



Modeling location-based social network data with area attraction and neighborhood competition

Thanh-Nam Doan¹  · Ee-Peng Lim¹

Received: 30 September 2017 / Accepted: 3 September 2018 / Published online: 18 September 2018
© The Author(s) 2018

Abstract

Modeling user check-in behavior helps us gain useful insights about venues as well as the users visiting them. These insights are important in urban planning and recommender system applications. Since check-in behavior is the result of multiple factors, this paper focuses on studying two venue related factors, namely, *area attraction* and *neighborhood competition*. The former refers to the ability of a spatial area covering multiple venues to collectively attract check-ins from users, while the latter represents the extent to which a venue can compete with other venues in the same area for check-ins. We first embark on empirical studies to ascertain the two factors using three datasets gathered from users and venues of three major cities, Singapore, Jakarta and New York City. We then propose the visitation by area attractiveness and neighborhood competition (VAN) model incorporating area attraction and neighborhood competition factors. Our VAN model is also extended to incorporate *social homophily* so as to further enhance its modeling power. We evaluate VAN model using real world datasets against various state-of-the-art baselines. The results show that VAN model outperforms the baselines in check-in prediction task and its performance is robust under different parameter settings.

Keywords Location-based social network · Check-in prediction · User behavior · Area attraction · Neighborhood competition · Matrix factorization

Responsible editor: Hanghang Tong.

This research is supported by the National Research Foundation, Prime Ministers Office, Singapore under its International Research Centres in Singapore Funding Initiative.

✉ Thanh-Nam Doan
tndoan.2012@smu.edu.sg

Ee-Peng Lim
eplim@smu.edu.sg

¹ School of Information Systems, Singapore Management University, Singapore, Singapore

1 Introduction

The popularity of smartphones and wearable devices in recent years has helped to create new location based social networking (LBSN) applications for users to publish their visits to different venues, also known as check-ins. For example, in 2017, Foursquare is used by 50 millions users each month and it covers more than 65 million venues around the world. These users have generated 8 billion check-ins worldwide.¹ By analyzing these check-in data, one may derive useful insights for urban planning (Smarzaro et al. 2017a, b; Quercia and Saez 2014), business recommendation (Lin et al. 2016a, b; Georgiev et al. 2014; Zhao et al. 2017), and other applications (Yuan et al. 2012; Backstrom et al. 2010; Isaacman et al. 2012; De Nadai et al. 2016; Yu et al. 2016).

Previous works on LBSN data have shown that users prefer to visit venues near their home locations (Doan et al. 2015b; Cho et al. 2011; Song et al. 2010). This is also known as the *distance effect*. It underscores the importance of home location of users when analyzing their movement. Other than the distance effect which is user specific, there are other venue factors that have not yet been well studied and modeled. In particular, *distance effect* is limited in explaining why some venues may still attract check-ins from users far away. To address this limitation, Li et al. (2012) introduced influence scope for measuring the attractiveness of a venue to its followers. In this paper, instead of examining attractiveness at the venue level, we model attractiveness at the area level. There are three significant advantages of doing so. Firstly, it reduces the number of parameters in modeling which in turn reduces the learning time. Secondly, area level check-in data will be less sparse for modeling area attraction. Finally, we believe that the area a venue belongs to has a major influence over its ability to attract users. This will be verified in our empirical analysis.

Research objectives In this paper, we introduce *area attraction* and *neighborhood competition* as two new venue factors for analyzing and modeling check-in behavior. Area attraction says that each spatial area containing multiple venues has the ability to collectively attract visitation from users. Neighborhood competition determines the extent a venue competes with its neighbors in the same area to gain check-ins from users. We combine the two factors by the hypothesis that when a user decides a venue to visit, she will first select an area before she picks a particular venue in the area. This two stage process suggests that some areas attract more visitors than others. The choice of area will reduce the cognitive load on the user as she has fewer candidate venues in the area to choose from. To improve the accuracy of our modeling, we also incorporate *social homophily* into our model by allowing a user and her friends share more common venues.

Learning the area attraction and neighborhood competition factors from check-in data gives rise to several useful applications. Urban planners can redesign a city's transportation network by making attractive areas more accessible. Businesses need to know both area attraction and neighborhood competition in order to decide the new store locations that can maximize their profit. A store location recommendation system can also leverage on the two factors when making suggestions to its users.

¹ <https://foursquare.com/about>—Retrieved in August 2017.

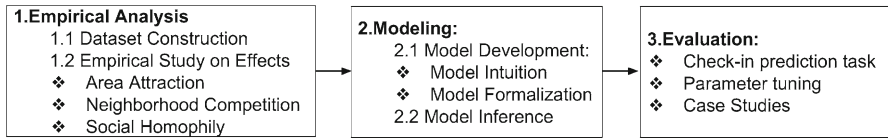


Fig. 1 Research framework

There are however several research challenges. Firstly, area attraction and neighborhood competition are new concepts that have not been formally studied earlier. It is not easy to illustrate the effects of these two factors using real world data. Hence, there is a need to conduct empirical studies on the factors. Secondly, the check-ins from users to venues are the results of multiple user and venue factors interacting with one another. Exactly how the interaction takes place is unclear. We thus have to create some generative stories to describe this interaction. Finally, there is no obvious ground truth in the datasets to evaluate the proposed model. We will need to adopt an indirect approach to conduct model evaluation.

We now describe the research steps carried out in this paper as shown in Fig. 1. First of all, we construct datasets for our research by crawling check-ins from LBSN and then conduct empirical studies on the datasets to illustrate the presence of area attraction, neighborhood competition and social homophily. The next step is modeling which includes two sub-steps: model development and model inference. The former introduces the intuitions behind the model as well as the mathematical formalization to capture the effects of venue and user factors on check-in behavior. The latter step develops algorithms to infer the parameters of our proposed model. Finally, the accuracy and robustness of our proposed model are evaluated using real world datasets. In particular, we evaluate our model using check-in prediction task. The experiments also evaluate our model under cold start condition and different parameter settings. Case studies are also examined to verify the effectiveness of our model.

Our results and findings of this research are summarized as follows:

- We introduce two important venue specific factors, i.e., area attraction and neighborhood competition. With real world LBSN datasets collected for three urban cities, we conduct an empirical analysis of the gathered check-in data and demonstrate the existence of neighborhood competition, area attraction factors. Furthermore, the effect of social homophily is also illustrated in our empirical analysis.
- We propose a matrix factorization-based model called **VAN** to capture the check-in behavior of users incorporating area attraction and neighborhood competition. Moreover, we also extend our model to incorporate *social homophily*.
- The performance of **VAN** model is evaluated on real world datasets so as to demonstrate its superior accuracy and robustness. In our experiments, we compare **VAN** model with other baselines in check-in prediction task. We show that **VAN** model outperforms the baselines. The parameters of **VAN** model are also carefully examined in our experiments.

Paper outline The remainder of the paper is organized as follows. Section 2 covers the literature review of previous works related to our research. Section 3 shows the data science aspect of our works to study check-in related factors. Section 4 describes our model and the parameter learning steps. Sections 5 shows its performance on real datasets. Lastly, Sect. 6 concludes the paper and suggests some future works.

2 Related work

In this section, we summarize related work in modeling check-ins considering different venue and user factors.

The visitation of users to venues occurs under the influence of multiple effects (Gao and Liu 2015). For example, distance effect (Chang and Sun 2011; Cho et al. 2011; Doan and Lim 2016; Huff 1963; Li et al. 2012) states that users tend to visit nearby venues rather than further away ones. This effect however will not be included in this research because it requires knowledge of users' home locations which are usually not available due to privacy reasons. In this section, we only focus on surveying previous research works on *Area Attraction*, *Neighborhood Competition* and *Social Homophily*. Before going into details of each effect, Table 1 summarizes the previous related works according to the factors considered in their models.

To the best of our knowledge, *area attraction* and *neighborhood competition* are two new features that have not been studied together in previous models. Our earlier work (Doan and Lim 2016) is the first work which examines both factors and builds a Bayesian model that incorporates both factors. Particularly, it models check-in behavior considering area attractiveness based on the aggregation of the competitiveness of the venues within each area. Moreover, it illustrates neighborhood competition by showing that check-ins within a small spatial area are usually performed on very

Table 1 Taxonomy of related works

	Area attraction	Neighborhood competition	Social homophily
Qu and Zhang (2013) and Huff (1963)			
Quan et al. (2012) and Yu et al. (2013)			
Karamshuk et al. (2013)	✓		
Church and Murray (2009)			
Fu et al. (2016)			
Doan et al. (2015a) and Liu et al. (2013)		✓	
Doan and Lim (2016)	✓	✓	
Gao et al. (2012b) and Li et al. (2016)			
Cheng et al. (2012) and Cho et al. (2011)			
Doan et al. (2015b) and Li et al. (2012)			✓
Ma et al. (2008) and Gao et al. (2012a)			
Ma et al. (2011)			
VAN (our model)	✓	✓	✓

few venues instead of uniformly across all venues in the area. The work then introduces a probabilistic model to combine neighborhood competition with distance effect and area attraction. While the proposed model improves the performance of check-in prediction over some baselines such as PMF (Mnih and Salakhutdinov 2008) and Expo-MF (Liang et al. 2016), it still has some limitations. Firstly, it requires the home locations of users, a private and not readily available information. Secondly, the work also assigns a competitiveness value to each venue based on how the venue wins over its neighboring venues in gaining check-ins without considering the latent factors of users and venues which account for the users' inherent interest on venues. In this work, we therefore improve this model by (1) discarding the user home location assumption and drop distance effect from model design (2) incorporating the user and venue latent factors to enhance the modeling of neighborhood competition.

Area attraction The effect focuses on that venues within a spatial area tend to support each other to gain visitation from users. The early work by Huff (Huff 1963) could be considered as the first work studying this effect. A specific shopping mall is an area in their model and its attractiveness is determined based on two factors: travel time from users' locations to the shopping mall and the area size of shopping mall. This work cannot be applied to data from LBSN since it again requires the home location information of users. Moreover, the work has not been applied to non-shopping mall venues which may not be affected by area size by the same degree. Qu and Zhang (2013) generalized the work of Huff (1963) and applied the Huff analysis method to data from LBSN. For each user, the proposed method derives his/her activity centers and defines the center of mass of his top 3 most active activity centers as the user's home location of user. It was found that the center of the mass and home location of 64% of users are less than 2 miles apart. Given the spatial closeness between user's center of mass and home location, Qu and Zhang (2013) used the former as the home location in Huff model.

There is some previous work (Church and Murray 2009; Fu et al. 2016; Karamshuk et al. 2013; Quan et al. 2012; Yu et al. 2013) which measures the attractiveness of areas using LBSN data for ranking the areas. However, with the lack of considering users' preference, the application of these approaches is limited to area ranking.

Yan et al. (2017) is an attractive recent work in understanding user movement. In their paper, they proposed a user movement model based on two assumptions (1) user chooses an area under the memory effect—user preferentially visits his/her previous locations (2) user chooses a venue based on its attraction which depends on its population. The differences between our work and their model are (1) their work is unable to model the choice of user at individual level (2) their work does not consider the matching between user preference and venue characteristics (3) their work models the attraction at venue level. Our model improves over their work by modeling area level attraction and by using matrix-factorization based technique to learn the preference of users.

