel.

Everything in our king bedded room appeared as if it was brand new and it was very well designed, very functional and very clean.....towels, amenities, housekeeping, etc. were excellent....wireless Internet connectivity was good and included in room charge.

The fitness room/gym is very well outfitted and the indoor pool is beautiful and very large. We didn't try the spa or the hotel's restaurants, so no comments on them. Checking in and out was easy and quick and the lobby lounge/bar is lovely and a nice place to have tea, coffee or a drink.

Overall, an excellent experience and I would highly recommend this hotel. This hotel may easily be the best value in the 5 star category in Beijing. We will certainly return to this hotel on future trips to Beijing.

Room Tip: If interested, request a high floor room on West side of hotel for a view of CCTV Tower

See more room tips

Stayed January 2012, traveled on business

Figure 2 Reviews of *Sofitel Wanda Beijing Hotel* from www.tripadvisor.com

and Topic Order. Two main observations are utilized to discriminate the interest groups.

(1) the topic frequency, as different people may focus on different aspects to different extent as reflected on how much words they are talking about certain topic/aspect.
(2) the topic order, as different people may have different concern priorities as reflected on which topic/aspect comes into their mind firstly.

With the above considerations, we focus on the task of identifying K interest groups, with each of the groups taking a similar taste on aspect interests. Note that we term these as "users' interest groups" instead of "text clusters" to emphasize the actual human interests and concerns under the text, and our learning result is far beyond just clustering documents into groups.

The rest of the article is organized as follows. In Section 2 we briefly review some related works. In Section 3, we define our problem formulation and propose our model: *KMM*. In Section 4, we address the parameter inference problem. The experiment on several real world review datasets is presented in Section 5. Finally we conclude our work and outlook further work in Section 6.

2 Related works

Online review analysis has attracted much attention recently, including opinion summarization [22, 27, 31, 33], sentiment analysis [1, 4, 11, 20], and opinion spam detection [16, 17, 24, 25]. Topic modeling has been explored for the task of aspect identification [23, 33] where aspects are treated as topics. The interested posterior distributions are estimated using approximate inference techniques such as Gibbs sampling [6] or variational inference [18].

Compared to the bag-of-words and bag-of-topics assumption [8], newly developing topic models are more likely to integrate topic models with structural models [7, 21] with the consideration of inner-connected relationships between topics. Many works break the bag-of-topics assumption and introduce extra-sentential constraints on topic assignment with structural considerations [14, 28, 33, 34]. In particular, the relationships between topics assigned to adjacent textual units (sentences, paragraphs or sections instead of words) bias the topic distributions of adjacent textual units to be similar [28, 34], forming a Markovian transition process. So the topic assignments are locally infected. For example, the Hidden Topic Markov Model (*HTMM*) [14] defines a generative process for the documents' topics, in which sentence i gets the same topic assignment as $i-1$ with a relatively high probability.

Structural topic models drive latent variables control the generation of words under certain structural topics constraints. Instead of modeling topic distribution over documents, Author-Topic (AT) models [29] makes use of authorship information, each author is associated with a multinomial distribution over topics and each topic is associated with a multinomial distribution over words. Author-Persona-Topic (APT) [19] and Author-Interest-Topic (AIT) [19] both extend the AT model and learn the author's personal interests. After obtaining the personal interests, interest group can be obtained via Frequency Pattern Mining (*FPM*) [2]. While APT and AIT focus on author's research interests reflected from their academic publications which have

very clear author information. In contrast, in the online review domain, the author information would not be clearly presented in most of the cases. Even if some review datasets have such information, it would be quite sparse due to the fact that each author only posts very few reviews. Learning the author-specific interests in review analysis application would not be as practical as in academic literature applications. Thus we don't consider any author information of the reviews in our model. Instead, we learn interest groups based on certain common tastes of their product usage experience.

As the Markovian process only makes local decisions regarding the topic transitions, Chen et al. [9] proposed a structural topic model which learned a global document structure under the assumption that there existed a global topic structure in a domain-based document collection. For example, when an article introduces a city, it mostly introduces its history first, followed by geography, politics, economy, etc., that is, the order as "history, geography, politics, economy, etc." defines the centroid topic ordering when we introduce a city. Each document follows the centroid ordering with some possible dispersion to get its own topic ordering (for example, to introduce the economy before politics sometimes). Their work makes use of the *Generalized Mallows Model* (*GMM*) [10] over permutations to express the centroid topic ordering, but it can only find one global structure which is not adapted to the discourse-level text corpus such as online reviews, thus we extend their model with an aim to solve the interest grouping problem.

3 Structural topic model

We define in this section the problem of research and propose a new structural topic model called *K Mallows Model* (*KMM*).

3.1 Problem formulation

Given a product review corpus $\{d_1, \cdots, d_D\}$ in a certain domain, each document contains N_d sentences $\{s_{d1}, \cdots, s_{d_{N_d}}\}$. Regarding a product aspect as a topic, each sentence gets a topic assignment $z_{d,s} \in \{1, \cdots, T\}$. Our research focus is to identify certain number of users' interest groups revealed from the review corpus. We assume there exist K such interest groups $\{C_1, \cdots, C_K\}$ with separative common product interests as indicated by their review writings.

Take the hotel reviews as an example, we may discover one interest group who cares mostly on the *Room Condition*, followed by *Staff Service*, and lastly on the *Food* a little bit. There may be another group who cares about the *Food* mostly and firstly, followed by the *Staff Service* and lastly on the *Room Condition*. Obviously, those two groups draw different topic concern frequencies and concern priorities. One possible output of our *KMM* on this example is illustrated in Table 1, which contains two observations for each of the groups C_0 and C_1 namely:

(1) the topic frequency: C_0 draws most attention on *Room Condition* (60 %), followed by *Staff Service* (35 %), and lastly *Food* (5 %), while for C_1 it is 50 % for *Staff Service*, 35 % for *Room Condition*, and 15 % for *Food*.
(2) the centroid topic ordering, which is "*Room Condition* → *Staff Service* → *Food*" for C_0, while that of C_1 is "*Food* → *Staff Service* → *Room Condition*";

The combination of the above two observations can help us discriminate one group from another. As this example shown, our task is to find out K ($K = 2$ in this example) clusters on the review text corpus (each document gets a clusterId assignment) accompanied by a meaningful explanation (topic frequencies and topic orders). *KMM* jointly learns:

(1) K different aspect frequencies (e.g., the *Frequency* columns in Table 1);
(2) K centroid aspect orders (e.g., the *Ordering* columns in Table 1);
(3) the interest groups based on (1) and (2) (e.g., C_0 and C_1 in Table 1);
(4) the shared language model of each aspect described by the word distribution (e.g., Table 3).

3.2 *Generalized Mallows Model* (*GMM*) over permutations

Generalized Mallows Model (*GMM*) is a distribution over permutations [10]. *GMM* exhibits two important properties. Firstly, *GMM* concentrates probability mass on one centroid ordering, which represents the in-domain documents' structural similarity; orderings which are close to the centroid will get high probability mass while those whose many elements have been moved will get less probability mass. Secondly, its parameter set scales linearly with the number of elements being ordered, making it sufficiently constrained and tractable for inference [9].

In *GMM*, the order of a permutation is represented as an inversion count vector $(v_1, \cdots, v_T)$, where v_T should always be 0.

