SAR: A Sentiment-Aspect-Region Model for User Preference Analysis in Geo-tagged Reviews

Kaiqi Zhao, Gao Cong, Quan Yuan
School of Computer Engineering
Nanyang Technological University, Singapore
{kzhao002@e., gaocong@, qyuan1@e.}ntu.edu.sg

Kenny Q. Zhu
Department of Computer Science & Engineering
Shanghai Jiao Tong University, Shanghai, China
kzhu@cs.sjtu.edu.cn

***Abstract*—Many location based services, such as FourSquare, Yelp, TripAdvisor, Google Places, etc., allow users to compose reviews or tips on points of interest (POIs), each having a geographical coordinates. These services have accumulated a large amount of such geo-tagged review data, which allows deep analysis of user preferences in POIs. This paper studies two types of user preferences to POIs: topical-region preference and category aware topical-aspect preference. We propose a unified probabilistic model to capture these two preferences simultaneously. In addition, our model is capable of capturing the interaction of different factors, including topical aspect, sentiment, and spatial information. The model can be used in a number of applications, such as POI recommendation and user recommendation, among others. In addition, the model enables us to investigate whether people like an aspect of a POI or whether people like a topical aspect of some type of POIs (e.g., bars) in a region, which offer explanation for recommendations. Experiments on real world datasets show that the model achieves significant improvement in POI recommendation and user recommendation in comparison to the state-of-the-art methods. We also propose an efficient online recommendation algorithm based on our model, which saves up to 90% computation time.**

I. Introduction

Nowadays, users can easily convey their opinions on points of interest (POIs) by tapping on their smart mobile devices in location based services (LBS) like Yelp, TripAdvisor, etc. These LBS systems contain three kinds of useful information for user preference modeling. First, they provide a large amount of user reviews on POIs. Different from tips in FourSquare or geo-tagged tweets in Twitter, the user reviews contain more details about why the users like/dislike the POI, which aspects of the POI satisfy them while which aspects dissatisfy them. The availability of such reviews makes it possible to model user preferences on the aspect level. Second, the geographical coordinates of the POIs in these systems reveal the users' activity areas and their spatial preferences. For example, some users may like to visit a shopping street while some often visit a region famous with bars. Third, the category of POIs may help analyze the aspects of the POIs and the aspect preferences of users on certain category of POIs, because POIs in the same category share some common aspects (e.g., room cleanliness/comfort of hotels, taste of restaurants, etc.).

Recently, several studies on geographical topic modeling [5], [27] model user preferences on geo-tagged tweets. User preferences are modeled as a distribution over topical-regions (called *Topical-Region Preference*). A topical-region represents a geographical area in which users do similar things (such as dining). It comprises two components: geo-location and semantics. For example, POIs in Central Park and those on Wall Street, Manhattan may form two different topical-regions. The ones in Central Park may have tweets that contain words like concert, ticket, bird, running, etc., while the ones on Wall Street may have tweets about stocks or finance. However, geo-tagged tweets are too short for aspect extraction and the aforementioned studies do not model the user preferences on aspect level, i.e., they are not able to capture the *Topical-Aspect Preferences*. Topical-aspects are the aspects of POIs that are commented by users, such as environment, taste, price, etc. An example for the topical-aspect preferences is that a user may like the environment in western restaurants, but the good taste in Chinese restaurants. Moreover, these topical aspect preferences are often category-aware as illustrated in the previous example. To the best of our knowledge, no existing work on modeling geo-tagged textual data models users' topical-aspect preferences.

On the other hand, several proposals [13], [18], [19], [7] on opinion mining aim at identifying latent topical-aspects and predicting ratings/sentiments on identified aspects in product reviews. Most of these studies aim at mining item aspects and sentiments but not user preferences, and thus they do not incorporate user information in their models. Zhang et al. [29] consider aspect, sentiment and user information for recommendation. However, their work and all of the aforementioned proposals for opinion mining do not consider the geo-tags in the reviews, which are important signals for modeling user preferences on spatial data.

In summary, neither existing geographical models nor opinion mining models consider aspects, sentiment, regions and category at the same time.

In this paper we propose a novel, unified model to learn user preferences based on reviews, categories and geolocations of POIs. Our model is able to capture the interdependency of three latent factors including topical-region, topical-aspect, and sentiment simultaneously for identifying the topical-region and topical-aspect preferences for each user. There are three benefits to model the interdependencies. First, the learning of topical-region preference benefits from topical-aspects and sentiment. Because a user may not like the place she visited before, mining topical-regions based on visit history without considering the user's sentiment would lead to incorrect user preference on some regions. We solve this problem by in-

corporating aspect and sentiment in the learning of topical-region. Second, the topical-aspect extraction benefits from the topical-regions. Some words in the text are related to functional and spatial information of the region, e.g., "restaurant", "New York", etc. The topical-regions help recognize these words and make the topical-aspects more accurate, i.e., "restaurant" and "New York" would not appear as representative words in topical-aspects. Third, we can apply the model to many applications, such as POI recommendation, user recommendation and aspect satisfaction analysis in regions, etc., and answer questions like:

- Which aspect of a location is favored by people and which is not?
- Which POI would a user like to visit?
- Who would be interested in a given POI?
- What is the overall positive aspect of POIs in the same category in each region?

Building such a unified model is a challenging task. First, the interaction among the three types of latent variables (for topical-aspect, sentiment, topical-region, respectively) and the category of POIs is unknown. Second, aspect and sentiment are usually modeled at sentence level [7] while region is modeled at the review or document level [5], [25], which makes it difficult to estimate the parameters of a unified model. To overcome the first challenge, we model both the category-aware topical-aspect preferences and the topical-region preferences as conditional multinomial distributions. In addition, we propose a generative process of the review words, which generates both topical-aspect words and topical-region words, to capture the implicit interaction between topical-aspects and topical-regions. To overcome the second challenge, we propose a two-level expectation-maximization inference algorithm. We estimate the document-level parameters (related to topical-regions) in the first step and the sentence-level parameters (related to topical-aspects and sentiments) in the second step.

We demonstrate the effectiveness of our model on various applications such as POI recommendation, user recommendation, and aspect satisfaction analysis in regions. We also develop an efficient algorithm to speed up online recommendation based on our model.

In summary, this paper makes the following contributions:

1) We propose a novel unified probabilistic model to capture the interaction of aspect, sentiment, category as well as spatial information, and an inference algorithm to estimate the model parameters;
2) We apply our proposed model to recommend POIs and users and analyze the aspect satisfaction in regions. To the best of our knowledge, this is the first work for user recommendation; We also propose an efficient online recommendation algorithm using our model;
3) We evaluate our model on real world datasets and the experimental results show that the model outperforms the state-of-the-art methods in POI recommendation and user recommendation; We also demonstrate that our model is able to offer explanation for recommendations while the baseline methods [22] fail to offer.

The rest of the paper is organized as follows: Section II introduces related work; Section III presents our unified model and corresponding inference algorithm; Section IV applies our model to several applications; Section V demonstrates the model selection and tuning and then evaluates the model on three recommendation tasks. Section VI concludes this paper.

II. Related Work

A. Geographical Topic Modeling

Based on the traditional topic modeling techniques such as Latent Dirichlet Allocation (LDA) [1] and Probabilistic Latent Semantic Analysis (PLSA), some recent studies [3], [17], [5], [27] incorporate geographical information in topic models. Benefit from the geographic information, these models can discover regional topics, which are shown to be useful in POI recommendation.

Some of these studies [14], [3], [17], [16], [25] focus on analyzing the relation between locations and words/topics without considering users. Mei et al. [14] sample each word in their model conditioned on the time, location (id only) and background words. Eisenstein et al. [3] consider the coordinates of locations and use the Gaussian distribution to generate coordinates for locations from latent regions. GeoFolk [17] generates latitude and longitude of a location from two Gaussian distributions determined by the topic of the location's document. Yin et al. [25] propose a PLSA model with two latent variables, region and topic, in two levels. Regions are generated in the document level, which is shared by all of the documents. Topics are generated in the word level. Each latent region is modeled as a Gaussian distribution in their model. Different from these proposals, our work focuses on modeling user preferences and takes users and users' sentiment into account.