Neighborhood competition Venues compete with their neighbors to attract users' visitation. The approach of Liu et al. (2013) is able to incorporate such information in their model. Specifically, it infers the popularity score of each venue which also

captures the competitiveness of the venue in its neighborhood. The work assumes that the probability of observing check-ins on venue j by user i is inversely correlated with the distance between i and j , popularity of venue j , and the interest of i to j . To model the interest of users on venues, the work utilizes *Latent Dirichlet Allocation* (Blei et al. 2003) and *Bayesian Non-negative Matrix Factorization* (Schmidt et al. 2009) to derive the latent factors of users and venues. In Doan et al. (2015a), PageRank model has been adapted to measure the competitiveness of venues. The work defines transition probabilities between users based on their check-in competition, as well as two variants of PageRank to model the competition of venues in LBSN. From their experiments, by comparing the result of their model with groundtruth, the authors conclude that modeling competition of venues provides a reasonable venue ranking in LBSN. In Doan and Lim (2017), the authors model neighborhood competition by adopting idea from personalized ranking in matrix factorization (Rendle et al. 2009). From their experiment, they conclude that neighborhood competition has more influence than spatial homophily in check-ins prediction.

Social homophily Social homophily is widely used to understand users' check-in behavior in LBSN (Gao et al. 2012b; Li et al. 2012). The work in Doan et al. (2015b) derived features based on *social homophily* to predict number of check-ins between a user and a venue. These features include the number of check-ins of his friends to the venue, and the number of check-ins of his friend to venues whose type is similar to the venue. Cheng et al. (2012) and Ma et al. (2011) introduced a regularizer to penalize the latent factor difference between users and their friends based on matrix factorization framework (Koren et al. 2009; Lee and Seung 2001; Mnih and Salakhutdinov 2008). Cho et al. (2011) proposed periodic mobility model by viewing check-ins locations of users as the mixture of check-ins near *home* and *work*. They later extended their model by considering the influence of users' friends. Their results concluded that using *social homophily* could more accurately predict users' movement behavior. Check-in prediction is a special class of product recommendation problems. Ma et al. (2008) showed that by considering social homophily, their proposed model SoRec improves up to 11% over the baselines in the prediction of ratings users assign to product items. Li et al. (2016) is a recent research work on studying users' movement in LBSNs by introducing three types of friends: *social friends*, *neighboring friends* and *location friends*. They developed a matrix factorization method to incorporate the visitation of these different types of friends so as to perform check-in venue prediction. Gao et al. (2012a) proposed a Bayesian model which combined the information of social network and historical check-in data of users. Particularly, they found that the history of users' check-ins has two properties: power law distribution and short-term effect. From the experiment, these two effects helped to explain the behavior of users' movement. However, their model does not include the preference of users and venues which can limit the understanding of users' behaviors.

3 Empirical analysis of check-in behavioral data

In this section, we conduct empirical analysis on check-in behavior of users to determine the presence of *area attractiveness*, *neighborhood competition* and *social*

homophily in the behavior. This empirical analysis and subsequently prediction task evaluation are performed on three datasets to be described in Sect. 3.1. Our empirical analysis are divided into three parts corresponding to area attraction, neighborhood competition, and social homophily which will be covered in Sects. 3.2, 3.3 and 3.4 respectively.

3.1 Datasets

In our research, we gathered the Foursquare check-in data of users and venues from two cities, Singapore and Jakarta. Both are major cities in Southeast Asia with more than 5M population. The two cities also have relatively many active Foursquare users performing check-ins. For more extensive evaluation, we also include the publicly available Gowalla dataset covering users and venues from New York City (Cho et al. 2011). The statistics of the three datasets are shown in Table 2.

SG dataset This dataset consists of 1.11 millions check-ins by 55,891 Singapore Foursquare users on 75,346 venues from August 15, 2012 to June 3, 2013 (see Table 2). The users and venues are determined to be located in Singapore based on their profile locations and venue location coordinates respectively. This dataset is the largest among the three.

JK dataset Similarly, we crawled another Foursquare dataset for the users and venues in Jakarta from July 2014 to May 2015. There are 119,618 check-ins performed by 14,974 users on 38,183 venues. **JK** dataset is the smallest among the three datasets.

NYC dataset To test our model in other LBSN platform, we use the public dataset of Gowalla from February 2009 to October 2010. Since we only focus on venues within city, we select check-ins of venues from New York City and denote them as **NYC**.

3.2 Area attraction

The empirical analysis of area attraction is non-trivial for a number of reasons. Firstly, to tell whether an area is attractive, we need some external knowledge for reference. For example, experts such as real estate valuers can determine the commercial value of an area using property and land sales information. Unfortunately, this approach is costly for us to adopt. Instead, we analyze the difference area can make to a set of venues that are expected to be similar in attracting visitors.

Table 2 Dataset statistics

Dataset	# users	# venues	# check-ins	# user-venue pairs with > 0 check-ins
SG	55,891	75,346	1.11M	541,588
JK	14,974	38,183	119,618	81,188
NYC	7092	21,287	138,067	102,906
H_SG	856	12,020	63,777	28,298
H_JK	455	4380	9557	5422

In this empirical analysis, we postulate that if different areas can be differentiated by attractiveness, users will then be more willing to make trips to visit venues in attractive areas.

To perform the analysis, we identify a subset of users whose the home locations could be determined so as to allow us to derive the distance between users and areas. The details below describe how we can extract this information

- We selected a subset of venues under the “home (private)” category which is in turn a sub-category of the “residence” category. We found 8447 and 1985 venues satisfying this criteria in the **SG** and **JK** datasets respectively.
- We further identified 3276 and 891 users who performed check-ins at only one “home (private)” venue each in the **SG** and **JK** datasets respectively. This rules out users who performed check-ins at multiple “home (private)” venues.
- We finally selected an even smaller set of users who also shouted some home relevant messages during their check-ins to their “home (private)” venues. These messages have to include some “home” related key phrases, e.g., “back home”, “home finally”, etc. For the **JK** dataset, we use the matching Indonesian key phrases like “Tidur dulu” (sleep first), “Rumah” (House), “Pondok” (cottage), “sampai di rumah” (arrived to home), “bobo” (sleep).

Since we do not obtain the shout of each check-ins in **NYC** dataset, the analysis does not involve **NYC** in this empirical experiment.

We finally obtained 856 users with home locations in the **SG** dataset. We denote the Foursquare data of these users and their check-in venues by **H_SG**. These users have 63,777 check-ins on 12,020 venues (see Table 2). Similarly, we obtained the **H_JK** dataset for 455 Jakarta users with home locations. This dataset covers 4380 venues and 9557 check-ins.

To embark this empirical analysis, we select all well known business chains which have more than three branches in each dataset. Specifically, McDonald, KFC and Starbucks are selected in both **H_SG** and **H_JK**. We expect branches of the same chain to be very similar to one another by food variety, food quality, ambience and service. Hence, at the venue level, we should not expect any difference among their abilities to attract users from other locations.

To construct areas for each dataset, we divide a city into square grid cells. We first determine the smallest rectangle that covers all venues of the city. We then divide the rectangle into square areas of width equals to 0.01° (equivalent to about 1.11 km on the equator) and assign every venue to exactly one area. Each area is assigned a center of the mass derived from the average of locations of its venues. We call the top five areas with most number of venues the *dense areas* while the areas from ranks 10 to 15 the *sparse areas*. We exclude other lower ranked areas as they do not contain any venues of the selected business chains.

For each business chain, we examine the distances between each dense area (represented by its center of mass) and the home locations of users who perform check-ins to its venues inside the area. We then generate a boxplot for the user-area distance of all the dense areas. We perform the same procedure for the sparse areas.

Figure 2 shows that for each business chain, branches within the *dense areas* attract users farther than branches in the *sparse areas*. This suggests that the attractiveness

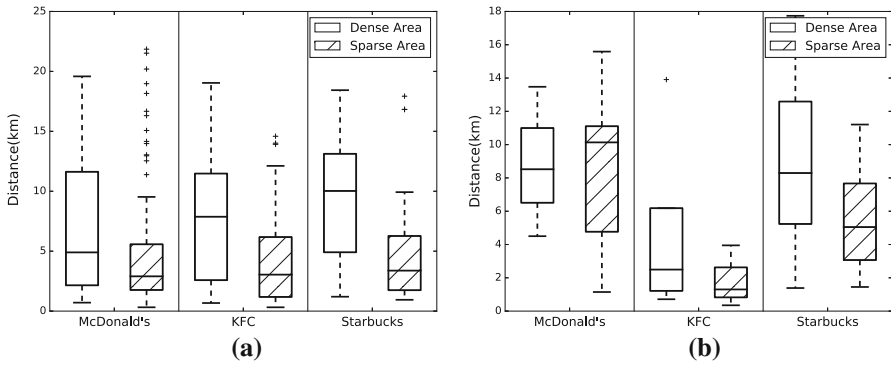


Fig. 2 Boxplot of distance from areas containing business chain to their check-ins users in **H_SG** and **H_JK**. **a H_SG**, **b H_JK**

Table 3 The number of stores in **H_SG** and **H_JK** datasets

	McDonald	KFC	Starbucks
H_SG	108	89	95
H_JK	37	101	94

of area plays an important role bringing far away users to the venues in the area. In Fig. 2, there is an exception involving McDonald branches in **H_JK** dataset. It could be attributed to the much fewer McDonald branches in **H_JK**, one third of that in **H_SG**. This may have caused Jakarta users having to travel further to the McDonald branches. The number of Starbuck and KFC venues in both dataset are quite similar (see Table 3).

3.3 Neighborhood competition

To show competition among venues within the same area, we adopt the method originally proposed by Weng et al. (2012) to study competition among memes. We divide the check-in history into weeks. We then measure the following entropies for each week.

- *System entropy* (E_s) $E_s(t) = - \sum_v f_v(t) \log f_v(t)$ where $f_v(t)$ is the fraction of check-ins in week t performed on venue v , i.e., $f_v(t) = \frac{\#cks(v,t)}{\sum_v \#cks(v,t)}$. The system entropy essentially measures the degree to which the distribution of check-ins concentrates on a small fraction of venues.
- *Average area entropy* (E_A) We first define the entropy of an area a to be $E_a(t) = - \sum_{v \in a} f_{v,a}(t) \log f_{v,a}(t)$ and $f_{v,a}(t) = \frac{\#cks(v,t)}{\sum_{v \in a} \#cks(v,t)}$. We then take the average of all area entropies, i.e., $E_A(t) = Avg_a E_a(t)$. We divide the city into square cells of 0.01° width. The construction of areas is discussed further in Sect. 4. Similar to system entropy, average area entropy captures the degree to which the distribution of check-ins of an area concentrates on a small fraction of venues (in the area).

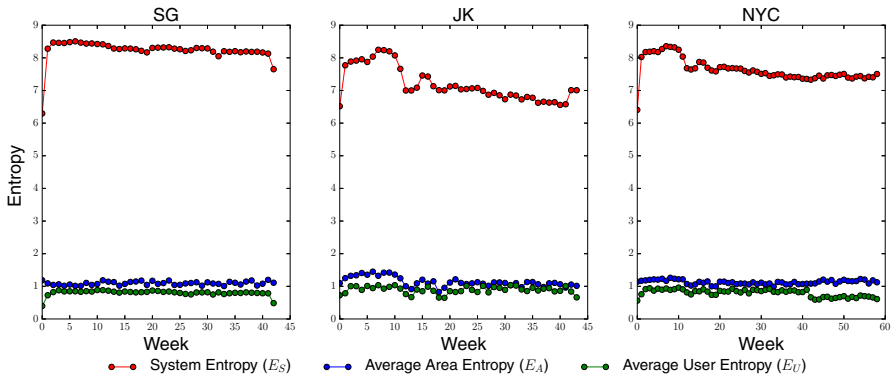


Fig. 3 Weekly entropy in SG, JK and NYC datasets

- *Average user entropy (E_U)* We next define the average user entropy as $E_U(t) = \text{Avg}_{u \in U} E_u(t)$ where entropy of user u is $E_u(t) = -\sum_v f_{u,v}(t) \log f_{u,v}(t)$ and $f_{u,v}(t) = \frac{\#cks(u,v,t)}{\#cks(u,t)}$. This entropy quantifies the concentration of users’ attention on the venues they perform check-ins on.