Take one simplest example for explanation, we assume that there is a centroid ordering as $< 0, 1, 3, 2 >$. When a new ordering $< 1, 0, 2, 3 >$ comes, the following steps show the calculation process of its inversion count vector:

1. Get the ordering relationships
   According to the centroid: $< 0, 1, 3, 2 >$, the ordering relationships are defined by the following 6 inequalities:

   $o(0) < o(1), o(0) < o(3), o(0) < o(2)$;
   $o(1) < o(3), o(1) < o(2)$;
   $o(3) < o(2)$;

2. For every item of the centroid, test its inversion count according to the new ordering: $< 1, 0, 2, 3 >$ and the above ordering relationships:
   - for 0, according to the ordering relationships we know: $o(0) < o(1)$, but 1 occurs before 0 in the new ordering which introduces an inversion, so, its corresponding inversion count is 1.
   - for 1, it is in the first position of the new ordering, i.e., no one exceeds it, so its inversion count is 0.
   - for 3, it is exceeded by element 2, as $o(3) < o(2)$, so its inversion count is 1.
   - for 2, it is the "biggest" element in the centroid, so no element should exceed it, resulting its inversion count to be 0 always, so we could ignore it.

3. Thus, the inversion count vector for the new ordering $< 1, 0, 2, 3 >$ w.r.t. the centroid $< 0, 1, 3, 2 >$ should be (1,0,1).

The sum of all the components of the inversion count vector is the Kendall τ distance between the new ordering and the centroid ordering, which reflects the minimum number of adjacent elements' swaps needed to transform the new ordering into the centroid. Given the centroid order, the inversion count vector can uniquely determine the new order and vice versa. In our work, we get the topic order by firstly obtaining its inversion count vector, then we convert it to its corresponding order, and also we do sampling on the inversion count vector instead of topic order.

The probability mass function of *GMM* is defined as follows:

$$GMM(\boldsymbol{v};\boldsymbol{\rho}) = \frac{e^{-\sum_j \rho_j v_j}}{\psi(\boldsymbol{\rho})} = \prod_{j=1}^{T-1} \frac{e^{-\rho_j v_j}}{\psi_j(\rho_j)} \tag{1}$$

where $\psi(\boldsymbol{\rho}) = \prod_j \psi_j(\rho_j)$ is the normalization factor, with:

$$\psi_j(\rho_j) = \frac{1 - e^{-(T-j+1)\rho_j}}{1 - e^{-\rho_j}} \tag{2}$$

For parameter $\rho_j (> 0)$, the *GMM* assigns the highest probability mass to each $v_j = 0$, and the probability mass drops exponentially as the inversion counts become bigger. When all $\{v_j\} = 0$, it becomes the centroid itself.

3.3 Model overview

There are two constraints considered in the *GMM* topic model of [9]: the first posits that each document exhibits coherent, nonrecurring topics; the second states that documents from the same domain tend to present a similar topic structure. We extend their model by removing the second constraint and assume that K similar topic structures exist in the review corpus, so as to help identify K different interest groups. We term the extended model as *K Mallows Model* (*KMM*) topic model.[1]

Two points are considered when making the extension:

1. *Lack of a global uniform topic structure* Online review writings contributed by various writers/reviewers are of totally free styles, as these writers may be freshmen, experienced customers, domain experts or sometimes even spam makers. Furthermore, people always focus on different aspects w.r.t. their own interests, so no common guideline on "how to write a review" is given. As a result, it is hard to derive a global uniform topic structure.
2. *Power of the discriminative groups* Product manufacturers succeed from adopting a customer-oriented strategy. As the consuming market grows, all-in-one product design no longer applies. Customers should be discriminatively treated, so it is critically important to identify the discriminative groups of various kinds of customers and design multiple strategies. Identifying K different interest groups help understand the major and potential customers' product concerns.

[1] In our *KMM* model, we set each ρ_j to be a scalar number: ρ for simplicity, which reduces the *Generalized Mallows Model* to be the standard *Mallows Model*.

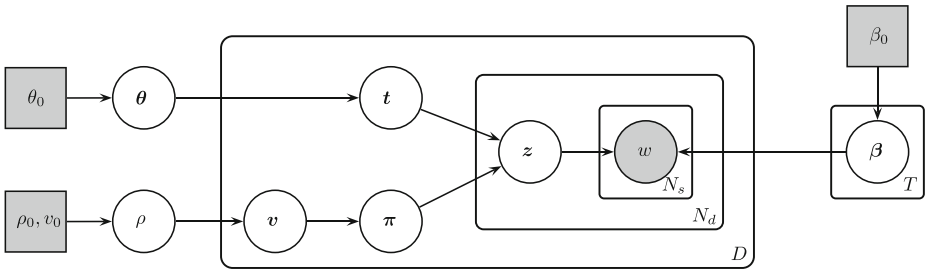


Figure 3 *GMM* generative Bayesian graphical model [9]

Similar to the *GMM* topic model, our *KMM* model firstly finds out how frequently each topic is expressed in the document and how the topics are ordered. The ordered topic sequence then determines the selection of words for each sentence (we treat "sentence" as the basic text unit of topic assignment). The graphical model of the original *GMM* and our *KMM* are shown in Figures 3 and 4, respectively.

As seen from Figure 4, there are K interest groups corresponding to K topic frequency distributions: $\{\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_K\}$ and K centroid topic orders/permutations: $\{\boldsymbol{\pi}'_1, \cdots, \boldsymbol{\pi}'_K\}$. Each $\boldsymbol{\theta}_i$ is a T dimensional vector, representing the parameters of a multinomial distribution over topics $\{1, \cdots, T\}$. Each $\boldsymbol{\pi}'_i$ defines a permutation on the topics $\{1, \cdots, T\}$. The combination of these two reflects K interest groups with respect to similar product interests. So, during the generative process of each document, we firstly select the groupId/clusterId: k_d, then draw the topic frequency and topic order w.r.t. the cluster it belongs to.

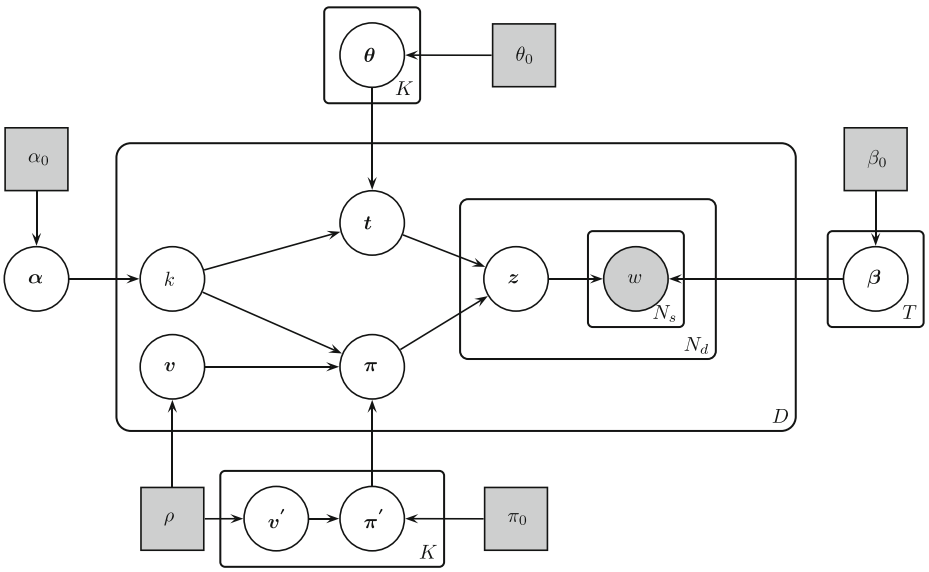


Figure 4 *KMM* generative Bayesian graphical model

Here is the specification of all the parameters and variables in Figure 4:

1. User setting parameters
   - T - number of topics
   - K - number of clusters/groups
2. Document characteristics
   - V - vocabulary size
   - D - number of documents in the corpus $\{1, \cdots, D\}$
   - N_d - number of sentences in document $d (\in \{1, \cdots, D\})$
   - N_s - number of words in sentence $s (\in \{1, \cdots, N_d\})$
3. Symmetric Dirichlet priors
   - α_0 - prior of the cluster size distribution
   - θ_0 - prior of the topic frequency distribution
   - β_0 - prior of the language model
4. Dirichlet distributions
   - $\boldsymbol{\alpha}$ - parameters of the distribution of clusters' member size: $\boldsymbol{\alpha} \sim Dirichlet(\alpha_0)$
   - $\boldsymbol{\theta}$ - parameters of the multinomial distribution over topics $\{1, \cdots, T\}$: $\boldsymbol{\theta} \sim Dirichlet(\theta_0)$
   - $\boldsymbol{\beta}$ - parameters of the language models which are V dimensional parameters of multinomial distributions: $\boldsymbol{\beta} \sim Dirichlet(\beta_0)$
5. Standard *Mallows Model*
   - ρ - dispersion parameter of the standard *Mallows Model*
   - $\boldsymbol{\pi_0}$ - natural ordering: $\boldsymbol{\pi_0} = \{1, \cdots, T\}$
   - $\boldsymbol{v}'$ - inversion count vector of each cluster's centroid ordering w.r.t. π_0
   - $\boldsymbol{\pi}'$ - centroid ordering of each cluster
   - $\boldsymbol{v}$ - inversion count vector of each document w.r.t. the $\boldsymbol{\pi}'$ of the belonging cluster
   - $\boldsymbol{\pi}$ - topic ordering of each document
6. Other hidden variables
   - $\boldsymbol{k}$ - groupId/clusterId of each document
   - $\boldsymbol{t}$ - topic occurrence vector of each document
   - $\boldsymbol{z}$ - topic assignment of a sentence
7. Observed variable
   - $\boldsymbol{w}$ - words in sentences of a document

As mentioned earlier, for each document d with N_d sentences, we firstly draw a clusterId: k_d, then obtain a bag of topics $\boldsymbol{t}_d$ by sampling N_d times from *Multinomial*(θ_{k_d}) (specifying the topic frequency) and the topic ordering $\boldsymbol{\pi}_d$ (controlling the topic occurrence order).

Here, the bag of topics $\boldsymbol{t}_d$ is drawn in the traditional *LDA* [8] way under the multinomial distribution with parameter vector $\boldsymbol{\theta}_{k_d}$, the latter is shared among all

the members of cluster: k_d. This indicates the similar interests in terms of the topic frequency in that group as a multinomial distribution, with K different $\boldsymbol{\theta}$ parameters representing the different frequency distributions separately.

Similarly, the topic ordering variable $\boldsymbol{\pi}_d$ is a permutation over topics 1 to T, indicating the topic occurrence order in the document. This is drawn from the standard *Mallows Model* under its centroid ordering $\boldsymbol{\pi}'_{k_d}$, the latter is the centroid for all members of cluster: k_d. This indicates the similar interests in terms of the topic ordering in that group, with K different $\boldsymbol{\pi}'$ parameters representing the different orderings separately.

Combining $\{\boldsymbol{\theta}_k\}$ and $\{\boldsymbol{\pi}'_k\}$, there are K different groups/clusters taking their individual aspect interests. Unlike these cluster related parameters, the T language models $\{\boldsymbol{\beta}_1, \cdots, \boldsymbol{\beta}_T\}$ are shared among the whole document corpus.

3.4 Generative process

The generative process defines how the documents are produced by introducing hidden variables. In Section 4, we shall present details of the learning process of these variables and parameters. The specific steps of our generative process are given below:

1. For each topic $t \in \{1, \cdots, T\}$, draw a language model $\boldsymbol{\beta}_t \sim Dirichlet(\beta_0)$, which specifies the word distribution over topic t.
2. For each cluster $k \in \{1, \cdots, K\}$, draw its parameters separately to reveal the different topic structures, as follows:
   - draw a topic distribution $\boldsymbol{\theta}_k \sim Dirichlet(\theta_0)$, which expresses how likely each topic would occur in documents of each cluster k;
   - draw a centroid ordering $\boldsymbol{\pi}'_k$ by firstly drawing its corresponding inversion count vector: $\boldsymbol{v}'_k$ according to (1), and then convert it to the ordering: $\boldsymbol{\pi}'_k$ which expresses the topic occurrence priority of each cluster.
3. Draw a cluster distribution $\boldsymbol{\alpha} \sim Dirichlet(\alpha_0)$, $\boldsymbol{\alpha}$ is a K dimensional vector used as the parameters of the multinomial distribution, indicating how likely each cluster is assigned to each document.
4. For each document d with N_d sentences:
   - draw a sample $k_d \sim Multinomial(\boldsymbol{\alpha})$ which indicates the clusterId of d;
   - draw a bag of topics $\boldsymbol{t}_d$ by sampling N_d times from $Multinomial(\boldsymbol{\theta}_{k_d})$;
   - draw a topic ordering $\boldsymbol{\pi}_d$ by sampling an inversion count vector $\boldsymbol{v}_d \sim GMM(\rho, \boldsymbol{\pi}_{k_d})$;
   - compute the topic assignment vector $\boldsymbol{z}_d$ for document d's N_d sentences by sorting $\boldsymbol{t}_d$ according to $\boldsymbol{\pi}_d$;
   - for each sentence s in document d, sample each word w in s according to the language model of topic $t = z_{d,s}$: $w \sim Multinomial(\beta_{z_{d,s}})$.

3.5 Analysis of user setting parameters

Parameter T represents the number of topics in the data corpus, so the choice of T decides the granularity of the presented topics. According to the common sense, different products take different number of aspects but mainly fall into the range

of [2,10]. Also, because our interest groups are modeled under the random space of aspect frequencies and orders, a too small setting of T will restrict the ability to find enough discriminative groups because of the random space limitation. On the other hand, a too big setting of T will make our algorithm drop greatly in time efficiency during the learning process because of the random space explosion. In experiment, we try different T settings and choose it by heuristic experience.

Parameter K decides the number of distinct groups. If we set $K = 1$, we just get one global topic structure as [9] which reflects the human users' common consideration on product aspects, so no discriminative interest groups are available. On the other hand, if we only consider the different aspect orderings, there would be a maximum of $T!$ different groups. But in fact, we are only interested in some main discriminative groups. For example, from the manufacturers' point of view, K should be decided according to their improvement ability. Indeed, manufacturers are often not possible to identify and accommodate every specific customer's need within a certain period of time.

4 Inference

We use Gibbs sampling [6] which is a stochastic inference method to infer the parameters. It is a kind of Markov Chain Monte Carlo method which can construct a Markov chain over the hidden variable space, in which its stationary distribution converges to the target joint distribution.

In our Gibbs sampling process, there are in total four hidden variables to be re-sampled: $\{\boldsymbol{k}, \boldsymbol{t}, \boldsymbol{\pi}, \boldsymbol{\pi}'\}$, $\boldsymbol{k}$ is the clusterId of a document, $\boldsymbol{t}$ indicates the topic occurrence in a document, $\boldsymbol{\pi}$ determines the topic ordering in a document, and $\boldsymbol{\pi}'$ determines the centroid topic ordering in each cluster. The topic assignments $\{\boldsymbol{z}\}$ of all sentences are not directly sampled; instead, $\boldsymbol{z}$ is a combination of $\boldsymbol{t}$ and $\boldsymbol{\pi}$, and also $\boldsymbol{\pi}$ and $\boldsymbol{\pi}'$ are sampled indirectly by transformation from their corresponding inversion count vectors.