There also exist proposals [5], [27], [24] that model user preferences to geographical topics based on geo-tagged tweets. Hong et al. [5] propose a model to analyze the geographical topics in geotagged tweets. In this model, latent regions are modeled as Gaussian distributions and each region contains a topic distribution. Yuan et al. [27] explore the temporal information together with the location, topic, and user information to model the time-aware personalized topic region. Yin et al. [23] consider that the user behaviors depend on both the user's topic preferences and temporal topic distributions. They incorporate category and locality preference into consideration to make further improvement on modelling user profiles [24]. Our work is different from these proposals in that we focus on user reviews rather than tweets, and we not only explore the topic-regions but also analyze the topical-aspects and corresponding sentiments in the reviews.

B. Sentiment Analysis

Our work is also related to the work on sentiment aspect joint modeling. According to different granularities of sentiment, we divide the existing studies into two types: sentence level, and phrase level.

Sentence level sentiment analysis supposes each sentence expresses one aspect of the product. Titov and McDonald [18] present a sentence level model, namely MG-LDA, to

extract aspects from reviews. Based on MG-LDA, Titov and McDonald further propose a method [19] which jointly models the aspect and rating. The model is then used to predict aspect ratings from a review. This model can extract aspect level sentiment/rating, but needs aspect rating as observable variable, which is often not available. Jo and Oh [7] model aspect and sentiment jointly in an LDA-based framework. Both aspect and sentiment are modeled as latent variables in the model, and each word has a joint distribution over topics and sentiment polarities. To identify the polarities, they keep a list of seed words for each sentiment polarity, and give higher probability to generate a seed word from its corresponding polarity.

Studies on phrase level sentiment [13], [20], [12], [15], [30] use NLP tools to analyze the dependency between the words in a sentence and extract aspect-opinion phrase pairs, e.g., $< screen, bright >$, with some predefined patterns. The aspect-opinion phrase pairs are then used for further analysis on the sentiment polarities on the aspects. Mei et al. [13] build a PLSA model in which a topic is modeled as a linear mixture of multinomials from neutral topics and two sentiments (positive and negative). Wang et al. [20] build a regression model to capture the aspect ratings from the overall ratings. The overall ratings are modeled as the weighted sum of the sentiments from all aspects. However, aspects are fixed and extracted using a list of seed words and a bootstrapping algorithm. Moghaddam and Ester [15] propose an interdependent LDA model in which the aspects of a product is modeled and the corresponding ratings to each aspect is drawn depending on the aspect. However, the ratings are treated as 5 clusters and the model cannot tell which cluster has the rating of 1 and which has the rating of 5.

Our model differs from these models in that we jointly consider region, aspect and sentiment in a unified model.

C. POI recommendation

We divide the existing approaches into three categories: memory-based collaborative filtering, matrix factorization and topic models. Since topic models are mentioned in Section II-A, we focus on memory-based collaborative filtering and matrix factorization in this section.

Several proposals recommend POIs based on collaborative filtering (CF) [22], [9], [26]. Ye et al. [22] propose a fusion framework to combine user-based, friend-based and geo-based collaborative filtering. In the geographic model, the probability of transporting from one POI to another is drawn from a power law distribution over the distances between the two POIs. Levandoski et al. [9] use an item-based CF for POI recommendation, but they mainly focus on how to make the memory based method efficient on a large dataset. Yuan et al. [26] propose the problem of recommending POIs for a user specified time, and incorporate the temporal factor into the user-based CF model for recommendation. Yuan et al. [28] also propose a graph-based approach for time-aware POI recommendation which integrates geographical and temporal influences.

In the proposals based on matrix factorization, Liu et al. [10] and Cheng et al. [2] propose latent factor models by incorporating the geographical information using Gaussian distribution. Yang et al. [21] propose a sentiment-enhanced personalized location recommendation system using probabilistic matrix factorization. Very recently, Zhang et al. [29] propose an explicit factor model which takes aspect and sentiment into account. However, these proposals do not consider the geographical information.

In summary, no existing work models aspect, sentiment, spatial information and category at the same time. And no existing work is able to discover the latent relation between these variables.

III. Sentiment-Aspect-Region Model

We first present our objectives to build the unified sentiment-aspect-region model. To achieve the objectives, we present several intuitions based on which we build our model. We then describe the details of the model, and propose a parameter estimation method.

A. Intuitions

In this paper, we aim at building a model that is able to 1) extract latent variables, i.e., topical-aspect, sentiment, and topical-region from the review data; 2) capture the interdependencies among category, POI, user, words and the three latent variables; and 3) discover user's topical-region and topical-aspect preferences. To achieve these objectives, we exploit the following intuitions in designing our model:

Intuition 1: A user visits POIs in a topical-region because the region is geographically convenient to the user (e.g., close to her activity areas) and its topics (e.g., shopping street, education area, etc.) satisfy the user's interest. Each user has her own preferences on the topical-regions.

Intuition 2: A user rates highly of a POI because she likes some aspects of the POI. Such preferences might be indicated in her review. Some users like to check the price range of a restaurant first while others might be more concerned with the environment. Moreover, POIs in different categories may have different aspects of interest.

Intuition 3: A user decides to visit a POI in a region by considering the category, category-aware topical-aspects of the POI and the distance to it. For example, users may visit POIs of the restaurant category with good environment, but she may first consider the restaurants nearby.

Intuition 4: When a user writes a review on a POI, she will use words for both the aspects of the POI and her sentiments about the aspects. The user may also use words for the topical-region of the POI. For example, a review on a shop in Times Square may say: "This shop offers best prices in Times Square." The reviewer uses "price" for *aspect*, "best" for *sentiment* and "Times Square" for *region*.

B. Model Description

We first define the notations to be used in the proposed model. Let D be the set of user reviews, and U be the set of users. For each review, we denote the number of its sentences by M and number of words in each sentence by N. In our model, a location has two attributes: identifier and coordinates.

We use l to represent a location identifier and $\boldsymbol{cd}_l$ to denote its corresponding coordinates. Here $\boldsymbol{cd}_l$ is a latitude and longitude pair. We denote the topical-aspect, sentiment and topical-region by a, s, and r, respectively. The notations used in this paper are listed in Table I. Following the intuitions discussed in Section III-A, we proceed to present our model.

TABLE I: Description of Symbols

Symbol	Description
u, U	individual user and the set of users
l, L	individual POI and the set of POIs
c	category
r	topical-region
a, s	topical-aspect and sentiment
d, D	single review and the set of reviews
M	the number of sentences in a review
w, N	single word and the number of words in a sentence

Based on **Intuitions 1&2**, we model the user topical-region preferences and topical-aspect preferences as multinomial distributions $p(r|u)$ and $p(a|u,c)$, respectively.

Based on **Intuition 3**, a user chooses a POI to visit by considering both the category and the distance. We define the probability of visiting a POI l given category c and region r proportional to $p(l|c) \cdot p(l|r)$. Here $p(l|c)$ is the probability of selecting POI l from the category c; $p(l|r)$ is a the probability of selecting POI l in region r by considering the distance from l to r. After normalization, we have the definition $p(l|c,r) = \frac{p(l|c)p(l|r)}{\sum_{l'} p(l'|c)p(l'|r)}$. To model the spatial distance, we use a Gaussian mixture model, i.e., $p(l|r) \sim N(\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r)$, where $\boldsymbol{\mu}_r$ is the center of region r and $\boldsymbol{\Sigma}_r$ is the covariance matrix which depicts the area of region r. To model the membership of a POI to a category, we use a uniform distribution for $p(l|c)$.

Based on **Intuition 4**, we model the relationships among words, topical-aspects, sentiments and topical-regions by $p(w|a,s,r) = \lambda p(w|a,s) + (1-\lambda)p(w|r)$, where a, s, r are topical-aspect, sentiment and topical-region, respectively. Here $p(w|a,s)$ is the probability that the users write word w when they have sentiment s on aspect a; $p(w|r)$ is the probability that the users use word w to describe region r; parameter λ is used to balance the portion of words drawn from topical-aspect, sentiment or topical-region. We model $p(w|a,s)$ instead of $p(w|a)$ and $p(w|s)$ because aspects and sentiments are closely coupled, and modeling by $p(w|a)$ and $p(w|s)$ needs an additional tuning parameter. Similar to proposals of sentence level sentiment analysis [18], [19], [7], we assume each sentence expresses opinions on exactly one topical-aspect and each topical-aspect is associated to a positive, negative or neutral sentiment.