Figure 3 shows the three entropies over weeks in SG, JK and NYC datasets which remain mostly unchanged over the weeks. The first important observation is that the average user entropy is much smaller than system entropy. It clearly suggests that each user’s attention is limited to very small fraction of venues in the entire city. Venues therefore have to compete to gain attraction from users. Secondly, we observed from Fig. 3 that system entropy is much larger than average area entropy in both datasets. This implies that check-ins within an area concentrated on smaller fraction of venues than the fraction of venues in the entire city receiving check-ins from the whole user population.

The above empirical analysis concludes that venues compete more with their nearby neighbors than those farther away. Thus, grouping venues into areas and modeling competition among venues in each area is an appropriate modeling approach.

3.4 Social homophily

Social homophily is the tendency that users and their friends share more common check-in venues than that between users and other ones. To show the existence of social homophily, we calculate the average Jaccard similarity score of all pairs of users and their friends. Then, we compute the same score for equal number of random pairs of users.

Table 4 shows that the average Jaccard scores between users and their friends are significantly higher than that between random pairs of users. Moreover, the phenomenon is consistent across all the five datasets. The average Jaccard score between users and their friends is 3.1 times higher than that of pairs of random users in SG dataset. In the JK and NYC datasets, the Jaccard score between users and their friends is seven and eight times respectively larger than that of pairs of random users. Therefore, we

Table 4 Average Jaccard scores between user-friend pairs versus random pairs of users across five datasets

	SG	H_SG	JK	H_JK	NYC
Users and their friends	0.01411	0.01818	0.00697	0.01812	0.01921
Random pair of users	0.00448	0.00867	0.00097	0.00085	0.00211

Table 5 Table of notations

Notations	Meaning
U, V, C	Set of all users, venues and check-ins
U_i	Latent feature vector of user i
V_v	Latent feature vector of venue v
w_{iv}	Number of check-in of user i to venue v
w_v	Total number of check-in of venue v
a_v	Area a_v containing venue v
s	The width of area
$N(v)$	Set of neighbor venues of v
$L_a(\cdot)$	Logistic function with steepness a
$p(i \rightarrow a_v)$	Probability of user i visiting area a_v
$\lambda_u, \lambda_v, \lambda_f$	Regularization of user, venue vectors and friendship

conclude that in LBSNs, users share more check-in venues with their friends than with other users.

4 Proposed model

In this section, we propose a model called *Visitation by Attractiveness and Neighborhood competition* (VAN). The VAN model is an extension of standard matrix factorization to model check-in behavior incorporating area attraction, neighborhood competition and social homophily factors. In Sect. 4.1, we will first define the important concepts in the VAN model and its model assumptions. We then introduce the model formally in Sect. 4.2. The learning of VAN model parameters is given in Sect. 4.3.

4.1 Model description

In the VAN model, we model each user i or venue v as a vector of latent features U_i and V_v respectively. When user i and venue v have preferences on similar latent features, $U_i^T V_v$ returns a large value implying that user i is likely to perform check-in on venue v . We also use w_{iv} to denote the number of check-ins by user i on venue v . Readers can refer to Table 5 for the notations used in the VAN model.

To model area attraction, we divide the city into mutually exclusive square grid cells of width s . We use a_v to denote the square or *area* which contains v . The VAN model makes the following assumptions for each check-in between a user and a venue:

- First of all, every user chooses an area to perform a check-in based on a combination of area attractiveness and the user’s preference on the area. Area attractiveness is a quantitative measure defined to capture how well the area can attract users based on the venues within the area.
- Secondly, every venue inside an area must compete against its neighboring venues in order to gain a check-in from the user.

The **neighbors** of a venue v , denoted as $N(v)$, are venues within a_v and the areas adjacent to a_v are denoted by $Adj(a_v)$. That is, $N(v) = \{v' | v' \in Adj(a_v)\} \cup \{v' | v' \in a_v\} \setminus \{v\}$. We consider the venues in $Adj(a_v)$ as neighbors because we want to include venues in these nearby areas as competitors of v even when v is near the border of a_v .

For user i , the **attractiveness** $\sigma_{a_v}^i$ of area a_v is defined by the summation of the interaction between the user preference U_i and each latent features $V_{v'}$ of venue v' inside an area a_v . That is, $\sigma_{a_v}^i = \sum_{v' \in a_v} U_i^T V_{v'}$. It means that the venues inside the area contribute their preference together to attract the check-in from user i .

Every check-in of user i to venue v follows a two-step process. Firstly, user i must select the area a_v . Secondly, the venue v in area a_v must win over all other neighboring venues in $N(v)$ to gain a check-in from user i .

- User i selects the area a_v under the effect of attractiveness $\sigma_{a_v}^i$ of area a_v . We represent this by assigning a probability which is proportion to $\sigma_{a_v}^i$.
- To model the winning of venue v over its neighbors, we need to employ the preference of user i since he/she is the main factor to decide if the visitation is made or not. We assume that if the latent similarity between user i and venue v is higher than the one between user i and the neighbors v' of v , the probability that i visits v (denoted as $p_i(v > v')$) is higher than the one between i and v' . We therefore map the value of $U_i^T V_v - U_i^T V_{v'}$ to interval $[0, 1]$ so as to model $p_i(v > v')$. When $p_i(v > v') > p_i(v' > v)$, user i is likely to make check-in on v rather than v' . We define $p_i(v > v') = L_a(U_i^T V_v - U_i^T V_{v'}) = \frac{1}{1 + \exp(-a(U_i^T V_v - U_i^T V_{v'}))}$ where L_a is a logistic function (Jordan et al. 1995) with steepness parameter a . Logistic function is a function family which Sigmoid function belong to. Sigmoid function is a logistic function with $a = 1$. When a goes to infinity, logistic function turns into an indicator function as shown in Fig. 4.

Example Figure 5 depicts two check-ins at venue v by user i i.e. $w_{iv} = 2$. To perform each check-in at venue v , user i has to select area $(b, 3)$ (enclosed by a red box) considering similarity between the preference of user i and the venues within the area. Moreover, venue v needs to *win* over all of its neighbors in the adjacent areas enclosed by the square box in green.

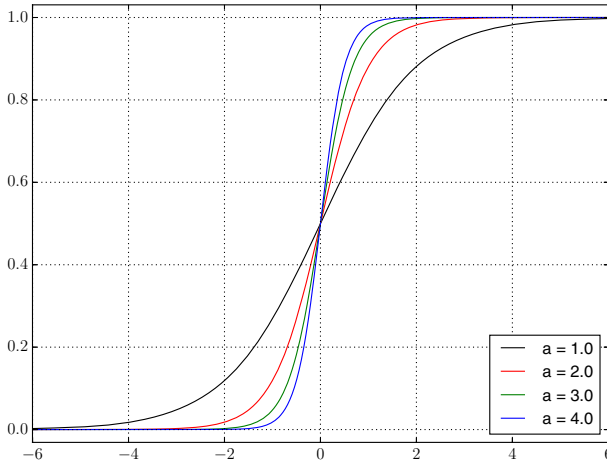


Fig. 4 Logistic function with different values of steepness

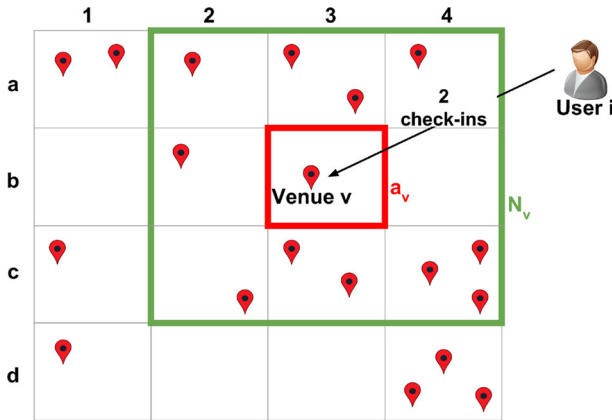


Fig. 5 Example of Check-in graph

4.2 Model formalization

We now formally define the VAN model. In the VAN model, the probability p_{iv} of a check-in from user i to venue v is defined by the following formula:

$$p_{iv} = p(i \rightarrow a_v) \prod_{v'' \in N(v)} p_i(v > v'') \tag{1}$$

Equation 1 says that p_{iv} has two components inside: $p(i \rightarrow a_v)$ denoting the probability of user i selecting area a_v and $p_i(v > v'')$ representing the probability that user i prefers to perform check-in on venue v over its neighbor v'' .

Recall that U_i and V_v denote the latent feature vectors of user i and venue v respectively. We thus define $p(i \rightarrow a_v)$ as

$$p(i \rightarrow a_v) = \sum_{v' \in a_v} p(v'|i) = \sigma_{a_v}^i = \sum_{v' \in a_v} U_i^T V_{v'} \tag{2}$$

The second component of Eq. 1 is defined as:

$$p_i(v > v'') = L_a(U_i^T V_v - U_i^T V_{v''}) \tag{3}$$

By substituting the components in Eq. 1, we have:

$$\begin{aligned} p_{iv} &= \left(\sum_{v' \in a_v} p(v'|i) \right) \prod_{v'' \in N_v} p_i(v > v'') \\ &= \left(\sum_{v' \in a_v} U_i^T V_{v'} \right) \prod_{v'' \in N_v} L_a(U_i^T V_v - U_i^T V_{v''}) \\ \log p_{iv} &= \log \sum_{v' \in a_v} U_i^T V_{v'} + \sum_{v'' \in N_v} \log L_a(U_i^T V_v - U_i^T V_{v''}) \end{aligned} \tag{4}$$

Next, we define the log-likelihood $\mathcal{L}(C)$ of a set of check-ins C from users of U on venues of V has the following form:

$$\mathcal{L}(C) = \sum_{(i,v) \in C} w_{iv} \log p_{iv} = L_1(C) + L_2(C) \tag{5}$$

where

$$\begin{aligned} L_1(C) &= \sum_{(i,v) \in C} w_{iv} \log \left(\sum_{v' \in a_v} U_i^T V_{v'} \right) \\ L_2(C) &= \sum_{(i,v) \in C} w_{iv} \sum_{v'' \in N_v} \log L_a(U_i^T V_v - U_i^T V_{v''}) \end{aligned} \tag{6}$$

To learn the latent features and other variables of users and venues in VAN model, we maximize the log-likelihood defined in Eq. 5. Formally, we have the optimization problem as below:

$$\{U_i^*, V_v^*\}_{i \in U, v \in V} = \arg \max_{i \in U, v \in V} (\mathcal{L}(C) - \lambda(C)) \tag{7}$$

where $\lambda(C)$ is the regularization term that prevent overfitting (Friedman et al. 2001). In our model, we use L_2 norm for $\lambda(C)$ since it can be solved easily (Friedman et al.

2001) and it is widely applied in matrix factorization method (Koren et al. 2009; Lee and Seung 2001; Mnih and Salakhutdinov 2008). Formally, $\lambda(C)$ is defined as

$$\lambda(C) = \lambda_u \sum_i \|U_i\|_2^2 + \lambda_v \sum_v \|V_v\|_2^2 \quad (8)$$

where λ_u and λ_v are the regularization parameters for the latent features of users and venues respectively.