The sampling process of $\boldsymbol{t}$ and $\boldsymbol{\pi}$ is almost the same as that in [9], except for the following:

1. the topic occurrence statistics are computed at the level of each cluster's sub-collection instead of the whole corpus;
2. instead of a global one, each document gets its own topic centroid ordering indicated by its clusterId.

All sampling equations are obtained by the following four steps:

1. Resample the topicId for each sentence of each document:

$$p(\boldsymbol{t}_{d,i} = t|\cdots) \propto p(\boldsymbol{t}_{d,i} = t|\boldsymbol{t}_{-(d,i)}, k_d, \theta_0) * p(\boldsymbol{w}_d|\boldsymbol{t}_d, \boldsymbol{\pi}_d, \boldsymbol{w}_{-d}, \boldsymbol{z}_{-d}, \beta_0)$$
$$= \frac{N_{k_d}(\boldsymbol{t}_{-(d,i)}, t) + \theta_0}{N_{k_d}(\boldsymbol{t}_{-(d,i)}) + T\theta_0} * p(\boldsymbol{w}_d|\boldsymbol{z}, \boldsymbol{w}_{-d}, \beta_0) \quad (3)$$

Here, $N_{k_d}(\boldsymbol{t}_{-(d,i)}, t)$ is the total number of sentences assigned to topic t in cluster k_d without counting $t_{d,i}$. And $N_{k_d}(\boldsymbol{t}_{-(d,i)})$ is the total number of sentences in cluster k_d except for $s_{d,i}$.

2. Resample the topic ordering $\boldsymbol{\pi}_d$ for each document by resampling every component of corresponding inversion count vector $\boldsymbol{v}_d$:

$$\begin{aligned} p(\boldsymbol{v}_{d,j} = v|\cdots) &\propto p(\boldsymbol{v}_{d,j} = v|\rho) * p(\boldsymbol{w}_d|\boldsymbol{t}_d, \boldsymbol{\pi}_d, \boldsymbol{w}_{-d}, \boldsymbol{z}_{-d}, \beta_0) \\ &= GMM(v; \rho) * p(\boldsymbol{w}_d|\boldsymbol{z}, \boldsymbol{w}_{-d}, \beta_0) \end{aligned} \tag{4}$$

3. Resample the clusterId: k_d for each document:

$$\begin{aligned} p(\boldsymbol{k}_d = k|\cdots) &\propto p(\boldsymbol{k}_d = k|\boldsymbol{k}_{-d}, \alpha_0) * p(\boldsymbol{v}_d|\boldsymbol{\pi}_d, \boldsymbol{\pi}^{'}_k, \rho) * p(\boldsymbol{t}_d|k, \theta_0) \\ &= \frac{N(\boldsymbol{k}_{-d}, k) + \alpha_0}{N(\boldsymbol{k}_{-d}) + K\alpha_0} * GMM(\boldsymbol{v}_d; \boldsymbol{\pi}_d, \boldsymbol{\pi}^{'}_k, \rho) * \prod_{s \in d} \frac{N_k(\boldsymbol{t}, t_s) + \theta_0}{N_k(\boldsymbol{t}) + T\theta_0} \end{aligned} \tag{5}$$

Where, $N(\boldsymbol{k}_{-d}, k)$ is the total number of documents assigned to clusterId: k without counting document d. $N(\boldsymbol{k}_{-d})$ is the total number of documents except for d. $N_k(\boldsymbol{t}, t_s)$ is the number of sentences assigned to t_s in cluster: k, and $N_k(\boldsymbol{t})$ is the total number of sentences in cluster: k. The three factors in (5) represent the prior consideration of choosing cluster, the dispersion between document's topic order and cluster's centroid, and the dispersion between document's topic frequency and cluster's topic frequency, respectively.

4. Resample the centroid topic ordering $\boldsymbol{\pi}^{'}$ of each cluster by resampling corresponding inversion count vector $\boldsymbol{v}^{'}$:

$$p(\boldsymbol{v}^{'}_{k,j} = v|\cdots) \propto \prod_{d \in C_k} p(\boldsymbol{v}_d|\boldsymbol{\pi}_d, \boldsymbol{\pi}^{'}_k, \rho) = \prod_{d \in C_k} GMM(\boldsymbol{v}_d; \boldsymbol{\pi}_d, \boldsymbol{\pi}^{'}_k, \rho) \tag{6}$$

It can be interpreted that, given the cluster's member assignments, cluster's centroid ordering is updated so that it can well represent the topic order of all its members.

In (3) and (4), the document probability is computed in the same way as [9] except that statistics are taken for each cluster separately, i.e.:

$$\begin{aligned} p(\boldsymbol{w}_d|\boldsymbol{z}, \boldsymbol{w}_{-d}, \beta_0) &= \prod_{t=1}^{T} \int_{\beta_t} p(\boldsymbol{w}_d|\boldsymbol{z}_d, \boldsymbol{\beta}_t) p(\boldsymbol{\beta}_t|\boldsymbol{z}, \boldsymbol{w}_{-d}, \beta_0) d\boldsymbol{\beta}_t \\ &= \prod_{t=1}^{T} DCM(\{w_{d,i} : z_{d,i} = t\}|\{\boldsymbol{w}_{-(d,i)} : \boldsymbol{z}_{-(d,i)} = t\}, \beta_0) \end{aligned} \tag{7}$$

Here, $DCM(.)$ is the result of integrating over multinomial parameters with a Dirichlet prior (*Dirichlet Compound Multinomial Distribution* [5]).

For a Dirichlet prior with parameters $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_W)$, the DCM assigns the following probability to a series of observations $\boldsymbol{x} = \{x_1, \cdots, x_n\}$:

$$DCM(\boldsymbol{x}; \boldsymbol{\alpha}) = \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \prod_{i=1}^{W} \frac{\Gamma(N(\boldsymbol{x}, i) + \alpha_i)}{|\boldsymbol{x}| + \sum_j \alpha_j}, \tag{8}$$

where $N(\boldsymbol{x}, i)$ refers to the number of times word i appears in $\boldsymbol{x}$, and $\Gamma(.)$ is the Gamma function (see [9]).

The DCM's posterior probability density function conditioned on a series of observations $\boldsymbol{y} = \{y_1, \cdots, y_n\}$ can be computed by updating each α_i with counts on how often word i appears in y:

$$DCM(\boldsymbol{x}|\,\boldsymbol{y}, \alpha) = DCM(\boldsymbol{x}; \alpha_1 + N(\boldsymbol{y}, 1), \cdots, \alpha_W + N(\boldsymbol{y}, W)). \tag{9}$$

The overall sampling algorithm is given as Algorithm 1.

During the sampling process, we try every possible topic assignment (T in total) to every sentence (N_d in total) in a document (D in total). So the time complexity for one iteration would be $O(D * N_d * T)$.

Algorithm 1 Resampling Algorithm by Gibbs Sampling

```
init centroid topic ordering for clusters: {π'_1, ··· , π'_K}
init clusterId for all documents: {k_1, ··· , k_D}, k_d ∈ {1, ··· , K}
init topic counts for all documents: {t_1, ··· , t_D}
init topic ordering for all documents: {π_1, ··· , π_D} by initializing its v_d
  combining with π'_{k_d}
init topic assignments for all documents: {z_1, ··· , z_D} by combining t_d and π_d
for it = 1 to MaxIteration step 1 do
  for d = 1 to D step 1 do
    remove statistic on d
    resample t_d according to (3)
    resample π_d according to (4)
    resample k_d according to (5)
    add back statistic on d
  end for
  for k = 1 to K step 1 do
    resample π'_k according to (6)
  end for
end for
```

5 Experiment

In this section, we apply our algorithm on several real-world online review datasets. Our experiment demonstrates the competitive grouping performance and show how different groups with specific aspect interests are discovered from data collections.