In summary, the graphical representation of our model is shown in Figure 1 and the generative process of the reviews written by user u is described as follows:

- For each review $d \in D_u$, where D_u is the set of reviews written by user u.
  - Draw topical region $r \sim p(r|u)$
  - Draw category $c \sim p(c|u)$
  - Draw location $l \sim p(l|c,r) = \frac{p(l|r)p(l|c)}{\sum_{l'} p(l'|c)p(l'|r)}$, where $p(l|r) \sim N(\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r)$

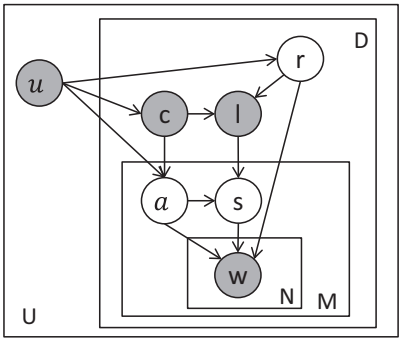


Fig. 1: Sentiment-Aspect-Region Model (SAR)

  - For each sentence in review d
    - Draw aspect $a \sim p(a|u,c)$
    - Draw sentiment $s \sim p(s|a,l)$
    - For each word position in the sentence
      - Draw word $w \sim p(w|a,s,r) = \lambda p(w|a,s) + (1-\lambda)p(w|r)$

In the model, $p(l|c)$ and $p(c|u)$ can be estimated directly from a given corpus. The other distribution parameters need to be inferred. We first present how to estimate $p(l|c)$ and $p(c|u)$, and then show the inference algorithm for the remaining distributions in Section III-C.

As described in **Intuition 3**, a POI l is generated from both category and region. Since POI l and category c are observable variables, we simply compute $p(l|c)$ by Equation (1).

$$p(l|c) = \frac{I(l,c)}{\#\ of\ POIs\ in\ c} \tag{1}$$

$$I(l,c) = \begin{cases} 1 & l \in c \\ 0 & otherwise \end{cases} \tag{2}$$

Similarly, we compute the category preferences of each user, i.e., $p(c|u)$, directly from the corpus. To handle the over-fitting problem, we apply the additive smoothing technique. After smoothing, even though a user did not a visit some category of POIs, the probability of visiting that category still has a small value. The computation of $p(c|u)$ is shown in Equation (3).

$$p(c|u) = \frac{n_c + \alpha}{N + \alpha C}, \tag{3}$$

where n_c is the number of reviews of POIs in category c that user u writes; N is the total number of reviews on POIs in c; C is the total number of categories; α is the smoothing parameter which is usually set to a value smaller than 1. In this paper, we set $\alpha = 0.1$.

C. Inference Algorithm

To infer the parameters of the model, we use the expectation-maximization (EM) approach. In this section, we present the computation of the corpus likelihood, the two-step EM algorithm used to infer our parameters, and initialization of the EM algorithm.

1) Likelihood Computation: Our model has several levels, i.e., word level, sentence level, and document level. The latent variables are on two levels. Region r is at document level while aspect a and sentiment s are at sentence level. This multi-level structure poses challenges to the estimation of the log-likelihood. According to the generative process, we have the likelihood of the corpus D:

$$p(D;\Phi) = \prod_{d}^{D} p(u_d) \sum_{r}^{R} p(r|u_d) p(l_d, \mathbf{w}_d | r, u_d) \tag{4}$$

$$p(l_d, \mathbf{w}_d | r, u_d) = p(c_{l_d}|u_d) p(l_d|r, c_{l_d}) \prod_{i}^{M} p(\mathbf{w}_{d,i} | c_{l_d}, r, u_d, l_d) \tag{5}$$

$$p(\mathbf{w}_{d,i}|c_{l_d}, r, u_d, l_d) = \sum_{a,s} p(a|c_{l_d}, u_d) p(s|a, l_d) \prod_{j}^{N} p(w_{d,i,j}|a, s, r) \tag{6}$$

In Equation (4), Φ is the set of parameters in the model, i.e., $p(r|u)$, $p(a|c,u)$,$p(l|r)$,$p(s|a,l)$,$p(w|a,s)$, $p(w|r)$,$\boldsymbol{\mu}_r$ and $\boldsymbol{\Sigma}_r$. Variables u_d,l_d,$\mathbf{w}_d$ are the user, location and words of review d, respectively. Variable $\mathbf{w}_{d,i}$ represents the set of words in sentence i of review d while $w_{d,i,j}$ is the j^{th} word in sentence i of review d. Taking logarithm of $p(D;\Phi)$ leads to a summation inside the logarithm:

$$L = \sum_{d} \log p(u_d) + \log \sum_{r} p(r|u_d) p(l_d, \mathbf{w}_d | r, u_d) \tag{7}$$

Since this likelihood cannot be estimated directly, we adopt Jessen's inequality to the log-likelihood, and estimate the lower bound of the likelihood and the parameters in an iterative manner.

2) Expectation-Maximization: Due to the aforementioned difficulty of computing log-likelihood directly, we apply Expectation-Maximization (EM) algorithm to estimate the model parameters.

In **E-step**, we compute the expectation of latent variables given the observed data. By applying Jessen's inequality to Equation (7), we get the lower bound of the likelihood as:

$$L_{LB} = \sum_{d} \log p(u_d) + \sum_{d,r} p(r|d)(\log p(r|u_d) + \log p(l_d, \mathbf{w}_d | r, u_d)) \tag{8}$$

As shown in Equation (8), we need to estimate $p(r|d)$ to compute the full likelihood. We apply Bayes rule, and obtain the update function of the posterior distribution as

$$p(r|d) = \frac{p(r,d)}{\sum_r p(r,d)} \tag{9}$$

$$p(r,d) = p(u_d) p(r|u_d) p(l_d, \mathbf{w}_d | r, u_d) \tag{10}$$

In Equation (10), $p(l_d, \mathbf{w}_d|r, u_d)$ is computed by Equation (5), and $p(u_d)$ appears both in the numerator and the denominator, and thus is not necessary to estimate.

In **M-step**, by maximizing the lower bound of likelihood, we can obtain the update function of parameters at document level that are related to topical region r as below.

$$p(r|u) = \frac{\sum_{d \in D_u} p(r|d)}{\sum_r \sum_{d \in D_u} p(r|d)} \tag{11}$$

However, we cannot obtain a close form solution for $\boldsymbol{\mu}_r$ and $\boldsymbol{\Sigma}_r$ due to the normalization term. We adopt a gradient method to obtain the update value of $\boldsymbol{\mu}_r$ and $\boldsymbol{\Sigma}_r$ in M-step. Specifically, we use the BFGS quasi-Newton method [8], [11]. In the gradient method, we compute the gradient of $\boldsymbol{\mu}_r$ and $\boldsymbol{\Sigma}_r$ as follows:

$$\frac{\partial L_{LB}}{\partial \boldsymbol{\mu}_r} = \sum_d p(r|d) \boldsymbol{\Sigma}_r^{-1} \left(\frac{\sum_{l'} q(l')(\boldsymbol{cd}_{l'} - \boldsymbol{\mu}_r)}{\sum_{l'} q(l')} - (\boldsymbol{cd}_{l_d} - \boldsymbol{\mu}_r) \right) \tag{12}$$

$$\frac{\partial L_{LB}}{\partial \boldsymbol{\Sigma}_r} = \sum_d p(r|d) \left(\frac{\sum_{l'} q(l') g(l', r)}{\sum_{l'} q(l')} - g(l_d, r) \right), \tag{13}$$

where $q(l') = p(l'|c_l) p(l'|r)$ and $\boldsymbol{cd}_l$ denotes the coordinates of POI l. The function $g(l,r)$ in Equation (13) is the gradient of the Gaussian distribution for region r w.r.t. $\boldsymbol{\Sigma}_r$ at point l.