Incorporating social homophily Similar to Cheng et al. (2012), we model social homophily by adding a social regularizer $\lambda_f \sum_{(i,i') \in F} \|U_i - U_{i'}\|^2$ to Eq. 7. In other words, if two users i and i' have social connection between them, their latent feature vectors U_i and $U_{i'}$ are expected to be similar. λ_f is the parameter to control the importance of social homophily effect. Formally, we have a new objective function

$$\{U_i^*, V_v^*\}_{i \in U, v \in V} = \arg \max_{i \in U, v \in V} (\mathcal{L}(C) - \Lambda(C)) \quad (9)$$

where

$$\Lambda(C) = \lambda(C) + \lambda_f \sum_{(i,i') \in F} \|U_i - U_{i'}\|^2 \quad (10)$$

4.3 Model inference

To solve the optimization problem in Eqs. 7 and 9, we use *Stochastic Gradient Descent* (SGD) (Boyd and Vandenberghe 2004). SGD is a widely used technique to learn latent features in matrix factorization-based framework (Hu et al. 2014; Liu et al. 2014; Koren et al. 2009)

In SGD, we first derive the derivative of the objective function with respect to each variable. Each step of SGD only considers one user-venue pair (i, v) .

We firstly select one user-venue pair randomly and take the derivative of user feature vector U_i of the regularization

$$\frac{\partial \Lambda((i, v))}{\partial U_i} = 2\lambda_u U_i + 2\lambda_f \sum_{(i,i') \in F} (U_i - U_{i'}) \quad (11)$$

$$\begin{aligned} \frac{\partial L_1((i, v))}{\partial U_i} &= w_{iv} \frac{1}{\sum_{v' \in a_v} U_i^T V_{v'}} \sum_{v' \in a_v} \frac{\partial U_i^T V_{v'}}{\partial U_i} \\ &= w_{iv} \frac{1}{\sum_{v' \in a_v} U_i^T V_{v'}} \sum_{v' \in a_v} V_{v'} \end{aligned} \quad (12)$$

$$\frac{\partial L_2((i, v))}{\partial U_i} = w_{iv} \sum_{v'' \in N_v} \frac{1}{L_a(U_i^T V_v - U_i^T V_{v''})} \frac{\partial L_a(U_i^T V_v - U_i^T V_{v''})}{\partial U_i} \quad (13)$$

To simplify the formula, we introduce $d_{i,v,v''} = U_i^T V_v - U_i^T V_{v''}$. Recall that $L_a(d_{i,v,v''})$ is Logistic function of $d_{i,v,v''}$ with steepness a i.e. $L_a(d_{i,v,v''}) = \frac{1}{1+\exp(-a d_{i,v,v''})}$. Hence, we have the derivative of $L_a(d_{i,v,v''})$ respected to U_i :

$$\frac{\partial L_a(d_{i,v,v''})}{\partial U_i} = \frac{a}{(1 + \exp(-a d_{i,v,v''}))^2} \exp(-a d_{i,v,v''})(V_v - V_{v''}) \tag{14}$$

Secondly, we take the derivative of V_v . The derivative of regularization is

$$\frac{\partial \Lambda((i, v))}{\partial V_v} = 2\lambda_v V_v \tag{15}$$

The derivative of each component of the log-likelihood regarding V_v is

$$\begin{aligned} \frac{\partial L_1(i, v)}{\partial V_v} &= w_{iv} \frac{1}{\sum_{v' \in a_v} U_i^T V_{v'}} U_i + \sum_{v^* \in a_v} w_{iv^*} \frac{1}{\sum_{v' \in a_v} U_i^T V_{v'}} U_i \\ \frac{\partial L_2(i, v)}{\partial V_v} &= w_{iv} \sum_{v'' \in N_v} \frac{1}{L_a(d_{i,v,v''})} \frac{\partial L_a(d_{i,v,v''})}{\partial V_v} \end{aligned} \tag{16}$$

Therefore, we have the derivative of $L_a(d_{i,v,v''})$ respected to V_v as follow:

$$\frac{\partial L_a(d_{i,v,v''})}{\partial V_v} = \frac{a}{(1 + \exp(-a d_{i,v,v''}))^2} \exp(-a d_{i,v,v''}) U_i \tag{17}$$

The second step of SGD is to update latent feature vectors of users and venues

$$\begin{aligned} U_i &\leftarrow U_i - \alpha \left(\frac{\partial \mathcal{L}(i, v)}{\partial U_i} - \frac{\partial \Lambda(i, v)}{\partial U_i} \right) \\ V_v &\leftarrow V_v - \alpha \left(\frac{\partial \mathcal{L}(i, v)}{\partial V_v} - \frac{\partial \Lambda(i, v)}{\partial V_v} \right) \end{aligned} \tag{18}$$

where α is the learning step parameter of SGD.

Then, we repeat to the first step until the objective function gets convergence.

5 Experiment

In the absence of ground truth data, our model will be evaluated via *check-in prediction task* which predicts the number of check-ins between user-venue pairs. We compare the check-in prediction performance of our model with other baselines. We will also study the effects of model parameter settings on the model performance. These parameters include the steepness of Logistic function, area width and regularization. The variant of VAN model with social homophily denoted as VAN_s is also evaluated in the next experiment. Finally, we conduct experiment to evaluate the effectiveness of VAN

model in venue ranking against the Foursquare venue scores. We also present some latent feature of venues learned by VAN.

5.1 Experiment setup

We divide check-in data into training and test sets. We sort check-ins in the **SG**, **JK** and **NYC** datasets by their created time and then divide each dataset into five folds. For each run of experiment, we hide one fold as test set and use the remaining four ones as training set. Particularly, for each run, we use four folds for learning model parameters, then these learned values are used to predict the number of check-ins between users and venues in the hidden fold.

Performance measures We use two sets of metrics to measure the performance of our models as well as the baselines. The first set consists of $recall@k$ and $nDCG@k$ which focus more on top ranked results returned by each model. The second set includes *average precision (AP)* and *area under the curve (AUC)* which measure the overall performance.

After training, for each user, we rank all venues according to their prediction scores returned by each model. The venues visited by the same user in the test data are the ground truth. We then compute the different performance measures based on the predicted venue ranking. The performance measures are averaged over all users. We finally derive the mean of the average performance measures over all the folds. We do not use $precision@k$ because we cannot distinguish between a user disliking a venue and a user not knowing the venue (Wang and Blei 2011).

The formula of $recall@k$ and $nDCG@k$ are presented below:

$$\begin{aligned} recall@k &= \frac{1}{|U|} \sum_{u \in U} \frac{|\mathcal{L}(u, k) \cap \mathcal{L}^{test}(u)|}{|\mathcal{L}^{test}(u)|} \\ nDCG@k &= \frac{1}{|U|} \sum_{u \in U} \frac{DCG@k_u}{IDCG@k_u} \end{aligned} \quad (19)$$

where $\mathcal{L}(u, k)$ is the top k venues of each user u returned by the model; $\mathcal{L}^{test}(u)$ represents the set of venues of user u in test set. Function $|\cdot|$ returns the set cardinality.

$DCG@k_u = \sum_{i=1}^{|\mathcal{L}(u, k)|} \frac{2^{rel_{ui}} - 1}{\log_2(i+1)}$ and $IDCG@k_u = \sum_{i=1}^{|\mathcal{L}^{test}(u)|} \frac{2^{rel_{ui}} - 1}{\log_2(i+1)}$. To measure $DCG@k$, we first select the top k venues of each user returned by each method. rel_{ui} is the relevance score of the i th rank venue of user u . In our experiment, $rel_{ui} = 1$ if $i \in \mathcal{L}^{test}(u)$; otherwise, $rel_{ui} = 0$. The $nDCG@k_u$ is $DCG@k_u$ normalized by the $DCG@k_u$ of the ideal ranking $IDCG@k_u$ of top- k venues for user u .

The formal definitions of AUC and AP are described below:

$$\begin{aligned} AUC &= \frac{1}{|U|} \sum_{u \in U} \frac{1}{|E(u)|} \sum_{(v, v') \in E(u)} \delta(p_{uv} > p_{uv'}) \\ AP &= \frac{1}{|U|} \sum_{u \in U} \sum_n (R_n^u - R_{n-1}^u) P_n^u \end{aligned} \quad (20)$$

where $E(u) = \{(v, v') | v \in \mathcal{L}^{test}(u) \wedge v' \notin (\mathcal{L}^{test}(u) \cup \mathcal{L}^{train}(u))\}$ and $\mathcal{L}^{train}(u)$ represents the set of venues of user u in training set. In other words, $E(u)$ is the set of venue pairs whose one is in test set of user u but the other is a venue without having any implicit feedbacks from user u . Function $\delta(\cdot)$ is the indicator function return 1 if the boolean expression inside is true and 0 otherwise.

AP is average precision metric which summarizes the plot as the weighted mean of precision achieved at each threshold with the increase in recall from the previous threshold used as the weight. In the formula of AP , P_n^u and R_n^u are the precision and recall at the n th threshold of user u .

Default parameter setting For all experiments, we set the number of latent features to 10. The width of area is $s = 0.01$ geographical degree. The default steepness of Logistic function is $a = 2.0$ since it yields us the best prediction performance for the VAN model (see more details in Sects. 5.4 and 5.6). For regularization, we use the default $\lambda_u = \lambda_v = 0.01$ because it does not bias toward users nor venues. In most of the experiments, we use $\lambda_f = 0$ since the performance with and without social homophily of VAN model show the same trends. The learning rate of SGD algorithm is kept at 10^{-6} .

5.2 Check-in prediction

In this section, we compare the performance of our VAN model and its extension VAN_s with social homophily with several baseline models. These baseline models are also based on matrix factorization framework and they include:

- Probabilistic Matrix Factorization *PMF* (Mnih and Salakhutdinov 2008): *PMF* factorizes check-in matrix into user-latent factor and venue-latent factor matrix alone. We use the number of latent factors $K = 10$. We use the implementation provided by the authors.²
- Multi-center Gaussian Model *MGM* (Cheng et al. 2012): *MGM* uses multiple Gaussian distributions to model the activity centers of users. For each user, we automatically detect the clusters of check-ins by applying the non-parametric method from Blei and Jordan (2006). We use the *MGM* implementation from Scikit-learn (Pedregosa et al. 2011). Each cluster is assigned as an activity center of a user. The α parameter of *MGM* which controls the weight of high frequent check-ins venues is set to default value $\alpha = 0.2$.
- Fusion Framework *PMF-MGM* (Cheng et al. 2012): *PMF-MGM* combines matrix factorization and *MGM*. It is reported to outperform *PMF* and *MGM* models. The probability of a user visiting a venue is determined by fusing the user's preference on that venue (returned by *PMF*) and the probability of if he/she will visit that place (returned by *MGM*).
- Matrix Factorization with Neighborhood Influence *N-MF* (Hu et al. 2014): *N-MF* explores the characteristics of geographical neighbors based on the matrix factorization framework. The authors focused on studying the *spatial homophily*. We use the number of latent features $K = 10$ and two venues are neighbors if

² <https://www.cs.cmu.edu/rsalakhu/software.html>.

their distance is less than a predefined threshold d . In our experiment, we set d to be 100m and 200m.

- Exposure Matrix Factorization *Expo-MF* (Liang et al. 2016): Expo-MF incorporates the location of venues and user exposure into the modeling of check-ins behavior of users. Similar to their experiment conducted in Liang et al. (2016), we apply K-Means to cluster venues, the location vector of each venue is its probability to each cluster. We use $K = 10$ for both the number of latent factors and the number of clusters in K-Means.³
- Social Bayesian Personalized Rankings *SBPR* (Zhao et al. 2014): SBPR assumes that users tend to assign higher ranks to items that their friends prefer. In our experiment, we adopt the default parameters represented in the original paper. Specifically, the number of latent feature is set to 10 and the regularization parameters of users, venues and bias are 0.015, 0.025 and 0.01 respectively.

Parameter setting For our experiment, we adopt a *default parameter setting*. The number of latent factors is 10 by default to compare fairly with the baselines i.e. $f = 10$. The steepness of logistic function is $a = 2.0$, the width of area is $s = 0.01$. For regularization, we use $\lambda_u = \lambda_v = 0.01$. We also test the performance of the extension VAN_s with social homophily. In VAN_s , the regularization of social homophily is $\lambda_f = 0.01$.