There are two main kinds of datasets designed with specific evaluation purpose.

5.1 Datasets

1. *Amazon* (AZ)[2] Crawled by [16] in 2006, the Amazon review dataset from www.amazon.com contains reviews of manufactured products. We choose several product categories and select a subset under several memberIds (reviewers) for each category. When preprocessing the data, review spams are found to

[2] http://liu.cs.uic.edu/download/data/

Table 2 Statistics on Online Review Datasets

Dataset	*AZ: Camera*	*AZ: Computer*	*AZ: Apparel*$_1$	*AZ: Apparel*$_2$	*AZ: Electronics*	*OR:hotel: beijing*	*OR:hotel: chicago*
#Reviews	192	67	110	92	480	5256	5000
#memberIds	3	3	3	5	4	N/A	N/A
avgTxtLen	227	265	84	155	153	169	165

exist in the form of duplicate reviews with different productIds under the same memberId. So we remove those spams by pair-wise checking through *TF-IDF* based cosine similarity (threshold is 0.9).
With the availability of memberId attribute which indicates the author of the reviews, we treat it as the true class label, through which we evaluate the grouping performance in the form of review clustering, by checking whether or not reviews contributed by the same author will be clustered into the same group. One point to emphasize is that in this task we only make use of part of the learning result from our algorithm, that is the clusterId assigned to each document.

2. *OpinionRank* (*OR*)[3] The OpinionRank review dataset [12] contains full reviews for cars and hotels from Edmunds and Tripadvisor. We choose the hotel reviews under *hotel:beijing* and *hotel:chicago* to demonstrate the resulting groups reflecting users' discriminative topic interests, and each topic is presented by its top 20 words in the corresponding language models.

The general statistics over those datasets are shown in Table 2.

5.2 Evaluation methodology

For the AZ dataset, by considering the memberId as the true class label, we can treat it as a text clustering problem and we evaluate the clustering performance via three popular criterion functions: F-score [35], Purity [35], Normalized Mutual Information (*NMI*) [36], as summarized below:

– FScore

$$Prec(i, j) = \frac{n_{ij}}{n_i}, \; Rec(i, j) = \frac{n_{ij}}{n_j},$$

$$FScore(i, j) = \frac{2\,Prec(i, j)\,Rec(i, j)}{Prec(i, j) + Rec(i, j)}$$

$$FScore = \sum_{i=1}^{K} \frac{n_i}{n} Max(FScore(i, *)) \tag{10}$$

– Purity

$$Purity = \frac{1}{n} \sum_{i=1}^{K} Max(n_{i*}) \tag{11}$$

[3] http://kavita-ganesan.com/entity-ranking-data/

– *NMI*

$$NMI(X,Y) = \frac{I(X,Y)}{\sqrt{H(X)H(Y)}} = \frac{\sum_{i=1}^{K}\sum_{j=1}^{C} n_{ij} log\left(\frac{n*n_{ij}}{n_i*n_j}\right)}{\sqrt{\left(\sum_{i=1}^{K} log\left(\frac{n_i}{n}\right)\right)\left(\sum_{j=1}^{C} n_j log\left(\frac{n_j}{n}\right)\right)}} \quad (12)$$

In the above equations, n is the total number of documents, n_{ij} is the member size of class j in cluster i, n_i is the member size of cluster i and n_j is the member size of class j. X is the cluster label variable for cluster assignments while Y is the actual class label variable. K represents the total cluster number while C represents the class number.

For the *OR* dataset, we split our data into train set and test set and compute the likelihood of the test data given our learned model from the train data to evaluate its effectiveness. Therefore, perplexity, originally introduced in language modeling [3], is used. Perplexity is the inverse of the geometric mean per-word likelihood defined as follows:

$$Perplexity(D_{\text{test}}) = exp - \frac{\sum_{d\in D_{\text{test}}} logp(w_d|M)}{\sum_{d\in D_{\text{test}}} N_d} \quad (13)$$

M is the learned model from the train set. The $p(w_d|M)$ is estimated by query sampling of the test set on the learned model, we keep the trained model fixed and do several sampling iterations on the test set to get the topic assignment for each word [15]. A lower perplexity indicates better generalization performance on the held-out test set.

5.3 Parameter setting

As suggested by Griffiths and Steyvers [13], the Dirichlet hyper-parameters are set as follows: $\alpha_0 = 50.0/K$, $\beta_0 = 0.1$, $\theta_0 = 50.0/T$. For parameter $\boldsymbol{\pi}_0$, it is regarded as the natural permutation over T topics: $\{1, \cdots, T\}$ without loss of generality. To simplify the learning process, we set the parameter ρ to be a scalar, instead of being a T-1 dimensional vector in *GMM*, which reduces the *Generalized Mallows Model* to be a standard *Mallows Model*. By experience, setting $\rho = 1$ results in a good balance between the real-world ordering randomness and punishment of ordering dispersion.

For dataset *AZ*, we set $T = 10$ by experience, and K is set to be the distinct count of memberId (which is 3, 3, 3, 5, 4 for *AZ:Camera, AZ:Computer, AZ:Apparel*$_1$, *AZ:Apparel*$_2$, *AZ:Electronics*, respectively). For dataset *OR*, since we do not have their author Ids, we set $T = 6$ and $K = 3$ to demonstrate the word distribution of six topics and specific aspect interests of three interest groups.

5.4 Comparison of perplexity with *LDA* on *OR* dataset

Unlike the *APT* and *AIT* models who focus on the authors' specific interests on academic literature application, we are interested in the common product interests. Especially in the review analysis application, the authorId is not given in most of the

cases (the *OR* dataset for example). Although sometimes we have the authorId for some dataset (the *AZ* dataset for example), it is sparse, that is, we have large size of distinct authors but each of them just have very few reviews.

So we choose *LDA* as the baseline directly and evaluate their performance in perplexity. The hyper-parameter of *LDA* is set to be the same as *KMM*. We use the GibbsLDA++ [26] software to apply *LDA* on our dataset.

We use 5-fold Cross-Evaluation on the whole data, that is , for each fold, we random select 20 % reviews as the test set (held-out), and the remain 80 % as the train set (held-in), each document get one chance to be a test data. For both *LDA* and *KMM*, we learn the model from the train set and compute the perplexity on the test set. Figure 5 show the comparison on dataset *OR:hotel:beijing* and *OR:hotel:chicago*. A lower perplexity measure means a better generalization ability on the test set, that is, we have less surprise to see the test set, demonstrating a better model fitting to the dataset.

Comparing to *LDA*, *KMM* reduce the perplexity of held-out set on the average of 27.1 % on *OR:hotel:beijing* and 60.8 % on *OR:hotel:chicago*. This is because *KMM* adds structural consideration on modeling which can help the model capture the corpus' inner topic structure, while in *LDA*, it assumes no direct relationship between topics under the bag-of-topics assumption.