Since sentiment and aspect are at the sentence level, we cannot compute $\log p(l_d, \mathbf{w}_d|r, u_d)$ in Equation (8) using $p(r|d)$. Thus, we propose a second level of EM iterations. Specifically, we introduce a new latent variable to estimate parameters related to aspect and sentiment. Specifically, we use ϕ_{a,s,r,d_i} to identify the probability that the i^{th} sentence in a review d from region r is assigned with aspect a and sentiment s. we use ϕ_{a,s,r,d_i} and $p(r|d)$ to compute the update function of $p(a|c,u)$, $p(s|l,a)$, $p(w|a,s)$, and $p(w|r)$.

Denote by $n(w,d_i)$ the number of occurrences of word w in sentence i of review d. We estimate ϕ_{a,s,r,d_i} as:

$$\phi_{a,s,r,d_i} = \frac{p(a,s,r,d_i)}{\sum_{a,s} p(a,s,r,d_i)} \tag{14}$$

$$p(a,s,r,d_i) = p(u_d) p(r|u_d) p(c_{l_d}|u_d, r) p(l_d|r, c_{l_d}) p(a|c_{l_d}, u_d) p(s|a, l_d) \prod_{w} p(w|a,s,r)^{n(w,d_i)} \tag{15}$$

By maximizing the lower bound of the likelihood, we obtain the update function of the rest parameters:

$$p(a|u,c) = \frac{\sum_{d \in D_u} \sum_r p(r|d) \sum_i \sum_s \phi_{a,s,r,d_i}}{\sum_{a'} \sum_{d \in D_u} \sum_r p(r|d) \sum_i \sum_s \phi_{a',s,r,d_i}} \tag{16}$$

$$p(s|l,a) = \frac{\sum_{d \in D_l} \sum_r p(r|d) \sum_i \sum_s \phi_{a,s,r,d_i}}{\sum_{s'} \sum_{d \in D_l} \sum_r p(r|d) \sum_i \sum_s \phi_{a,s',r,d_i}} \tag{17}$$

$$p(w|s,a) = \frac{\sum_d \sum_r p(r|d) \sum_i \phi_{a,s,r,d_i} n(w,d_i)}{\sum_{w'} \sum_d \sum_r p(r|d) \sum_i \phi_{a,s,r,d_i} n(w',d_i)} \tag{18}$$

$$p(w|r) = \frac{\sum_d p(r|d) \sum_i \sum_a \sum_s \phi_{a,s,r,d_i} n(w,d_i)}{\sum_{w'} \sum_d p(r|d) \sum_i \sum_a \sum_s \phi_{a,s,r,d_i} n(w',d_i)}, \tag{19}$$

where D_u is the set of reviews written by user u and D_l is the set of reviews for POI l.

3) Initialization of EM Algorithm: EM algorithm can only guarantee to find a local optima. Different initializations may lead to different results. In this section, we present our methods for initializing the assignment of aspect, sentiment and region.

Aspect is extracted from sentence level in our model. We initialize the aspect by a clustering process on sentences. Each sentence is represented as a vector of words. Given the number of aspects, we use K-means clustering algorithm to assign each sentence an aspect. We then initialize $p(w|a)$ by the probability that word w appears in sentences carrying aspect a.

Sentiment has 3 possible values in this paper: positive, negative and neutral. In order to know the polarity of each sentiment, we need some prior knowledge. We use the same predefined set of sentiment seed words as in Jo's proposal [7]. Moreover, we apply a syntactic parser to extract negation of the sentiment words such as "not good" and use a special word "not_good" to represent the phrase "not good" in our vocabulary. For each word in the seed word set, we assign a probability ($p(w|s)$) of 1 to its polarity and 0 to the other two polarities. For words not in the seed word set, we assign an equal probability for each polarity. We then use $p(w|a)p(w|s)$ to approximate $p(w|a,s)$.

Region is initialized by a K-means clustering algorithm based on the coordinates (latitude and longitude). The clustering algorithm partitions POIs to different regions. Then for each region r, we compute $\boldsymbol{\mu}_r$ and $\boldsymbol{\Sigma}_r$ using a regression over the POIs in the region. We compute $p(w|r)$ by the distribution of words in the reviews for POIs in region r and $p(r|u)$ by the portion of reviews that user u writes in region r.

For other parameters: $p(a|c,u)$ and $p(s|a,l)$, we initialize them by using the assignment of aspect and sentiment to a sentence (We assign sentiment to a sentence by voting from sentiment seed words extracted from the sentence). Specifically, $p(a|c,u)$ is proportional to the number of sentences that are assigned to a and that belong to a review written by u from category c; $p(s|a,l)$ is proportional to the number of sentences that belong to location l and are assigned to sentiment s and aspect a at the same time.

4) Efficiency Analysis: Let the number of sentiment be 3 and we treat it as constant. In E-step, the computation of the expectation of latent variables in Equation (9) and the variables ϕ_{a,s,r,d_i} in Equation (19) needs $O(|D|MNRA) = O(WRA)$, where W is the number of words in the reviews of all users' in training set, R is the number of regions and A is the number of aspects. In M-step, the cost for updating Equation (16) to (19) is $O(UA + LA + VA + VR)$, where U, L, V are the number of users, POIs and unique words, respectively. To update $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, we perform a quasi-Newton method. Since each $\boldsymbol{\mu}_r$ and $\boldsymbol{\Sigma}_r$ are two dimensional vector and 2×2 matrix, respectively. The computation cost of matrix operation can be treated as constant. Let D be the number of reviews, the cost of computing gradient in Equation (12) and (13) is $D + L$. Therefore, the complexity of quasi-Newton is $O(I_q R(D+L))$, where I_q is the number of iterations of quasi-Newton. In summary, the total complexity of the learning algorithm with I iterations is $O(I(WRA+I_qR(D+L)+UA+LA+VA+VR))$. Since $WRA \gg (UA + LA + VA + VR)$, we simplify the cost as $O(I(WRA + I_qR(D + L)))$. We can parallelize the computation of both E-step and M-step. In E-step, since the computation of $p(r|d)$ on each document is independent to others, we can compute $p(r|d)$ of each document in parallel. In M-step, the update of Equation (16) to (19) and the quasi-Newton iterations can also be parallelized in the similar way as $p(r|d)$. Therefore, the algorithm can be fully parallelized.

IV. Applications

We present three applications of our model, namely POI recommendation, user recommendation, and aspect satisfaction analysis in regions. In POI recommendation, we provide a way to explain the reason of recommending a POI and propose an efficient online recommendation algorithm.

A. POI recommendation

We apply our model to two POI recommendation tasks and propose an efficient online recommendation algorithm. The two recommendation tasks are *All-Category Recommendation* and *Single-Category Recommendation*.

1) All-Category Recommendation: All-Category Recommendation is a task of generating a rank list of POIs in any category given a set of POIs and a user. The aforementioned proposals are all for all-category recommendation. We calculate the probability of $p(l, s_+|u)$, i.e., the probability of user u visits POI l with positive sentiment, to score l for u as shown in Equation (20).

$$p(l, s_+|u) = \sum_r p(r|u)p(c_l|u)p(l|r, c_l) \sum_a p(a|u, c_l)p(s_+|a, l) \tag{20}$$

According to Equation (20), we make the recommendation by considering the matching between user preferences (i.e., $p(r|u)$, $p(c_l|u)$ and $p(a|u, c_l)$) and the attributes of the POI (i.e., $p(s_+|a, l)$ and $p(l|r, c_l)$).

This recommendation model enables us to explain why we recommend a POI to a user. We consider two factors: aspect and region. First, we rank the aspects by $p(s_+|a, l)p(a|u, c_l)$ to reveal which aspects match the user's preferences. Second, we rank the regions by $p(r|u)p(l|r)$ to reveal which regions contribute more to the recommendation. Finally, we choose top several aspects and regions for explanation.

2) Single-Category Recommendation: Single-Category Recommendation aims at ranking POIs given a user and a specific category (e.g., restaurants). It is a typical scenario for POI recommendation although it has not been covered in previous work. We compute $p(l, s_+|u, c)$ as shown in Equation (21). Compared to all-category recommendation, we fix the category i.e., remove $p(c|u)$ from Equation (20). All locations that are not in c will not be considered in this scenario.

$$p(l, s_+|u, c) = \sum_r p(r|u)p(l|r, c) \sum_a p(a|u, c)p(s_+|a, l) \tag{21}$$

We can also offer explanation for the single-category recommendation by following similar method as we employ for the all-category recommendation.