Result Table 6 shows the performance of our VAN model and the baselines under different metrics. Recall that the larger the value of each metric, the better the model. Therefore, the most important observation which we could draw from the table is that our model with default parameter setting outperforms all the baselines in general. In **SG**, **JK** and **NYC** datasets, the performance of our methods is always better than the baselines but the performance gap between VAN and the baselines in **SG** dataset is larger than that in **JK** and **NYC** datasets. The reason is that the data of **JK** and **NYC** is sparser than the one of **SG** dataset. Among the baselines, *PMF-MGM* and *Expo-MF* perform better than other baselines. It happens due to the fact that these baselines cluster venues in dataset into different groups so that they could create some area attraction effects. VAN model takes one step further by integrating the neighborhood competition inside. From the observation, we could conclude that the impact of neighborhood competition is crucial in understanding the visitation of users in LBSNs.

From Table 6, we observe that using *social homophily* actually improves the performance of our model since the performance of VAN_s is higher than that of VAN in the **SG**, **JK** and **NYC** datasets. The second observation is that the improvement with *social homophily* is more significant in **JK** and **NYC** dataset than in **SG** dataset. For example, in **SG** dataset, *social homophily* helps us enhance 6.13% on average. The improvement in **JK** dataset is 12.03%. The reason behind is that **JK** and **NYC** is sparser than **SG** so the additional information including to **JK** or **NYC** has more effective than the denser one (i.e. **SG** dataset).

The performance of SBPR depends heavily on the social networks of users. It is therefore not a surprise that its performance in the three datasets are not higher

³ <https://github.com/dawenl/expo-mf>.

Table 6 Check-in prediction results; we boldface the best results for each performance measure

Metrics	VAN (%)	VAN _s (%)	PMF (%)	MGM (%)	PMF-MGM (%)	N-MF (%)		Expo-MF (%)	SBPR (%)
						100m	200m		
SG									
<i>recall</i> @20	7.06 ‡	7.71 *‡	1.93	1.3	2.21	0.93	0.9	6.5	1.17
<i>recall</i> @50	10.84 ‡	11.24 *‡	2.6	2.17	3.12	1.3	1.26	7.8	1.95
<i>recall</i> @100	14.46 ‡	15.26 *‡	3.42	3.22	3.92	1.61	1.6	9.12	2.4
<i>nDCG</i> @20	9.21 ‡	9.5 *‡	5.21	4.92	5.08	1.94	1.4	8.69	3.21
<i>nDCG</i> @50	6.9 ‡	7.32 *‡	4.43	4.05	4.55	1.67	1.07	6.12	2.54
<i>nDCG</i> @100	6.08 ‡	6.85 *‡	4.13	3.83	4.16	1.11	0.94	5.72	2.03
AP	70.21 ‡	72.11 *‡	61.17	59.73	61.81	54.65	53.91	68.11	53.17
AUC	74.18 ‡	75.05 *‡	60.73	58.14	61.9	55.59	54.09	72.08	51.25
JK									
<i>recall</i> @20	3.63 ‡	4.03 *‡	2.5	0.15	2.8	0.17	0.175	2.7	0.75
<i>recall</i> @50	6.5 ‡	7.3 *‡	3.86	0.23	3.51	0.67	0.8	4.81	1.01
<i>recall</i> @100	8.75 ‡	9.87 *‡	5.81	0.31	5.9	1.8	1.95	6.01	1.78
<i>nDCG</i> @20	5.2 ‡	5.95 *‡	2.61	1.07	2.71	1.2	1.25	4.87	1.63
<i>nDCG</i> @50	4.74 ‡	5.02 *‡	2.09	0.92	2.44	0.94	0.95	4.05	1.13
<i>nDCG</i> @100	4.09 ‡	4.63 *‡	1.84	0.79	1.98	0.84	0.86	3.82	0.92
AP	68.28 ‡	69.78 *‡	58.28	54.28	59.79	53.25	54.39	62.02	55.77
AUC	75.41 ‡	76.35 *‡	61.51	58.12	52.13	49.23	47.42	74.08	56.36

Table 6 continued

Metrics	VAN (%)	VAN _s (%)	PMF (%)	MGM (%)	PMF-MGM (%)	N-MF (%)		Expo-MF (%)	SBPR (%)
						100 m	200 m		
NYC									
<i>recall</i> @20	4.39	4.53 *	3.2	1.47	3.51	2.07	2.51	4.78	2.72
<i>recall</i> @50	5.52 ‡	5.88 *‡	4.84	2.89	4.94	3.64	4.21	5.28	4.28
<i>recall</i> @100	7.58 ‡	8.17 *‡	6.26	3.4	6.93	4.29	4.95	6.91	4.89
<i>nDCG</i> @20	6.72 ‡	6.89 *‡	3.15	2.73	3.75	2.83	2.89	5.92	2.8
<i>nDCG</i> @50	5.27 ‡	6.06 *‡	2.43	2.18	2.58	2.34	2.44	5.01	2.03
<i>nDCG</i> @100	4.76 ‡	4.9 *‡	1.85	1.34	1.92	1.95	2.05	4.52	1.85
<i>AP</i>	69.54 ‡	69.71 *‡	61.45	58.73	62.12	59.51	59.91	65.29	60.24
<i>AUC</i>	74.15 ‡	75.92 *‡	62.38	58.14	63.49	59.66	60.15	73.4	61.52

$a = 2.0$, $s = 0.01$, $f = 10$, $\lambda_u = \lambda_v = 0.01$ and $\lambda_f = 0.01$ for VAN_s. The symbol * indicates that VAN_s method performs significantly better than VAN while ‡ indicates VAN or VAN_s performing significantly better than the best baseline

than Expo-MF which focuses more on location of venues. Specifically, among the three datasets, **NYC** has the highest ratio of social connection and total pairs of users (0.004%) but this ratio of the four datasets mentioned in the original paper (Zhao et al. 2014) is at least two times larger (0.01%). The reason could be that users in LBSN network focus more on spreading their visitation than building social network.

Significant test We further apply the hypothesis testing to examine if the improvement of our model is actually significant over the baselines. Since we have many baselines, we only compare the performance of VAN and VAN_s with the best baseline (i.e. Expo-MF). In this case, the *null hypothesis* is that the performances of our models (i.e. VAN and VAN_s) and the baseline are not different while *alternative hypothesis* is that our models are significantly better than the baseline. To verify the hypotheses, we apply pair t test (Hsu and Lachenbruch 2008) to compare the result of each metrics of VAN and VAN_s to the selected baseline. From the result in Table 6, we show that our models (VAN and VAN_s) are significant better than the best baseline (i.e. Expo-MF) in most of the cases. For the case of $recall@20$ in **NYC** dataset, the significant test fails to verify Expo-MF is better than VAN and VAN_s models. Particularly, the p value of the test is 0.07 so the outperformance of Expo-MF is not significantly better than our model. Moreover, we also apply the above statistical test to the results of VAN_s and VAN to illustrate if social homophily actually improves the performance of our model. Particularly, the *null hypothesis* is that the performance of both VAN and VAN_s models are not different while the *alternative hypothesis* is that VAN_s is significantly better than VAN model. As shown in Table 6, using social homophily helps us improve the performance of VAN model significantly.

5.3 Check-in prediction for cold start users

In this section, we evaluate VAN and VAN_s for cold start users who do not have many check-in records in our datasets.

Setup In this experiment, we keep the same test set as the previous one but in the training set, we define a user to be a cold start user if he/she has not more than 5 check-ins. The remaining users are removed from the training sets.

Parameter settings In this experiment, we keep the default parameter setting of VAN and VAN_s as described in Sect. 5.1. For the baselines, we use the parameter as described in the previous experiment.

Result Table 7 shows the performance of our models and the baselines. In most of the cases, the performances of VAN and VAN_s are better than the performances of the baselines. We have one exception of AUC in **JK** dataset when Expo-MF outperforms VAN model with a small gap. In this experiment, we also observe that Expo-MF is the best among the baseline models. For this reason, we apply the significant test between our models (i.e. VAN and VAN_s) and Expo-MF to check if our models are significantly better than the best baseline. Moreover, we also test the significance of

Table 7 Check-in prediction task (cold start users)

Metrics	VAN (%)	VAN _s (%)	PMF (%)	MGM (%)	PMF-MGM (%)	N-MF (%)		Expo-MF (%)	SBPR (%)
						100m	200m		
SG									
<i>recall</i> @20	7.09‡	7.92*‡	1.05	0.91	1.58	0.5	0.51	4.2	1.55
<i>recall</i> @50	8.81‡	9.06*‡	1.46	1.13	1.91	0.55	0.62	5.9	2.17
<i>recall</i> @100	8.94‡	9.65*‡	2.9	1.87	2.95	0.71	0.75	6.75	4.87
<i>nDCG</i> @20	7.13 ‡	8.9*‡	2.21	1.57	2.35	0.98	1.1	5.87	1.82
<i>nDCG</i> @50	6.44‡	7.32*‡	1.84	1.06	1.95	0.52	0.78	4.19	1.13
<i>nDCG</i> @100	5.08 ‡	6.13*‡	0.86	0.45	1.07	0.5	0.56	3.39	0.87
AP	65.91‡	67.41*‡	55.18	53.12	58.58	50.75	52.37	61.78	57.29
AUC	67.18‡	69.79*‡	52.18	51.14	53.09	51.45	53.91	63.46	58.45
JK									
<i>recall</i> @20	3.52‡	4.18*‡	1.03	0.93	1.34	0.67	0.72	2.86	1.37
<i>recall</i> @50	4.45‡	5.73*‡	1.27	1.02	1.96	0.93	0.98	3.42	2.31
<i>recall</i> @100	4.96‡	6.54*‡	1.88	1.25	2.37	1.71	1.83	4.04	3.28
<i>nDCG</i> @20	4.06‡	4.67*‡	1.02	0.98	1.24	0.82	0.93	3.67	1.31
<i>nDCG</i> @50	3.63‡	3.88*‡	0.95	0.74	1.03	0.71	0.81	2.54	1.03
<i>nDCG</i> @100	3.16‡	3.25*‡	0.88	0.58	0.91	0.68	0.7	2.01	0.92
AP	62.25‡	64.51*‡	53.17	52.18	53.58	52.25	52.94	60.71	55.48
AUC	61.87	64.35*	51.28	52.72	53.39	51.92	51.46	62.04	56.84

Table 7 continued

Metrics	VAN (%)	VAN _s (%)	PMF (%)	MGM (%)	PMF-MGM (%)	N=MF (%)		Expo-MF (%)	SBPR (%)
						100 m	200 m		
NYC									
<i>recall@20</i>	3.89 ‡	4.15 *‡	1.32	1.05	1.48	1.06	1.2	2.51	2.77
<i>recall@50</i>	4.55 ‡	4.78 *‡	1.59	1.3	1.68	1.4	1.77	3.17	3.71
<i>recall@100</i>	5.61 ‡	5.83 *‡	1.84	1.42	1.91	1.74	1.89	4.54	4.53
<i>nDCG@20</i>	3.66 ‡	3.78 *‡	1.2	1.01	1.57	1.2	1.24	2.62	2.83
<i>nDCG@50</i>	2.85 ‡	3.04 *‡	1.05	0.96	1.09	1.1	1.12	2.00	2.32
<i>nDCG@100</i>	2.15 ‡	2.58 *‡	0.92	0.83	0.98	0.99	1.09	1.87	2.01
<i>AP</i>	55.21 ‡	58.91 *‡	51.15	49.77	51.72	50.17	50.23	52.43	53.67
<i>AUC</i>	52.12 ‡	54.76 *‡	50.14	48.66	50.21	50.25	50.3	51.11	51.21

We boldface the best result for each performance measures. The parameters $a = 2.0, s = 0.01, f = 10, \lambda_d = \lambda_v = 0.01$ and $\lambda_f = 0.01$ for VAN_s. The symbol * indicates that VAN_s performs significantly better than VAN while ‡ indicates the superiority of VAN or VAN_s over the best baseline according to significant testing

improvement of adding social homophily by comparing VAN and VAN_s . From the result shown in Fig. 7, we find that VAN and VAN_s are significantly better than Expo-MF. Moreover, adding social homophily actually improves the performance of model. For the exception of AUC for **JK**, we also apply the statistical test but could not find Expo-MF perform significantly better than VAN and VAN_s .