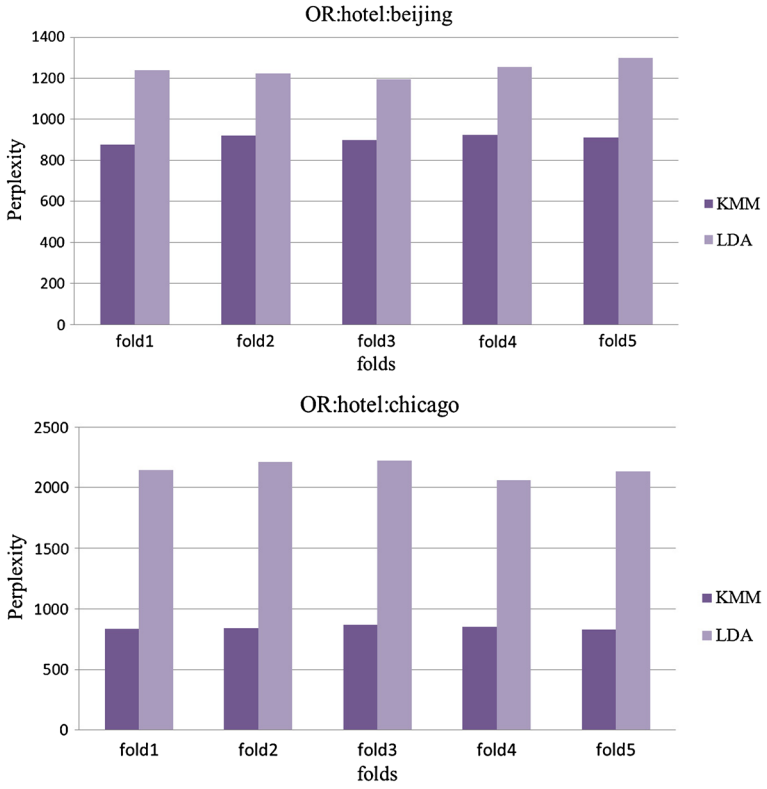


Figure 5 Perplexity comparison between *KMM* and *LDA* on *OR* datasets

5.5 Comparison of review clustering performance with K-Means on AZ

In this section, the task is to cluster reviews under the same author into the same group. For our algorithm, this task is only one part of the learned model, that is, the clusterId for each review document.

With the availability of authorId, we do experiment on *AZ:Camera, AZ:Computer, AZ:Appeal*$_1$, *AZ:Appeal*$_2$ and *AZ:Electronics*, each of which contains reviews selected from 3, 3, 3, 5, 4 different reviewers, respectively. Accordingly, K is set to be 3, 3, 3, 5, 4 respectively.

As we have already known the K, it is straightforward to use one of the most popular clustering algorithm K-Means as our baseline, and the pair-wise similarity is computed using the standard cosine similarity on the *TF-IDF* based *Vector Space Model* [30].

The performance comparison is shown in Figure 6. As observed, our *KMM* algorithm beats the K-Means baseline for datasets: *AZ:Camera, AZ:Computer*, *AZ:Apparel*$_1$ and *AZ:Electronics*, with respective improvement being around 54.3, 2.3, 53.7 and 103.2 % in terms of *NMI*.

When we look into the *AZ:Apparel*$_2$ which contains reviews selected from five distinct reviewers, we find that the performance is not as good as K-Means. The reason is due to that our clustering result is based on the identification of the common taste in product aspect interests, while K-Means is based on the *Vector Space Model* in individual review text. With the limited number of reviews for each reviewer, K-Means tends to be easier to discriminate individual reviewer, whereas in our model reviews from different reviewers could be clustered into one group when they share common aspect interests.

Our model transfers the data points from a high dimensional space (Vector Space Model) into a much smaller dimensional space based on common product interests with our concern. As a result, it indicates that our algorithm is not adapted for the task of individual person identification or authorship identification which requires much more refined individual characteristics.

5.6 Illustration of interest groups with specific aspect interests

In Section 5.5, the task is to cluster all reviews contributed by the same author into the same group, where we just make use of the clusterId learned from the model. To illustrate the validity of interest grouping, we further do experiment on *OR:hotel: beijing* by setting $K = 3$, $T = 6$, and present the learning result: interest groups and topics.

All six topics' definitions are shown in Table 3[4] where *L&S* means the *Location & Surrounding*. The probability is estimated by:

$$\hat{\beta}_{t,w} = \frac{N_\beta(t, w) + \beta_0}{N_\beta(t) + V\beta_0} \tag{14}$$

[4]only top 20 words are presented, and bold keywords indicate highly relevant features of a particular topic.

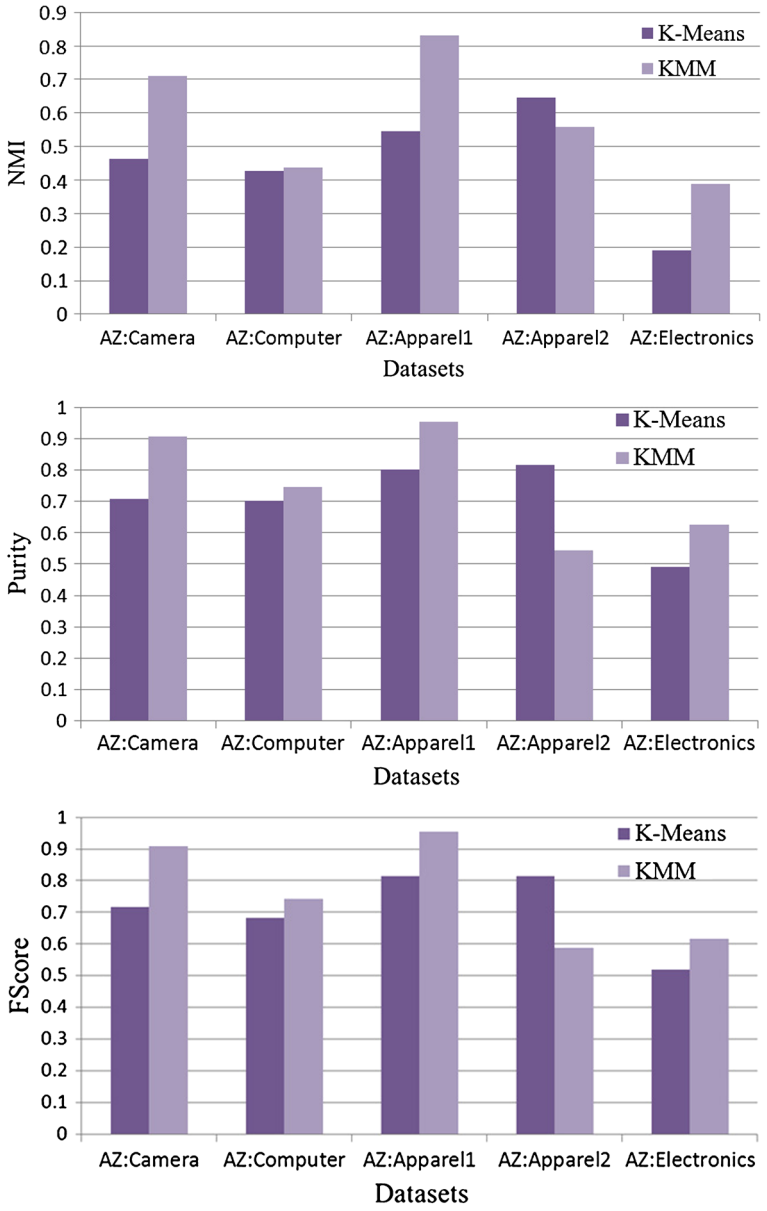


Figure 6 Clustering performance on *AZ* datasets

where $N_\beta(t, w)$ is the total number of times word w is assigned to topic t, $N_\beta(t)$ is the total number of words assigned to topic t, and V is the vocabulary size.