3) Efficient algorithm for Top-N Online Recommendation: Time efficiency is an essential part of online recommendation. A straightforward method of making recommendation is to compute the recommendation score as Equation (20) or Equation (21). This method requires traversing all the regions which is highly time consuming. Another choice is the threshold algorithm [4] that may save the computation for some POIs. However, in our applications, the number of attributes (i.e., regions and aspects) is large, and thus it is expensive to compute the recommendation score even for a single POI. The threshold algorithm cannot help with this, either. We propose an optimized top-N items recommendation algorithm that significantly reduces the time cost. As to be shown in the experiment, our algorithm is faster than the threshold algorithm in the top-N POI recommendation using our model. Our algorithm can be applied to all or single-category POI recommendation. We use all-category POI recommendation (Equation (20)) as an example to explain the algorithm.

Our algorithm is based on two observations: 1) A user only prefers a small number of regions; and 2) POIs in the center of the region are more likely to be recommended. These two observations indicate that only when a user prefers a region and the POI is near the center of the region, will the score $p(r|u)p(l|r,c_l)$ contribute significantly to the recommendation score. Therefore, after we have computed the most possible regions for a POI, it may not be necessary to compute the remaining regions. We design a branch and bound algorithm as shown in Algorithm 1 to prune the search space of the regions. Our algorithm contains two steps: *initialization* and *pruning*.

In the *initialization* step (line 2), we find N candidate POIs that are potentially good for recommendation. Specifically, we pick top K regions which cover most of the user's regional preferences (i.e., $\sum_{i=1}^{K} p(r_i|u) > 0.9$) with smallest K (line 21). If K is larger than N, we pick at most N regions to ensure that we can select at least one candidate from each region. In each of the top K region, we choose top $\lceil \frac{N}{K} \rceil$ POIs w.r.t. $p(l|r)$ as candidates.

In the *pruning* step (line 9-10), we check whether we can avoid traversing unnecessary regions for each POI. We traverse the regions according to the descending order of $p(l|r,c_l)$ for POI l. Suppose we have traversed regions $\{r_1,...,r_{i-1}\}$. The partial score we have computed for the traversed regions is

$$PScore = \sum_{j=1}^{i-1} p(r_j|u)p(l|r_j,c_l).$$

When we explore the i-th region, we compute the upper bound of recommendation score for the POI as:

$$Bound^{(i)}(l) = PScore + (1 - \sum_{j=1}^{i-1} p(r_j|u))p(l|r_i,c_l). \quad (22)$$

Because we check the regions in the descending order of $p(l|r,c_l)$, the actual value of $p(l|r,c_l)$ for the remaining regions should be less than the one for the current region, i.e., $p(l|r_i,c_l)$. Therefore, we have a partial recommendation score for the rest of the regions, which is at most

$$(1 - \sum_{j=1}^{i-1} p(r_j|u))p(l|r_i,c_l),$$

where $1 - \sum_{j=1}^{i-1} p(r_j|u)$ is the portion of user preferences for the rest regions. The upper bound of $\sum_r p(l|r,c)p(r|u)$ for all regions is $PScore + (1 - \sum_{r=r_1}^{r_{i-1}} p(r|u))p(l|r_i,c_l)$. Since $\sum_a p(a|u,c)p(s_+|a,l) \leq 1$, Finally, we obtain the upper bound of the recommendation score in Equation (20) for the POI by setting $\sum_a p(a|u,c)p(s_+|a,l) = 1$, which results in Equation (22).

If the upper bound is smaller than the N^{th} candidate (Line 9), we skip the current POI (no need to check the remaining regions). Otherwise, we continue to check the remaining regions. If all regions are examined for the POI and the POI is not pruned by the aforementioned upper bound, we compute the full score of the POI to compare with the N^{th} smallest candidate (line 12). We remove the N^{th} candidate in the list and insert the POI to the list if the full score is larger than the N^{th} candidate (line 13-15). To maintain the top-N candidate list, we use a binary min-heap.

Algorithm 1 POI Recommendation

```
1:  function REC(u, N)
2:      H ← InitialCandidates(N)
3:      for l ∈ L and l ∉ H do
4:          PartS ← 0, PartRPro ← 0, Skip ← false
5:          while there exists r not examined for l do
6:              r ← NextRegion()
7:              PartS ← PartS + p(r|u)p(l|r, c_l)
8:              PartRPro ← PartRPro + p(r|u)
9:              if PartS + (1 − PartRPro) * p(l|r, c_l) <
    H.Top() then
10:                 Skip ← true, break
11:         if Skip = false then
12:             S ← PartS * p(c_l|u) Σ_a p(s_+|a, l)p(a|u, c_l)
13:             if S > H.Top() then
14:                 H.DeleteTop()
15:                 H.Insert(< l, S >)
16:     Result ← Sort H by Score S
17:     return Result

18: function INITIALCANDIDATES(N)
19:     H ← ∅
20:     r_1, ..., r_R ← Sort the regions by p(r|u)
21:     Pick top K regions satisfies: K =
    min({k| Σ_{i=1}^{k} p(r_i|u) > 0.9}, N)
22:     From r_1 to R_K, Insert top ⌈N/K⌉ POIs ordered by p(l|r)
    to H until H contains N POIs
23:     return H
```

B. User Recommendation

We can also apply our model to recommend users for a POI. Predicting which users may favor a given POI is useful when the owner of the POI wants to target at or advertise to some of the users. Given a POI l, we compute the probability

$p(u, s_+|l)$ of user u favoring POI l by considering both topical-region and topical-aspect preferences of users as follows:

$$p(u, s_+|l) = \frac{p(u, s_+, l)}{\sum_{u,s} p(u, s, l)} \tag{23}$$

$$\begin{aligned} p(u, s, l) =& p(u)p(c_l|u) \sum_r p(r|u)p(l|r, c_l) \\ & \sum_a p(a|u, c_l)p(s|a, l), \end{aligned} \tag{24}$$

where prior $p(u)$ is calculated using the user's review history:

$$p(u) = \frac{\#\ of\ reviews\ u\ wrote}{\#\ of\ all\ reviews}.$$

Since the last two summations are the same as those in POI recommendation, Algorithm 1 can also be used to speed up the user recommendation.

C. Aspect Satisfaction Analysis in Regions

Discovering which aspect is satisfied or not by users in each region is useful when 1) someone wants to set up a new business or make strategies to attract more customers, or 2) policy makers make urban planning. For example, most of the restaurants in a region of a city may be complained for the long waiting time. By knowing the dissatisfaction of this aspect, a restaurant may think how to achieve competitive advantage over other restaurants in the region. We can infer the aspect satisfaction in each region based on our model. Specifically, we compute the aspect distribution of each sentiment s, category c and region r as

$$p(a|s, c, r) = \frac{\sum_{u,l} p(u)p(r|u)p(c|u)p(a|c, u)p(l|r, c)p(s|a, l)}{\sum_{a,u,l} p(u)p(r|u)p(c|u)p(a|c, u)p(l|r, c)p(s|a, l)} \tag{25}$$

This probability shows which aspect is most probably liked/disliked in POIs from category c and region r.

V. Experimental Studies

We conduct several experiments to evaluate the performance of our model by comparing with the state-of-the-art techniques. First, we present the experimental setup in Section V-A and the model selection strategy in Section V-B; Then, we apply our model to the three applications. We show the accuracy and efficiency of our recommendation algorithm in POI recommendation in Section V-C and explore the reason of making a POI recommendation in Section V-D. After that, we show the accuracy of user recommendation in Section V-E. Finally, we analyze the aspect satisfaction in regions in Section V-F and discuss the quality of topical regions in Section V-G.

A. Experimental Setup

We collect data from two different cities in Yelp, which hosts hundreds of thousands of user reviews and ratings for POIs. One of our datasets is from Yelp's Challenge Dataset[1], which contains 11,537 POIs and 43,873 users from Phoenix at the time of data collection. We remove users who wrote fewer than 2 reviews and POIs without any reviews after filtering the users. This preprocessing results in a dataset containing 11,359 POIs and 21,908 users. We crawl the other dataset from Yelp Singapore, and remove users and POIs without any reviews. The resulting Singapore dataset contains 8,846 POIs and 1,654 users. The statistics of the two datasets is shown in Table II. For each dataset, we hold out the recent 10% reviews of each user for tuning and 10% for testing, respectively, and use the remaining 80% data as training set. In the tuning and test data, we generate the ground truth, i.e., whether a user likes the POI or not, by checking how the user rates to that POI. When the user rates more than 3.5 stars, we consider that she likes the POI and put the user-POI pair into the set of ground truth.