As VAN and VAN_s are very similar and share similar performance, we will study the impact of parameter settings to VAN model only in the following subsections.

5.4 Tuning the steepness parameter

In this section, we seek to understand the role of steepness of Logistic function in modeling check-ins and its use in check-in prediction task. We try out different steepness values and observe its impact to our model performance. In this set of experiments, we only involve VAN model.

Parameter setting In this experiment, we vary the steepness variable a from 1.0 to 4.0 with a step size of 0.1 while keeping default values for the remaining parameters.

Result Figure 6 shows the performance of VAN model with different steepness values. The best performance occurs when the value of steepness $a = 2.0$ for the **SG** and $a = 3.0$ for both **JK**, **NYC** datasets. Since $a = 2.0$ yields reasonably good results for all the three datasets, using this setting as default is reasonable. We also observe that the performance of VAN model degrades with larger a settings. The reason is that larger steepness values make Logistic function behaves like an indicator function which no longer nicely models the probability of competition among venues.

5.5 Tuning the regularization parameters

In this section, we try to figure out the impact of regularization parameters in modeling movement of users through check-in prediction task. To achieve the goal, we try out different values of regularization parameters. In this set of experiments, we only involve VAN model.

Parameter setting In this experiment, we keep the value of λ_u equal to that of λ_v since we do not want to bias to user or venue features. Recall that λ_u and λ_v are regularization parameters for the latent features of users and venues respectively. Then, we tune the values of them within the range 0 and 1 while keeping default values for the remaining parameters.

Result Figure 7 shows the performance of VAN model for the three datasets **SG**, **JK** and **NYC** with different metrics. From the figure, we observe that without regularization (i.e. $\lambda_u = \lambda_v = 0$), the performance of VAN does not perform well while increasing the value of regularization parameter also harms our model. From the figure, we can observe that selecting $\lambda_u = \lambda_v = 0.01$ yields good check-in prediction

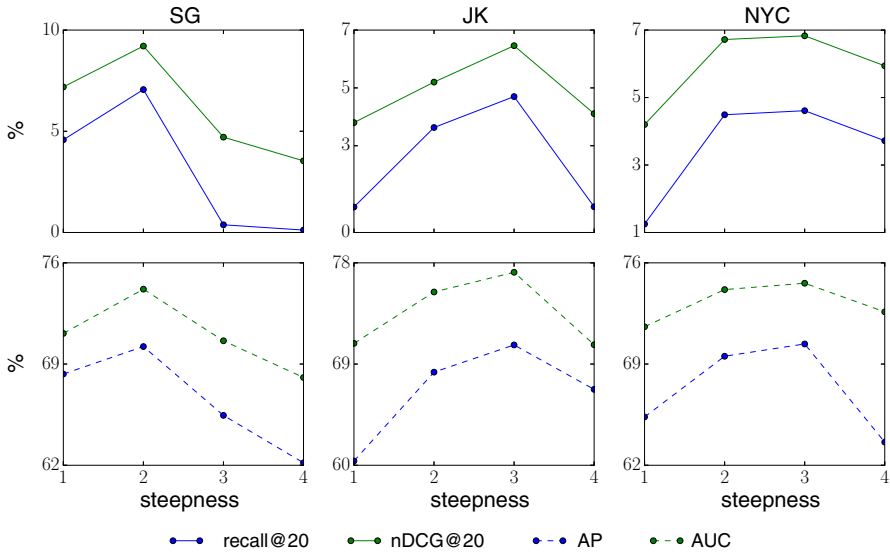


Fig. 6 Performance of check-in prediction task of our model in SG, JK and NYC datasets with different values of steepness

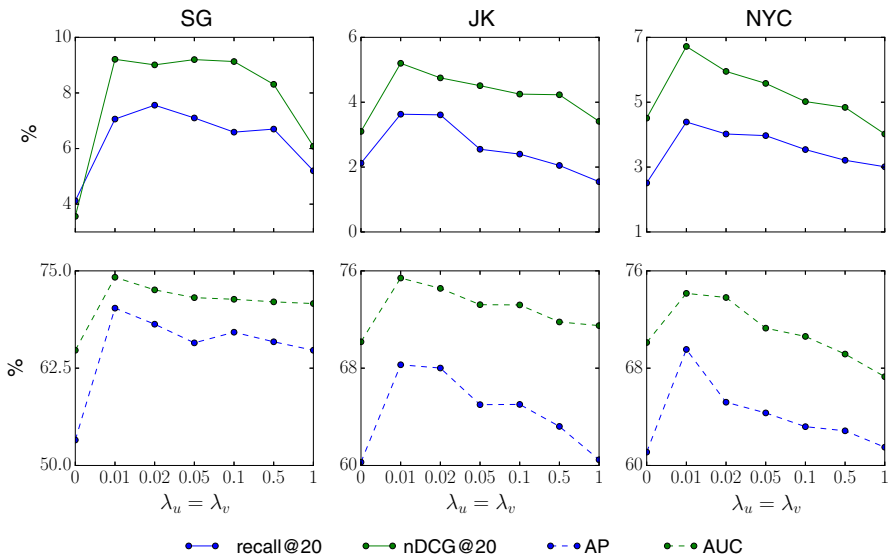


Fig. 7 Performance of check-in prediction task of our model in SG, JK and NYC datasets with different value of regularization parameter

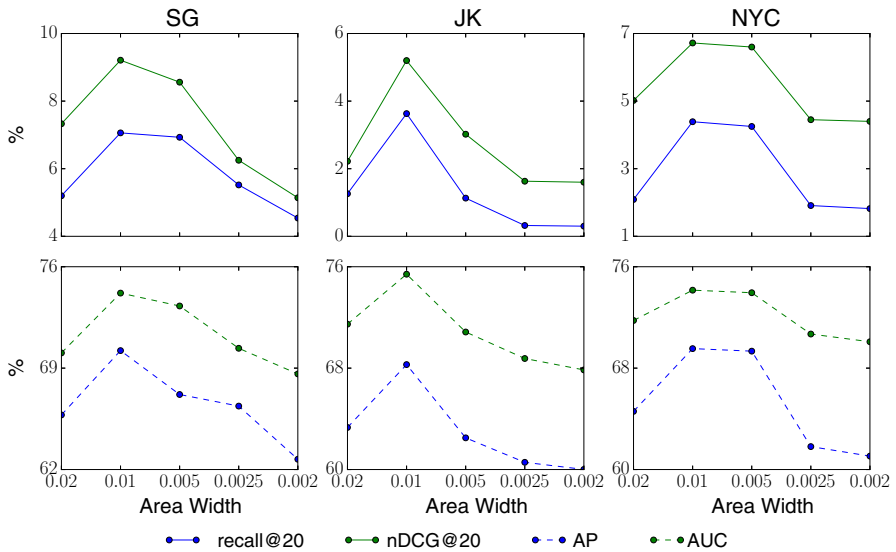


Fig. 8 Performance of check-in prediction task of our model in **SG**, **JK** and **NYC** datasets with different value of area width

results for all the three datasets. This result suggests that our default parameter setting is reasonable.

5.6 Choice of area width

In the earlier experiments, we have adopted a fixed area width setting, i.e. $s = 0.01$. To understand how this setting affect the performance of VAN model, we now vary s between 0.02 and 0.002 while keeping default settings for the remaining parameters.

Result Figure 8 shows very similar performance for **SG**, **JK** and **NYC** datasets. VAN model shows poorer results across different performance measures when $s = 0.02$ but peaks at $s = 0.01$ for the three datasets. Beyond $s = 0.01$, the performance decreases. From the result, we conclude that using $s = 0.01$ yields the best performance. In fact, when s is very small, each area may contain zero or one venue. Hence, the effect of area attraction is eliminated making the prediction less accurate.

5.7 Area boundary shift

In this section, we verify the robustness of our model as we shift the area boundary without changing the area size.

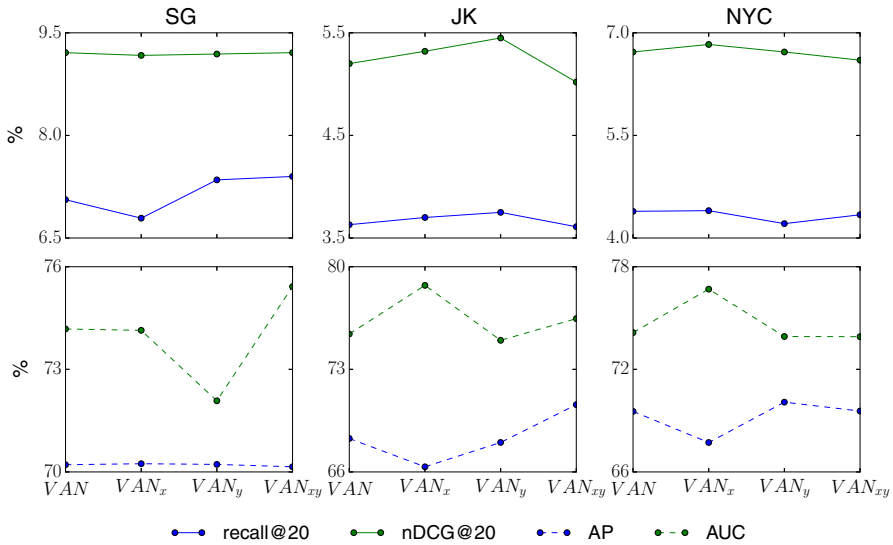


Fig. 9 Performance of check-in prediction task of VAN model with different way of constructing areas in SG, JK and NYC datasets

Parameter setting Recall that we create areas by dividing the city into grid cells of equal width. The boundaries of areas are defined by vertical and horizontal lines sharing the same longitudes and latitudes respectively. Since the choice of these boundary lines can change, we would like to know if shifting the grid cells could affect the performance of VAN model. We use VAN_x and VAN_y to denote our model if grid cells shift 0.005° along latitude and longitude axes respectively. Finally, VAN_{xy} is the model that shifts 0.005° on both latitude and longitude directions. Since the move is one half of the area width, a shift in either direction will lead to the same outcome.

Result Figure 9 shows the prediction result of our models using three area boundary shift settings for SG, JK and NYC datasets. From the result, we observe that the performance difference of VAN_x and VAN_y is less than 5% compared to the one of VAN model. The performance difference between VAN_{xy} and VAN model is 4.6%. Therefore, we conclude that VAN model is robust with different ways of area construction.

5.8 Venue ranking

Other than evaluating models in check-in prediction task, we now compare the ranking of venues derived from the VAN model with some known user provided venue ranking. The purpose is to find out how well VAN model could generate venue ranking similar to user generated venue ranking. We also compare the ranking similarity with that between other baseline models and user generated venue ranking. In this section, the user generated venue ranking comes from Foursquare score. It is a venue specific score derived by aggregating user feedback (e.g. number of likes, dislikes and tips) to the venue.

Parameter setting We use the default parameter setting to evaluate *VAN* in this experiment. Due to our lack of knowledge about local language in **JK** dataset and identifiable information (i.e. the names of venues) regarding check-ins in **NYC** dataset, we only apply this task to the **SG** dataset.