As observed from Table 3, we have identified six topics: $\{T_1 : General_1,\ T_2 : Room,\ T_3 : Food,\ T_4 : L\&S,\ T_5 : General_2,\ T_6 : Service\}$. All these topics are clearly identified by their top words except for the $General_1$ and $General_2$ which

Table 3 Top 20 words in topics' word distributions from *OR:hotel:beijing*

General$_1$	%	*Room*	%	*Food*	%	*L&S*	%	*General*$_2$	%	*Service*	%
hotel	6.62	**room**	4.73	**breakfast**	3.21	hotel	3.87	hotel	4.69	hotel	2.85
stayed	3.64	**rooms**	2.29	hotel	1.98	**location**	2.18	stay	2.10	**staff**	2.34
beijing	3.00	hotel	1.57	good	1.69	city	2.09	beijing	1.80	**english**	1.34
nights	1.79	**clean**	1.40	**food**	1.64	**walk**	2.06	staff	1.18	**helpful**	1.09
stay	1.36	**bathroom**	1.32	**buffet**	1.38	**shopping**	1.50	good	1.18	taxi	0.97
China	1.03	**comfortable**	1.05	room	1.31	**forbidden**	1.47	service	1.00	great	0.93
trip	0.84	**bed**	1.04	**restaurant**	1.26	**subway**	1.34	great	0.97	room	0.81
hotels	0.81	nice	1.02	great	0.91	**street**	1.31	recommend	0.94	**desk**	0.77
great	0.78	**shower**	0.83	**chinese**	0.89	beijing	1.24	room	0.72	quot	0.74
room	0.75	**large**	0.77	service	0.87	**taxi**	1.01	hotels	0.69	**service**	0.73
location	0.75	good	0.74	pool	0.86	**walking**	1.00	location	0.68	**friendly**	0.72
booked	0.72	floor	0.72	nice	0.70	**square**	0.97	place	0.60	chinese	0.69
days	0.69	**beds**	0.63	staff	0.66	great	0.91	time	0.58	**concierge**	0.69
good	0.65	**tv**	0.62	**western**	0.62	good	0.89	definitely	0.57	beijing	0.63
reviews	0.60	**spacious**	0.59	**free**	0.62	**station**	0.88	go	0.54	day	0.62
night	0.59	great	0.57	excellent	0.62	**minutes**	0.87	quot	0.52	good	0.60
business	0.58	**water**	0.52	**restaurants**	0.59	**close**	0.82	rooms	0.52	front	0.60
plaza	0.53	quot	0.49	quot	0.58	**located**	0.81	back	0.49	driver	0.59
star	0.50	staff	0.48	day	0.57	**distance**	0.79	China	0.49	wall	0.58
holiday	0.49	modern	0.47	internet	0.51	**area**	0.78	star	0.49	time	0.54

contain some general words instead of specific aspect related words. This is reasonable as people may also express some general idea about products, so we treat this kind of general topics as "General Aspects".

Similarly, the parameter $\theta_{k,t}$, representing the topic frequency of product interests, can be estimated by:

$$\hat{\theta}_{k,t} = \frac{N_\theta(k,t) + \theta_0}{N_\theta(k) + T\theta_0} \tag{15}$$

where $N_\theta(k,t)$ is the total number of sentences assigned to topic t in cluster k, and $N_\theta(k)$ is the total number of sentences in cluster k.

The orderings: $\{\boldsymbol{\pi}'\}$ of topics and their frequencies $\{\boldsymbol{\theta}_k\}$ (computed based on (15)) for all the three interest groups are listed in Table 4. We can see that C_0 group pay more attentions to *Room* and do not care *Service* very much. On the other hand, C_1 group care for the *Service* mostly, while C_2 group is mostly interested in *Food* and

Table 4 Three interest groups with specific topic frequency and order learned from *OR:hotel:beijing*

C_0		C_1		C_2	
Ordering	Freq (%)	Ordering	Freq (%)	Ordering	Freq (%)
General$_1$	12.7	*General*$_1$	12.4	*General*$_1$	9.1
Room	25.8	*Room*	13.5	*General*$_2$	13.5
Service	10.7	*L&S*	10.3	*Room*	17.2
Food	17.5	*Food*	11.6	*Food*	37.7
L&S	16.9	*Service*	35.5	*Service*	13.7
General$_2$	16.4	*General*$_2$	16.7	*L&S*	8.8

Table 5 Three interest groups with specific topic frequency and order learned from *OR:hotel:chicago*

C_0		C_1		C_2	
Ordering	Freq (%)	Ordering	Freq (%)	Ordering	Freq (%)
$General_1$	16.5	$General_1$	10.7	$General_1$	9.5
Room	21	*L&S*	7.7	*Room*	10.9
L&S	19.5	*Service*	16.1	*Service*	60
Service	14.7	*Room*	29	*Food*	4.9
Food	12	*Food*	22.6	G_2	12.2
$General_2$	16.3	$General_2$	13.9	*L&S*	2.5

less interested in *L&S*. Besides the topic frequencies, we also see that while their topic orderings are quite different, they all start from $General_1$, which reflects the fact that people often give a general description first, followed by different aspects w.r.t. their own interests.

Table 5 shows another grouping example from *OR:hotel:chicago*, with top 20 keywords shown in Table 6 where bold keywords indicate highly relevant features of a particular topic.

Users' interest groups as illustrated in Tables 4 and 5 are one kind of high level informative summarization focusing on customers' main product concerns. This summarization can help hotel managers make improvement in respective aspects with consideration on customers' specific interests, and may also help travel advisors develop different suggestions to accommodate various customers.

Table 6 Top 20 words in topics' word distributions from *OR:hotel:chicago*

$General_1$	%	*Service*	%	*Room*	%	*L&S*	%	*Food*	%	$General_2$	%
hotel	5.08	room	2.81	**room**	5.16	**location**	2.93	**breakfast**	2.61	hotel	4.39
stayed	3.54	**staff**	2.23	**clean**	1.81	hotel	2.92	room	1.73	stay	3.71
chicago	2.71	hotel	2.13	**rooms**	1.77	great	1.84	hotel	1.73	chicago	2.15
stay	1.90	**desk**	1.39	hotel	1.52	**michigan**	1.70	**free**	1.29	great	1.56
night	1.26	**front**	1.21	**bed**	1.47	**walk**	1.60	**nice**	1.18	recommend	1.15
great	1.23	quot	1.21	**bathroom**	1.41	**blocks**	1.36	**great**	1.17	location	1.02
nights	1.11	**friendly**	1.00	nice	1.25	**walking**	1.28	**good**	1.13	good	0.99
room	1.04	**helpful**	0.95	**comfortable**	1.19	**mile**	1.28	restaurant	1.01	definitely	0.97
booked	1.02	**service**	0.66	floor	0.83	**restaurants**	1.13	service	0.96	room	0.97
location	1.00	**check**	0.65	**shower**	0.81	**chicago**	1.09	staff	0.94	place	0.95
weekend	0.97	night	0.62	**great**	0.76	**distance**	1.04	**food**	0.82	price	0.88
reviews	0.83	time	0.57	**view**	0.76	**shopping**	1.03	**bar**	0.78	hotels	0.75
hotels	0.81	day	0.55	**beds**	0.74	**close**	1.02	**coffee**	0.75	night	0.75
price	0.68	rooms	0.55	**large**	0.72	right	1.00	quot	0.68	parking	0.73
trip	0.64	floor	0.52	quot	0.68	**park**	0.94	lobby	0.66	time	0.72
quot	0.62	stay	0.49	**tv**	0.55	ave	0.88	pool	0.56	back	0.70
time	0.58	**told**	0.49	good	0.53	**block**	0.87	day	0.53	go	0.67
good	0.57	back	0.46	**suite**	0.46	**away**	0.85	internet	0.51	city	0.59
rate	0.55	**nice**	0.45	**spacious**	0.45	**street**	0.77	morning	0.48	staff	0.59
place	0.50	called	0.44	night	0.43	good	0.74	floor	0.47	staying	0.53

6 Conclusion

Online data such as review texts can reveal much useful business information. The *KMM* topic model aims at efficiently identifying hidden interest groups accompanied by a detailed explanation in the forms of topic frequency and order. Both the topic frequency and order can exhibit customers' product concerns and jointly present an informative summarization. The experiment proves its effectiveness. The learned interest groups can be applied widely, such as in product improvement, marketing strategy development, or customer-targeting online advertisement plans.