[1] http://www.yelp.com.sg/dataset_challenge/

TABLE II: Statistics of the two datasets

	Phoenix	Singapore
#POIs	11,359	8,846
#Users	21,908	1,654
#Reviews	215,837	20,248
#Reviews per user	4.92	12.40
#Reviews per POI	18.71	2.36

We run all experiments on a machine with Intel Xeon E5-2680 (2.8 Ghz) 10-cores CPU and 64GB memory. We deploy the training program on 8 cores of the CPU and train the SAR models on Singapore and Phoenix datasets with the same settings as in Section V-B. The training time of Singapore data is 33.74 minutes and the Phoenix data is 965.61 minutes.

B. Model Selection

In our model, we have two free parameters: number of aspects A and number of topical regions R, and a tunable parameter λ. To find a proper value of A and R, we use Bayesian Information Criterion (BIC) which is usually used for model selection. BIC is defined as: $BIC = -2L + Kln(N)$, where L is the likelihood of the model, K is the number of parameters and N is the scale of the corpus. In our scenario, K is $A \times R$, while N is the number of review sentences. We set the default value of parameter λ to 0.6, and choose A,R which achieve lowest BIC in the tuning set of each dataset. In the Phoenix data, the lowest BIC appears when $R = 80$ and $A = 30$. In the Singapore data, smaller number of aspects and regions are preferred because Singapore is smaller than Phoenix. We set $R = 40$ and $A = 20$ for Singapore data.

C. POI recommendation

First, we introduce several existing methods for competition and the evaluation metrics. Then, we compare with the state-of-the-art baselines for both all-category and single-category recommendation. Finally, we evaluate the efficiency of our recommendation algorithm.

1) Recommendation Methods: We compare our model to five POI recommendation techniques.

- **CF**: User-based collaborative filtering model.
- **GCF**: A collaborative filtering model incorporating geographical influence [22].
- **W3**: A topic model with personalized regions [27].
- **STM**: A topic model with global regions [6].
- **EFM**: A matrix factorization based explicit factor model which extracts aspects and sentiment from

reviews, and models the relation among user, item, aspect and sentiment for recommendation [29].

- **GEFM**: We multiply the EFM rating score with a geographical score given by $exp\{-dist(u,l)\}$. Function $dist(u,l)$ is the average distance from POI l to POIs that the user has visited.

2) Evaluation Metrics: Evaluating a recommended list has two ways: one of them is how many true results are hit by the list and the other is how similar the resulting rank and the ground truth rank are. Therefore, we use two kinds of metrics to measure the performance of our model and the peers. These metrics are: 1) the precision and recall for the top N items, namely *Precision@N* and *Recall@N*, respectively. We investigate $N = 5$ and $N = 10$ because the top few results are most impressive to users. 2) *Mean Average Precision* (MAP) which is used to show the correctness of a rank list according to the position of true results in the list. If the true results are ranked high in the list, the list is probably a good recommendation result.

3) All-Category POI Recommendation: The result is shown in Figure 2. All results reported in this section pass t-test with p-value< 0.01, which means the improvements are significant. Our SAR model outperforms the best peer by 33% , 34% and 61% in terms of Precision@10, Recall@10 and MAP, respectively on the Phoenix dataset, while 59% , 90% and 62% in terms of Precision@10, Recall@10 and MAP, respectively on the Singapore dataset.

Among the baseline methods, CF and EFM do not consider geographical information, which limits the performance of these two methods. Compared to CF, EFM performs better because it explores the user preferences on aspect level. GEFM performs the best among the baselines, but still worse than SAR because it does not model the interdependencies. W3 has lower performance than GCF model in some cases because it learns small personalized regions for users who have limited number of visiting records, which leads to overfitting. STM performs worst because it estimates the probability $p(l|r)$ by the probability density function of Gaussian distribution without any normalization. The Gaussian distribution in STM overwhelms the other probabilities (i.e., the preferences of the user). GCF incorporates the graphical information into the model of CF. However, without considering the content of the reviews, GCF cannot reveal user's preferences on aspect level.

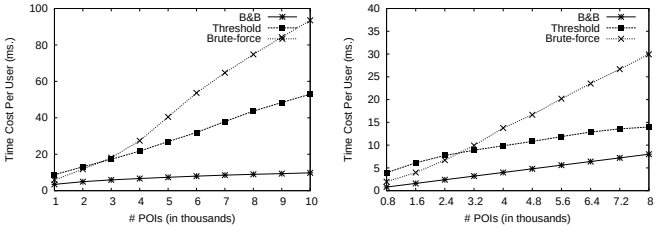
Our SAR model discovers user's latent interest on several factors: aspect, sentiment, category, and region. Benefiting from the user preference analysis on topical-aspects and topical-regions, SAR model outperforms these methods.

4) Single-Category POI recommendation: The peers are developed for all-category recommendation. To apply them to this task, we consider two methods. The first one is to pick the top N results that belong to the given category from the all-category recommendation results. The second one is to divide the visit history of each user by categories and learn the models on data from each category separately. The second method suffers from the problem of sparser data. Therefore, we adopt the first method in this experiment.

The result is reported in Figure 3. Our SAR model outperforms the best peer by 36% , 36% and 42% in terms of Precision@10, Recall@10 and MAP, respectively on the Phoenix dataset, while 52% , 58% and 80% in terms of Precision@10, Recall@10 and MAP, respectively on the Singapore dataset. The reason is that our model is able to discover the relation between category and aspect by modeling the user preferences to topical-aspects on each category, i.e., $p(a|c,u)$.

5) Recommendation Efficiency: We evaluate the efficiency of our optimized POI recommendation algorithm on both of the datasets by comparing with two algorithms. One of them is the brute-force algorithm which computes Equation (20) and uses a partial sorting method to find the top-N result. The other one is the threshold algorithm (TA) [4] that sorts $p(l|r,c_l)$ for each region and accesses the POIs on each sorted list in parallel. Since the scoring function is a monotonic increasing function, we follow the threshold algorithm to find the top-N results. We do not sort by aspects over POIs because the differences among POIs on an aspect ($p(s|a,l)$) is much smaller than those on a region. Since $\sum_a p(a|u,c)p(s|a,l)$ and $p(c|u)$ in Equation (20) are no larger than 1, we compute the threshold as: $T = \sum_r p(r|u) \max_{l'} p(l'|r,c_{l'})$.

In this experiment, we retrieve top 100 results for each user. To investigate the running time on POIs of different scales, we randomly select subsets with different sizes from the two datasets. Specifically, we get 10 different subsets with size from 1000 to 10,000 POIs in Phoenix data while another 10 subsets with size from 800 to 8000 POIs in Singapore data. The time of recommending top 100 POIs to a single user is computed by averaging over all users. The result is reported in Figure 4.

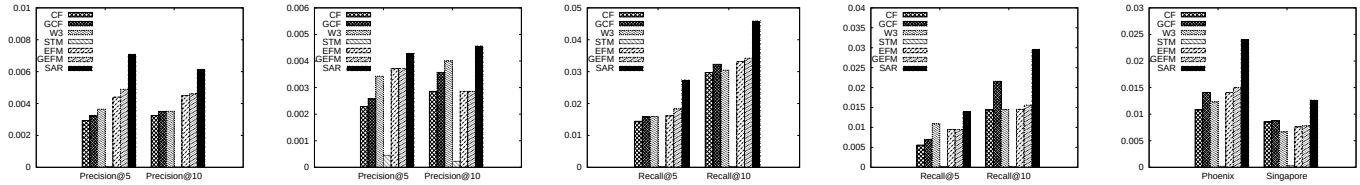


(a) Phoenix (b) Singapore

Fig. 4: All-Category POI Recommendation Time Consumption

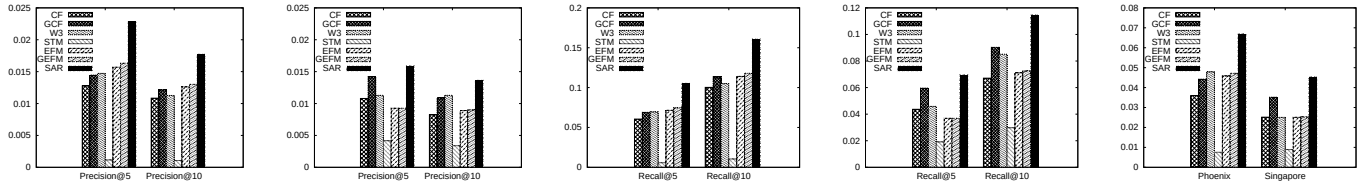
Our optimized algorithm, namely "B&B" in Figure 4, always takes the least amount of time on both datasets for different number of POIs. Compared to TA, we achieve 4 times faster in the Phoenix data and 2 times faster in the Singapore data, because TA needs to update the threshold for each sorted access on any sorted list. This extra computation makes TA perform worse than the brute-force algorithm when the number of POIs is small. We give a more detailed comparison for the threshold algorithm and our methods as below.