Result In the case of *VAN* model, we compute the score of a venue v : $score_v = \sum_i p_{iv}$. Recall that p_{iv} is the probability of user i interested in venue v ; hence, taking the sum over all users captures the overall interest on venue v . We then rank venues by their $score_v$'s. Table 8 depicts the top 10 venues that returned by *VAN* model. The topmost ranked venue is Changi International Airport which is a world's best airport with more than 50 million passengers per year.⁴ The remaining top venues are prominent shopping malls (e.g. Nex, VivoCity, Jurong Point, AMK Hub and Compass Point), theme parks (e.g. Universal Studios Singapore), immigration checkpoint (e.g. Woodlands Checkpoint) and large education institution (e.g. ITE College East).

Ideally, we want the *VAN* model ranking of venues to be compared against the *Foursquare score*.⁵ However, not all venues in **SG** dataset has *Foursquare* scores. For example, Woodlands Checkpoint and ITE College East venues do not have *Foursquare* score (see Table 8). For this reason, we select only venues whose *Foursquare* scores are available and calculate the Pearson correlation with *VAN*'s venue ranking. The Pearson correlation score of 0.13 suggests that *VAN* has positive correlation with *Foursquare* score. In other words, we can conclude that our ranking is reasonable. To quantify our ranking further, we also calculate the Pearson correlation between other models (PMF and N-MF) and *Foursquare* score. For PMF, the score of each venue j is $score_j^{PMF} = \sum_i U_i V_j$ and for N-MF, $score_j^{N-MF} = \sum_i \hat{R}_{ij}$ where \hat{R}_{ij} is the predicted check-ins between user i and venue j by N-MF. As shown in Table 9, the venue ranking from *VAN* model has the highest Pearson correlation suggesting that it performs better than other baselines by correlation with *Foursquare* score. Table 9 depicts the Jaccard similarity score between top- k ranked venues by *Foursquare* score and those returned by each model. The higher the value of $Jaccard@k$, the more similar the model is to *Foursquare* score. Specifically, suppose s_{FS}^k is the set containing top- k venues by *Foursquare* score and s_x^k is the set of top- k venues by model x . The Jaccard similarity score between them is $Jaccard@k = \frac{|s_{FS}^k \cap s_x^k|}{|s_{FS}^k \cup s_x^k|}$. In our experiment, we choose 20, 50 and 100 as the value of k . From Table 9, we observe that the Jaccard similarity score between *VAN* model and top venues of *Foursquare* score is higher than other baselines. Hence, we conclude that *VAN* model performs better than other baselines in order to rank venues.

5.9 Empirical case examples

Finally, in this section, we present several empirical case examples to illustrate the characteristics of the *VAN* model using the **SG** dataset. For simplicity, we use the default parameter settings to train the *VAN* model. In the first study, we examine the

⁴ <http://www.changiairport.com/content/cag/en/aboutus.html?tab=2017>.

⁵ <https://support.foursquare.com/hc/en-us/articles/201109274-Place-ratings>.

Table 8 Top 10 venues given by VAN model in SG dataset when $a = 2.0, s = 0.01, \lambda_u = \lambda_v = 0.01, \lambda_f = 0$ and the number of latent feature is 10

Rank	Venue name	# Check-in	# check-in users	Foursquare score	score _v
1	Changi International Airport	10,385	5990	9.0	185.01
2	Nex	4899	1716	6.8	113.08
3	VivoCity	5456	2901	8.9	108.05
4	Jurong Point	3814	1272	7.4	98.5
5	AMK Hub	2866	1065	6.7	78.71
6	Universal Studios Singapore	3015	2415	9.3	72.23
7	ITE College East	3065	363	-	67.78
8	Compass Point	2877	706	6.1	62.72
9	Woodlands Checkpoint (Causeway)	3152	1562	-	62.54
10	Cineleisure Orchard	6470	2328	7.8	62.23

Table 9 Pearson correlation and top- k Jaccard coefficient with foursquare venue score ranking

Metric	VAN	PMF	N-MF	
			100 m	200 m
<i>Jaccard@20 (%)</i>	8.1	2.6	2.6	2.6
<i>Jaccard@50 (%)</i>	11.1	2.1	5.3	7.5
<i>Jaccard@100 (%)</i>	14.2	5.3	9.3	8.1
Pearson correlation	0.13	0.07	0.10	0.11

The best performing results are boldfaced

latent factors learned by the VAN model. Each latent factor is represented by the most representative venues. In the second study, we examine the attractiveness of areas derived by the VAN model and compare this with some simple approaches. The final study focuses on showing the competition among venues within each area to win check-ins from users.

Latent factors In the first study, we show the latent factors of the learned VAN model and their most representative venues in Table 10. The most representative venues of a latent factor are those venues v with largest $V_v[t]$ values where V_v is the latent feature vector of venue v and t is the index corresponding to the latent factor. Our findings found several latent factors related to specific location regions or groups of similar type venues. For example, the latent factors 3, 4, 7 and 8 are related to specific location regions. Particularly, latent factor 3 is represented by venues in the east of the city. Latent factors 4 and 7 cover the Orchard and City Hall shopping area respectively. Latent factor 8 is represented by subway stations. Several latent factors are related to different venue types. For example, latent factors 1, 2 and 5 are mainly shopping venues, hotels and night clubs respectively. Latent factor 10 are venues frequently visited by youths. The remaining latent factors 6 and 9 are unfortunately too noisy for interpretation. On the whole, these latent factors appear to carry reasonable meaning reflecting the different types of venues that users may be interested to visit.

Area attraction In the second study, we plot the area attractiveness values derived by the VAN model in Fig. 10a. The attractiveness of an area is derived by aggregating the preference of all users to this area i.e. $\sigma_{a_v} = \sum_{i \in U} \sigma_{a_v}^i$. The larger the attractiveness value, the darker the area is shaded. Figure 10a shows that the high attractive areas are distributed in the downtown area located in the central south of the Singapore island. We now contrast area attractiveness values with area-specific check-in counts and user counts in Fig. 10b, c respectively. In these two figures, we normalize the attractiveness of each area by the maximum attractiveness of all areas. We also apply the similar procedure to normalize the check-in count and user count of each area. We then compute the difference between normalized attractiveness and normalized check-in count (or normalized user count) and assign shade intensity accordingly as shown in Fig. 10b, c respectively. The two figures show that area attractiveness is very different from check-in count and user count in one specific area in the East of Singapore (indicated by dark shaded area in the figures). This area covers Changi airport which is not assigned very high attractiveness value but is known to be highly

Table 10 Top 10 venues of each topic given by VAN model in **SG** dataset with $\alpha = 2.0$, $s = 0.01$, and $f = 10$

Topic 1 Shopping malls	Topic 2 Hotels	Topic 3 East of Singapore	Topic 4 Orchard area
Marina Bay Financial Centre	The Fullerton Hotel	Temasek Polytechnic (TP)	Takashimaya S.C.
Plaza Singapura	Swisstel The Stamford	Changi City Point	313@Somerset
The Shoppes At Marina Bay Sands	National Library Building	Pub Glassy	ION Orchard
The Cathay	Concorde Hotel	Tampines Bus Interchange	The Paragon
Velocity	Bugis MRT Interchange	Nex	Mandarin Orchard
Chinatown Point	Grand Hyatt	St. Gabriel's Secondary School	Chambre de Louie
Great World City	Wisma Atria	Geylang West Community Club	H&M
The Central	Clarke Quay	Bugis Street	Ippudo
United Square	Strand Hotel	Blk 71 Bedok South Road	Spize River Valley
Liang Court	Citylink Mall	Liang Court	Ngee Ann City
Topic 5 Night clubs	Topic 6 Unknown	Topic 7 City Hall area	Topic 8 Locations around subway station
Club V5	313@Somerset	Nanyang Academy of Fine Arts	Marina Square
Helipad	Marina Bay Sands Casino	Marina Square	313@Somerset
Zouk	Kaplan City Campus	Bugis Junction	Cineleisure Orchard
Club Nexus	Cineleisure Orchard	City Hall MRT Station	Golden Village
Cathay Cineplex	Funan DigitalLife Mall	Golden Village	Bugis+
Strictly Pancakes	Novena MRT Station	Sin Thai Him Building	Blk 639 Rowell Road
Liang Court	Starbucks	Raffles City Shopping Centre	Far East Plaza
Playhouse	Clarke Quay	MINK	Plaza Singapura
ZIRCA Mega Club	Zouk	Hotel Ibis	FairPrice Finest
Alfresco Gusto Italian Bistro	Marina Mandarin	Lau Pa Sat Festival Market	City Hall MRT Inter

Table 10 continued

Topic 9 Unknown	Topic 10 Youth-related venues
ION Orchard	Stereo Music Store
Raffles City Shopping Centre	Filmgarde Cineplex
Tanjong Pagar MRT Station	Starbucks
Funan DigitalLife Mall	Volcano Cybercafe
Orchard Central	Bon Riche @ North Br
Fitness First	Orchard MRT Station
Cold Stone Creamery	Plaza Singapura
Planet Paradise Thai Disco	Fitness First
Paris Baguette Caf	*SCAPE Flea Market
Esplanade—Theatres On The Bay	Rebel Boutique Club

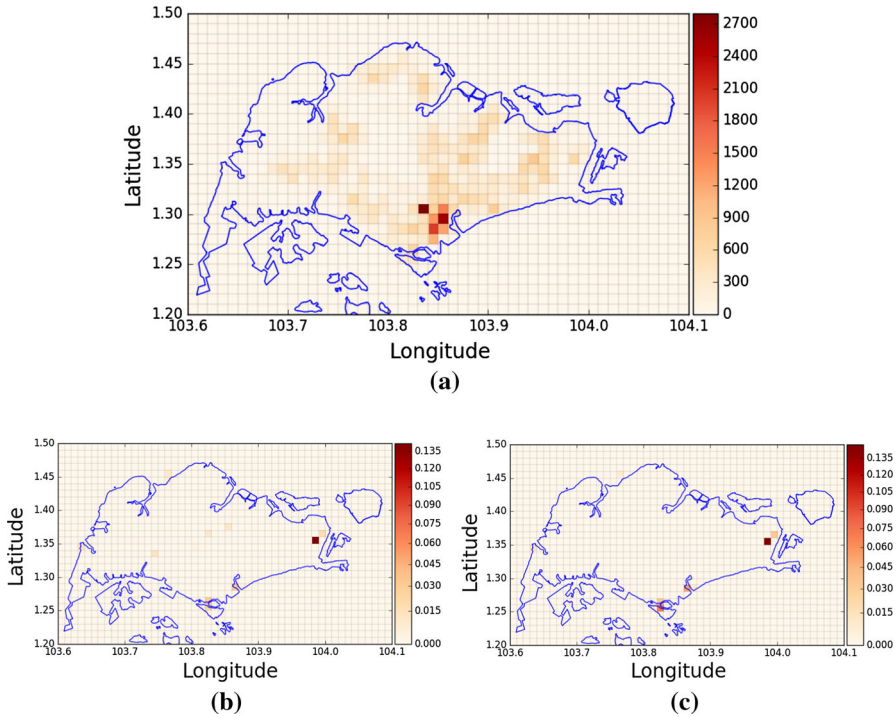


Fig. 10 Heat map of area attractiveness returned by VAN model and its comparison with check-in count and user count using **SG** dataset. **a** Area attractiveness, **b** area attractiveness versus check-in count, **c** area attractiveness versus user count

popular among the tourists and locals. This is a reasonable outcome since most users do not really like the airport and its neighboring venues (they are more likely to visit the airport for the purpose of making overseas trips.), unlike venues in the downtown areas.