The incorporation of structural restrictions into a traditional bag-of-topics topic model, such as *LDA*, can greatly improve the model's expressive power, which is important to automatic text understanding. We plan to develop an even more sophisticated and adaptive structural model in our subsequent research.

Acknowledgements The work described in this article has been supported by a strategic research grant from City University of Hong Kong (project no. 7002770), the NSFC Project (61272275, 61070011), and the 111 Project (B07037).

References

1. Abdul-Mageed, M., Diab, M.T., Korayem, M.: Subjectivity and sentiment analysis of modern standard arabic. In: ACL (Short Papers)'11, pp. 587–591 (2011)
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th International Conference on Very Large Data Bases, pp. 487–499. VLDB '94, Morgan Kaufmann Publishers Inc., San Francisco (1994). http://dl.acm.org/citation.cfm?id=645920.672836
3. Azzopardi, L., Girolami, M., van Risjbergen, K.: Investigating the relationship between language model perplexity and ir precision-recall measures. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval, pp. 369–370. SIGIR '03, ACM, New York (2003). doi:10.1145/860435.860505
4. Beineke, P., Hastie, T., Manning, C., Vaithyanathan, S.: An exploration of sentiment summarization. In: Proceeding of AAAI, pp. 12–15 (2003)
5. Bernardo, J.M., Smith., A.F.: Bayesian Theory. Wiley Series in Probability and Statistics (2000)
6. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, New York (2006)
7. Blei, D.M., Griffiths, T.L., Jordan, M.I.: The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. J. ACM **57**, 7:1–7:30 (2010)
8. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
9. Chen, H., Branavan, S.R.K., Barzilay, R., Karger, D.R.: Content modeling using latent permutations. J. Artif. Intell. Res. (JAIR) **36**, 129–163 (2009)
10. Fligner, M.A., Verducci, J.S.: Distance based ranking models. J. Roy. Stat. Soc. B Met. **48**(3), 359–369 (1986)
11. Gamon, M., Aue, A., Corston-Oliver, S., Ringger, E.K.: Pulse: Mining customer opinions from free text. In: IDA'05, pp. 121–132 (2005)
12. Ganesan, K., Zhai, C.: Opinion-Based Entity Ranking. Information Retrieval (2011)
13. Griffiths, T.L., Steyvers, M.: Finding scientific topics. PNAS 101(suppl. 1), 5228–5235 (2004)
14. Gruber, A., Rosen-Zvi, M., Weiss, Y.: Hidden topic Markov models. In: Artificial Intelligence and Statistics (AISTATS). San Juan, Puerto Rico (2007)
15. Heinrich, G.: Parameter estimation for text analysis. Tech. Rep., University of Leipzig, Germany (2004). http://www.arbylon.net/publications/text-est.pdf
16. Jindal, N., Liu, B.: Opinion spam and analysis. In: Proceedings of the International Conference on Web Search and Web Data Mining, pp. 219–230. WSDM '08 (2008)

17. Jindal, N., Liu, B., Lim, E.P.: Finding unusual review patterns using unexpected rules. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 1549–1552. CIKM '10 (2010)
18. Jordan, M. (ed.): Learning in Graphical Models. MIT Press, Cambridge (1999)
19. Kawamae, N.: Author interest topic model. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 887–888. SIGIR '10, ACM, New York, NY, USA (2010). doi:10.1145/1835449.1835666
20. Leung, C.K., Chan, S.F., Chung, F.L., Ngai, G.: A probabilistic rating inference framework for mining user preferences from reviews. World Wide Web **14**(2), 187–215 (2011). doi:10.1007/s11280-011-0117-5
21. Li, W., McCallum, A.: Pachinko allocation: Dag-structured mixture models of topic correlations. In: ICML (2006)
22. Liu, B.: Opinion observer: Analyzing and comparing opinions on the web. In: Proceedings of the 14th international conference on World Wide Web, pp. 342–351. WWW'05 (2005)
23. Mei, Q., Liu, C., Su, H., Zhai, C.: A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In: Proceedings of the 15th International Conference on World Wide Web, pp. 533–542. WWW '06 (2006)
24. Mukherjee, A., Liu, B., Glance, N.: Spotting fake reviewer groups in consumer reviews. In: Proceedings of the 21st International Conference on World Wide Web, pp. 191–200. WWW '12, ACM, New York, NY, USA (2012). doi:10.1145/2187836.2187863
25. Mukherjee, A., Liu, B., Wang, J., Glance, N., Jindal, N.: Detecting group review spam. In: Proceedings of the 20th International Conference Companion on World Wide Web, pp. 93–94. WWW '11 (2011)
26. Phan, X.H., Nguyen, C.T.: Gibbslda++: A c/c++ implementation of latent dirichlet allocation (lda) (2007)
27. Popescu, A.M., Etzioni, O.: Extracting product features and opinions from reviews. In: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 339–346. HLT '05 (2005)
28. Purver, M., Griffiths, T.L., Körding, K.P., Tenenbaum, J.B.: Unsupervised topic modelling for multi-party spoken discourse. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, pp. 17–24. ACL-44 (2006)
29. Rosen-Zvi, M., Griffiths, T., Steyvers, M., Smyth, P.: The author-topic model for authors and documents. In: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, pp. 487–494. UAI '04, AUAI Press, Arlington, Virginia, United States (2004). http://dl.acm.org/citation.cfm?id=1036843.1036902
30. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**, 613–620 (1975). doi:10.1145/361219.361220
31. Sensoy, M., Yolum, P.: Automating user reviews using ontologies: an agent-based approach. World Wide Web **15**(3), 285–323 (2012). doi:10.1007/s11280-011-0134-4
32. Si, J., Li, Q., Qian, T., Deng, X.: Discovering k web user groups with specific aspect interests. In: Proceedings of Machine Learning and Data Mining in Pattern Recognition, pp. 321–335. MLDM 2012 (2012)
33. Titov, I., McDonald, R.: Modeling online reviews with multi-grain topic models. In: Proceeding of the 17th International Conference on World Wide Web, pp. 111–120. WWW '08 (2008)
34. Wang, H., Zhang, D., Zhai, C.: Structural topic model for latent topical structure analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1, pp. 1526–1535. ACL-HLT '11, Association for Computational Linguistics, Stroudsburg, PA, USA (2011). http://dl.acm.org/citation.cfm?id=2002472.2002657
35. Zhao, Y., Karypis, G.: Criterion Functions for Document Clustering: Experiments and analysis. Tech. Rep., University of Minnesota Press, Minneapolis (2002)
36. Zhou, X., Zhang, X., Hu, X.: Semantic smoothing of document models for agglomerative clustering. In: Proceeding 20th International Joint Conf. Artificial Intelligence, pp. 2928–2933. IJCAI' 07 (2007)