In the threshold algorithm, we still need to access many POIs (around 40% of the POIs in Phoenix on average) in each sorted list for each region. However, our algorithm accesses very few regions for each POI to compute its partial score (less than 10 regions on average in Phoenix), and we only compute the full score for POIs that we access all the regions (less than 10% of the POIs in Phoenix). Suppose there are



(a) Pre@N - Phoenix (b) Pre@N - Singapore (c) Rec@N - Phoenix (d) Rec@N - Singapore (e) MAP in both datasets

Fig. 2: All-category POI recommendation



(a) Pre@N - Phoenix (b) Pre@N - Singapore (c) Rec@N - Phoenix (d) Rec@N - Singapore (e) MAP in both datasets

Fig. 3: Single-category POI recommendation

L POIs, R regions and A aspects and we compute the full score for $0.4L$ POIs in threshold algorithm and $0.1L$ POIs in our algorithm[2]. In the worst case, the threshold algorithm updates $0.4LR$ times of threshold. The total computation cost for threshold algorithm is $0.4L(A+2R)$. Our algorithm needs to compute the full score for $0.1L$ POIs as well as the partial score and the threshold for the rest POIs on 10 regions on average. The computation costs is $0.1L(A+R)+0.9L\times 2\times 10$. When $A = 30$ and $R = 80$ in the Phoenix data, the cost of threshold algorithm is $76L$ while the cost of our algorithm is $29L$. When the number of regions increases, the threshold algorithm has more sorted list to access, which makes it slower. Whereas in our algorithm, we still need to consider only a few regions (near a POI). Our algorithm is more suitable for the models that have a large number of regions.

D. Explanation of Recommendation

As discussed in Section V-C, the SAR model can tell why we recommend or not recommend a POI. To illustrate this, we randomly pick some examples from the test data. We explore aspects and regions respectively. To find out which aspects contribute most to the recommendation, we first show the top-5 favorite aspects of the selected user in the category of the POI according to $p(a|u,c)$. Then we report the top-5 good aspects of the POI according to $p(s_+|a,l)$. We manually give a name to aspect according to the word distribution. Table III shows three users and a recommended POI for each of them. *User 64* prefers the food and flavor aspects of a restaurant and *Paradise Dynasty*, a restaurant of Chinese food, has positive reviews on foods. *User 121* wants a good environment in a bar, and *Wala Wala Cafe Bar* has a good environment. *User 420* prefer a hotel with good facility and service and Marina Bay Sands provides good facility. This table shows that we recommend to users the POIs with aspects that match their preferences. This explains why our method makes a recommendation to a user.

[2]We estimate this average ratio on our datasets

TABLE III: User Aspect Preference & Positive Aspects of Recommended POIs

User	Preference		POI	Positive Aspect	
	Aspect	Prob		Aspect	Prob
64	menu	0.29	Paradise Dynasty (Restaurant)	general	0.98
	flavor	0.11		taste	0.96
	food	0.07		food	0.89
	quality	0.07		flavor	0.74
	service	0.07		menu	0.69
121	environment	0.36	Wala Wala Cafe Bar	quality	1
	general	0.11		location	1
	facility	0.10		taste	1
	traffic	0.10		flavor	0.97
	location	0.09		environment	0.87
420	facility	0.29	Marina Bay Sands (Hotel)	quality	1
	service	0.20		facility	0.84
	general	0.09		general	0.80
	traffic	0.07		environment	0.70
	quality	0.07		food	0.56

This is a very desirable feature for a recommendation system, although many existing recommendation methods cannot offer such explanation.

To explore the influence of regions, we draw the contour of the top 3 regions of *User 64* ranked by probability $p(r|u)$, and plot the top 5 recommended POIs to the user. We highlight *Paradise Dynasty* in blue color. Figure 5 shows the regions and recommended POIs. The 5 POIs are close to the center of *Region 36* or *Region 8*, two of the user's favorite regions, which is the geographical reason for recommending those POIs. Our recommendation algorithm tends to recommend POIs that are close to the center of the user's favorite regions.

E. User Recommendation

1) Peers and Metrics: W3 and STM, which are based on topic models, can be easily applied to user recommendation by multiplying the conditional probability $p(l|u)$ with user popularity $p(u)$. However, the existing CF (CF and GCF) and EFM

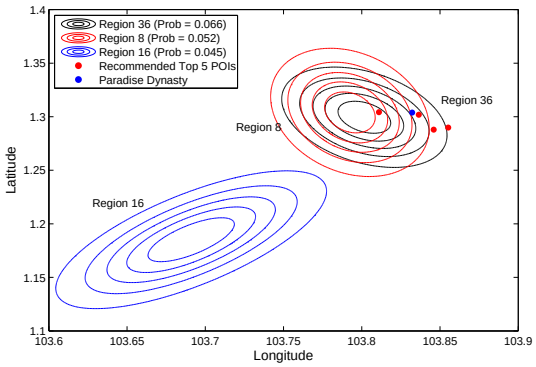


Fig. 5: The Region Preference of User 64 and the Recommended POIs

baselines are not designed for user recommendation, we simply reverse the rating matrix by treating users as items and items as users in CF methods. For GCF model, we compute the score in the geographic preference matrix by comparing the coordinates of the POI with each user's visit histories, and normalizing over all users: $Score_{geo}(l,u) = \frac{\prod_{l' \in L_u} Dist(l,l')}{\sum_{u'} \prod_{l' \in L_{u'}} Dist(l,l')}$. The set L_u is the visit history of user u. We then linearly combine the geographic preference score and the visit-based score as the original model does. For EFM and GEFM, we rank users by $R_{i,j}$ (i.e., the predicted rating of user i on item j) for each item. We use the same metrics as those in Section V-C to measure the performance of user recommendation.

2) Result: Figure 6 shows the comparison among our model and the peers. The results reported are significant with p-value< 0.01 in t-test. The SAR model outperforms the peers because we model the interaction among aspect, sentiment and region. The interaction identifies whether a user likes a region (i.e., $p(r|u)$) and an aspect given a category (i.e., $p(a|u,c)$). When performing recommendation, we check whether the given POI satisfies the preferences of users. The SAR model does not perform significantly better than the other methods in Singapore because the number of reviews per POI is very small in the data. Without enough reviews, one can hardly tell which aspect is good for the POI and which is bad. In contrast, the Phoenix POIs contain around 9 times the reviews than those in Singapore, which means SAR can take advantage of the aspect and sentiment analysis to improve recommendation results.

F. Aspect Satisfaction Analysis in Regions

We explore the aspect satisfaction in some regions in the Singapore dataset. The goal of this experiment is to show the satisfaction of aspects in the regions. We compute $p(a|s,c,r)$ described in Section IV-C to select most satisfied and dissatisfied aspects for analysis. We show the satisfaction on "service" of Beauty & Spas POIs in two regions in Table IV. The first region has the highest probability of having negative overall sentiment on the aspect "service", while the second one has the highest probability of having positive sentiment on "service". In different regions, the sentiment on different categories is different. As shown in the example, the service is negative in Region 1, but is positive in Region 2.

TABLE V: Representative Words for Regions

Region	Representative Words
Orchard	place, food, really, singapore, **store**, **shop**, **mall**, time, pretty, **shopping**, quite, **products**, japanese, little, **orchard**
Changi	food, **airport**, singapore, place, toast, really, time, **changi**, pretty, **terminal**, kaya, coffee, service, little, free
Sentosa	place, food, **sentosa**, singapore, **beach**, really, pretty, quite, time, day, **island**, little, chicken, ride, people

TABLE VI: Representative Words for Aspect and Sentiment

Aspect	Sentiment	Representative Words
waiting time	positive	time, little, long, shop, best, good, great, enjoy, come, people, place, food, like, worth, day, lunch, really, perfect, try, way
waiting time	negative	time, little, come, times, went, second, shop, wrong, like, food, place, came, getting, told, table, waiting, disappointed, eat, review, bad
service	positive	great, service, good, food, staff, place, friendly, excellent, restaurant, really, nice, location, love, pretty, like, selection, experience, little, prices, hotel
service	negative	service, order, food, minutes, place, wait, staff, time, times, bad, table, really, took, long, terrible, pretty, water, waitress, came, restaurant

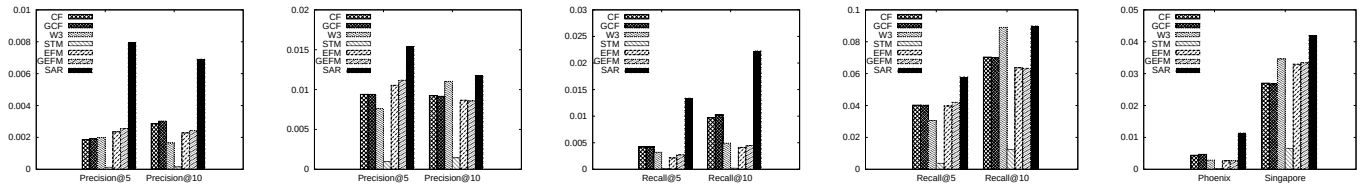
G. Discussion on Topical Regions

In Table V, we show the top 15 representative words with the highest probability (i.e., $p(w|r)$) for three regions in Singapore discovered by our model. The first region "Orchard" which is the largest shopping area in Singapore, contains words "store", "shop", "mall", "orchard", etc. The second region is near Changi Airport, so that words like "airport", "changi", "terminal", etc. are often used in that region. The third one contains words like "sentosa", "beach", "island", etc. These words are about Sentosa, which is the most famous island in Singapore. The word distribution in the three regions are characterized by spatial information.

We choose two aspects with positive and negative sentiments, and show top 20 words with highest $p(w|a,s)$ for each aspect and sentiment pair in Table VI. Different from region words, aspect words are like "service", "staff", "waiting", etc. and sentiment words are like "good", "best", "bad", etc. These words describe the aspects of the POIs and the sentiment to the aspects, while region words are more likely to be location words. As shown in Figure 1, aspects are constrained by categories while regions are constrained by coordinates.

VI. Conclusion

We present a novel generative model that considers aspect, sentiment and region. Motivated by user's opinions in POI reviews, we propose a way to make use of the opinions to enhance user preference modeling. Our model combines opinion mining and geographical modeling which has not been considered in the existing works. Both of the two aspects are considered to be important, and we show the effectiveness of the unified model by experiments on several applications, such as POI recommendation and user recommendation. We propose an online algorithm and show it is efficient in top-N recommendation using our model. We also explore the satisfaction of aspects in regions.



(a) Pre@N - Phoenix (b) Pre@N - Singapore (c) Rec@N - Phoenix (d) Rec@N - Singapore (e) MAP in both datasets

Fig. 6: User Recommendation

TABLE IV: Aspect Satisfaction in Two Regions

Region	Category	Sentiment	Aspect	Probability	Representative Review
1	Beauty & Spas	negative	service	0.39	I had an oily scalp...decided to try out Yun Nam Haircare... I immediately disliked how they automatically preyed on the chinks in our armor by pin pointing the problems that we had, ...
2	Beauty & Spas	positive	service	0.32	Nail Bar @ Cluny is a small nail salon ... I like it that it is clean and the nail technicians are professional and friendly...

ACKNOWLEDGEMENT

This work is supported in part by a grant awarded by a Singapore MOE AcRF Tier 2 Grant (ARC30/12), and a grant awarded by Microsoft Research Asia. Kenny Q. Zhu has been partially supported by NSFC Grants 61033002, 61100050 and 61373031, and Google Faculty Research Award.

REFERENCES

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

[2] C. Cheng, H. Yang, I. King, and M. R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, 2012.

[3] J. Eisenstein, B. O'Connor, N. A. Smith, and E. P. Xing. A latent variable model for geographic lexical variation. In *EMNLP*, pages 1277–1287, 2010.

[4] R. Fagin, A. Lotem, and M. Naor. Optimal aggregation algorithms for middleware. In *PODS*, pages 102–113, 2001.

[5] L. Hong, A. Ahmed, S. Gurumurthy, A. J. Smola, and K. Tsioutsiouliklis. Discovering geographical topics in the twitter stream. In *WWW*, pages 769–778, 2012.

[6] B. Hu and M. Ester. Spatial topic modeling in online social media for location recommendation. In *RecSys*, pages 25–32, 2013.

[7] Y. Jo and A. H. Oh. Aspect and sentiment unification model for online review analysis. In *WSDM*, pages 815–824, 2011.

[8] T. Kurashima, T. Iwata, T. Hoshide, N. Takaya, and K. Fujimura. Geo topic model: Joint modeling of user's activity area and interests for location recommendation. In *WSDM*, pages 375–384, 2013.

[9] J. J. Levandoski, M. Sarwat, A. Eldawy, and M. F. Mokbel. Lars: A location-aware recommender system. In *ICDE*, pages 450–461, 2012.

[10] B. Liu, Y. Fu, Z. Yao, and H. Xiong. Learning geographical preferences for point-of-interest recommendation. In *KDD*, pages 1043–1051, 2013.

[11] D. C. Liu and J. Nocedal. On the limited memory bfgs method for large scale optimization. *Math. Program.*, 45(3):503–528, Dec. 1989.

[12] Y. Lu, M. Castellanos, U. Dayal, and C. Zhai. Automatic construction of a context-aware sentiment lexicon: An optimization approach. In *WWW*, pages 347–356, 2011.

[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] Q. Mei, C. Liu, H. Su, and C. Zhai. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *WWW*, pages 533–542, 2006.

[15] S. Moghaddam and M. Ester. Ilda: Interdependent lda model for learning latent aspects and their ratings from online product reviews. In *SIGIR*, pages 665–674, 2011.

[16] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: Real-time event detection by social sensors. In *WWW*, pages 851–860, 2010.

[17] S. Sizov. Geofolk: Latent spatial semantics in web 2.0 social media. In *WSDM*, pages 281–290, 2010.

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

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

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

[21] D. Yang, D. Zhang, Z. Yu, and Z. Wang. A sentiment-enhanced personalized location recommendation system. In *HT*, pages 119–128, 2013.

[22] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*, pages 325–334, 2011.

[23] H. Yin, B. Cui, L. Chen, Z. Hu, and Z. Huang. A temporal context-aware model for user behavior modeling in social media systems. In *SIGMOD*, pages 1543–1554, 2014.

[24] H. Yin, B. Cui, L. Chen, Z. Hu, and C. Zhang. Modeling location-based user rating profiles for personalized recommendation. *TKDD*, Jan. 2014.

[25] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang. Geographical topic discovery and comparison. In *WWW*, WWW '11, pages 247–256, 2011.

[26] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann. Time-aware point-of-interest recommendation. In *SIGIR*, pages 363–372, 2013.

[27] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann. Who, where, when and what: Discover spatio-temporal topics for twitter users. In *SIGKDD*, pages 605–613, 2013.

[28] Q. Yuan, G. Cong, and A. Sun. Graph-based point-of-interest recommendation with geographical and temporal influences. In *CIKM*, pages 659–668, 2014.

[29] Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu, and S. Ma. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *SIGIR*, pages 83–92, 2014.

[30] Y. Zhang, H. Zhang, M. Zhang, Y. Liu, and S. Ma. Do users rate or review?: Boost phrase-level sentiment labeling with review-level sentiment classification. In *SIGIR*, pages 1027–1030, 2014.