Neighborhood competition To show neighborhood competition within an area, this study looks into users selecting the interesting venues in the area to perform check-ins and thus creating competition among the venues. We simplify this analysis by focusing on the most favorite area of each user. The same analysis can also be applied to the less favorite areas.

For a given user i , we divide the venues in his most favorite area into different bins according to the popularity of these venues. The popularity bins cover 1, 2, 3, 4, 5 and above 5 check-ins from all users respectively. Within each bin, user i may perform check-ins on only a subset of venues from the bin. We want to show that the venues gaining the check-ins are more likely the ones winning the interest of user i . In Fig. 11, we thus show the average user interest on these two subsets of venues for each bin of venues sharing the same popularity. The average interest of users on their visited (or unvisited) venues for each bin is computed as $\frac{1}{|U|} \sum_{i \in U} \frac{1}{|\text{bin}_k^i|} \sum_{v \in \text{bin}_k^i} U_i^T V_v$ where

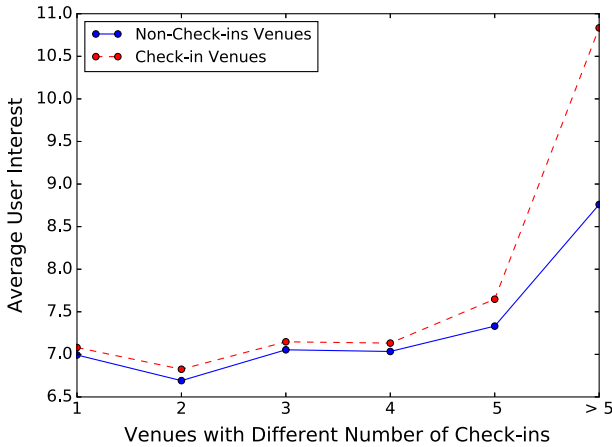


Fig. 11 The correlation of venues with different number of check-ins and the interest of users in their most attractive areas using SG

U is the set of users and bin_k^i is the set of venues with k check-ins such that user i has visited (or not visited) these venues. As shown in the figure, venues which interest users are more likely to be visited than the ones users are not interested given the same popularity.

6 Conclusion and future works

In this paper, we have proposed the VAN model (and its variant VAN_s) that incorporates area attraction, neighborhood competition and social homophily factors. Before introducing VAN model and its inference, we illustrate the existence of these factors through the check-ins datasets from Singapore and Jakarta. Finally, we evaluate our model in check-in prediction task and show that the proposed model yields better performance than baselines. Moreover, we also study the performance of our model via different parameter settings.

VAN model obviously is not perfect and there are still limitations to improve upon. Firstly, in the current VAN model, area size are fixed and pre-defined which may not match the natural urban regions better known to users. Further improvement can therefore be made to VAN model to allow a flexible way to define area. Secondly, VAN model does not cover factors such as venue type, distance effect which says that users usually visit nearby venues than further ones. Thirdly, VAN does not consider the venues to visit based on the time of the day or day of the week. Last but not least, social homophily regularization can have multiple forms such as vector space similarity (VSS) or Pearson correlation coefficient (PCC) (Ma et al. 2011) so we therefore want to apply these forms to understand more about users' movement behaviors. By incorporating the above factors in the future work, we believe a more expressive and accurate model can be produced.

References

- Backstrom L, Sun E, Marlow C (2010) Find me if you can: improving geographical prediction with social and spatial proximity. In: The 19th international conference on World Wide Web (WWW). ACM, New York, pp 61–70
- Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. *J Mach Learn Res* 3(Jan):993–1022
- Blei DM, Jordan MI (2006) Variational inference for Dirichlet process mixtures. *Bayesian Anal* 1(1):121–143
- Boyd S, Vandenberghe L (2004) Convex optimization. Cambridge University Press, New York
- Chang J, Sun E (2011) Location 3: how users share and respond to location-based data on social networking sites. In: 5th international AAAI conference on weblogs and social media (ICWSM), pp 74–80
- Cheng C, Yang H, King I, Lyu MR (2012) Fused matrix factorization with geographical and social influence in location-based social networks. In: The 26th AAAI conference on artificial intelligence (AAAI), vol 12, pp 17–23
- Cho E, Myers SA, Leskovec J (2011) Friendship and mobility: user movement in location-based social networks. In: Seventeenth ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD). ACM, New York, pp 1082–1090
- Church RL, Murray AT (2009) Business site selection, location analysis, and GIS. Wiley Online Library, New York
- De Nadai M, Staiano J, Larcher R, Sebe N, Quercia D, Lepri B (2016) The death and life of great Italian cities: a mobile phone data perspective. In: The 25th international conference on World Wide Web (WWW), pp 413–423
- Doan TN, Lim EP (2016) Attractiveness versus competition: towards a unified model for user visitation. In: The 25th ACM international conference on information and knowledge management (CIKM). ACM, New York, pp 2149–2154
- Doan TN, Lim EP (2017) Modeling check-in behavior with geographical neighborhood influence of venues. In: The 13th international conference on advanced data mining and applications (ADMA)
- Doan TN, Chua FCT, Lim EP (2015a) Mining business competitiveness from user visitation data. In: Eighth international conference on social computing, behavioral-cultural modeling, and prediction (SBP). Springer, Berlin, pp 283–289
- Doan TN, Chua FCT, Lim EP (2015b) On neighborhood effects in location-based social networks. In: IEEE/WIC/ACM international conference on web intelligence and intelligent agent technology (WI-IAT), vol 1. IEEE, Washington, pp 477–484
- Friedman J, Hastie T, Tibshirani R (2001) The elements of statistical learning. Springer series in statistics, vol 1. Springer, New York
- Fu Y, Xiong H, Ge Y, Zheng Y, Yao Z, Zhou ZH (2016) Modeling of geographic dependencies for real estate ranking. *ACM Trans Knowl Discov Data (TKDD)* 11(1):11
- Gao H, Liu H (2015) Mining human mobility in location-based social networks. *Synth Lect Data Min Knowl Discov* 7(2):1–115
- Gao H, Tang J, Liu H (2012a) Exploring social-historical ties on location-based social networks. In: ICWSM
- Gao H, Tang J, Liu H (2012b) gscorr: modeling geo-social correlations for new check-ins on location-based social networks. In: The 21st ACM international conference on information and knowledge management (CIKM). ACM, New York, pp 1582–1586
- Georgiev P, Noulas A, Mascolo C (2014) Where businesses thrive: predicting the impact of the olympic games on local retailers through location-based services data. In: The eighth international AAAI conference on weblogs and social media (ICWSM), AAAI
- Hsu H, Lachenbruch PA (2008) Paired t test. Wiley Encyclopedia of Clinical Trials, New York
- Hu L, Sun A, Liu Y (2014) Your neighbors affect your ratings: on geographical neighborhood influence to rating prediction. In: The 37th international ACM SIGIR conference on research and development in information retrieval (SIGIR). ACM, New York, pp 345–354
- Huff DL (1963) A probabilistic analysis of shopping center trade areas. *Land Econ* 39(1):81–90
- Isaacman S, Becker R, Cáceres R, Martonosi M, Rowland J, Varshavsky A, Willinger W (2012) Human mobility modeling at metropolitan scales. In: The 10th international conference on mobile systems, applications, and services (MobiSys). ACM, New York, pp 239–252
- Jordan MI et al (1995) Why the logistic function? A tutorial discussion on probabilities and neural networks

- Karamshuk D, Noulas A, Scellato S, Nicosia V, Mascolo C (2013) Geo-spotting: mining online location-based services for optimal retail store placement. In: The 19th ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD), pp 793–801
- Koren Y, Bell R, Volinsky C et al (2009) Matrix factorization techniques for recommender systems. *Computer* 42(8):30–37
- Lee DD, Seung HS (2001) Algorithms for non-negative matrix factorization. In: 14th advances in neural information processing systems (NIPS), pp 556–562
- Li H, Ge Y, Hong R, Zhu H (2016) Point-of-interest recommendations: learning potential check-ins from friends. In: The 22nd ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD), pp 975–984
- Li R, Wang S, Deng H, Wang R, Chang KCC (2012) Towards social user profiling: unified and discriminative influence model for inferring home locations. In: The 18th ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD). ACM, New York, pp 1023–1031
- Liang D, Charlin L, McInerney J, Blei DM (2016) Modeling user exposure in recommendation. In: The 25th international conference on World Wide Web (WWW), pp 951–961
- Lin J, Oentaryo R, Lim EP, Vu C, Vu A, Kwee A (2016a) Where is the goldmine? Finding promising business locations through Facebook data analytics. In: The 27th ACM conference on hypertext and social media (HT). ACM, New York, pp 93–102
- Lin J, Oentaryo RJ, Lim EP, Vu C, Vu A, Kwee AT, Prasetyo PK (2016b) A business zone recommender system based on Facebook and urban planning data. In: European conference on information retrieval. Springer, Berlin, pp 641–647
- Liu B, Fu Y, Yao Z, Xiong H (2013) Learning geographical preferences for point-of-interest recommendation. In: The 19th ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD). ACM, New York, pp 1043–1051
- Liu Y, Wei W, Sun A, Miao C (2014) Exploiting geographical neighborhood characteristics for location recommendation. In: The 23rd ACM international conference on information and knowledge management (CIKM). ACM, New York, pp 739–748
- Ma H, Yang H, Lyu MR, King I (2008) Sorec: social recommendation using probabilistic matrix factorization. In: The 17th ACM conference on information and knowledge management (CIKM). ACM, New York, pp 931–940
- Ma H, Zhou D, Liu C, Lyu MR, King I (2011) Recommender systems with social regularization. In: Proceedings of the fourth ACM international conference on web search and data mining. ACM, New York, pp 287–296
- Mnih A, Salakhutdinov RR (2008) Probabilistic matrix factorization. In: The 21th advances in neural information processing systems (NIPS), pp 1257–1264
- Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: machine learning in python. *J Mach Learn Res* 12:2825–2830
- Qu Y, Zhang J (2013) Trade area analysis using user generated mobile location data. In: The 22nd international conference on World Wide Web (WWW). ACM, New York, pp 1053–1064
- Quan X, Wenyan L, Dou W, Xiong H, Ge Y (2012) Link graph analysis for business site selection. *IEEE Comput* 45(3):64–69
- Quercia D, Saez D (2014) Mining urban deprivation from foursquare: implicit crowdsourcing of city land use. *IEEE Pervasive Comput* 13(2):30–36
- Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L (2009) BPR: Bayesian personalized ranking from implicit feedback. In: The 25th conference on uncertainty in artificial intelligence (UAI), pp 452–461
- Schmidt MN, Winther O, Hansen LK (2009) Bayesian non-negative matrix factorization. In: The 8th independent component analysis and signal separation (ICA), vol 9, pp 540–547
- Smarzaro R, Lima TFM, Davis Jr CA (2017a) Could data from location-based social networks be used to support urban planning? In: The 26th international conference on World Wide Web (WWW)
- Smarzaro R, de Lima TFM, Davis Jr CA (2017b) Quality of urban life index from location-based social networks data: a case study in Belo Horizonte, Brazil. In: Volunteered geographic information and the future of geospatial data. IGI Global, Hershey, pp 185–207
- Song C, Qu Z, Blumm N, Barabási AL (2010) Limits of predictability in human mobility. *Science* 327(5968):1018–1021

- Wang C, Blei DM (2011) Collaborative topic modeling for recommending scientific articles. In: The 17th ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD). ACM, New York, pp 448–456
- Weng L, Flammini A, Vespignani A, Menczer F (2012) Competition among memes in a world with limited attention. *Sci Rep* 2:335
- Yan XY, Wang WX, Gao ZY, Lai YC (2017) Universal model of individual and population mobility on diverse spatial scales. *Nat Commun* 8(1):1639
- Yu Z, Zhang D, Yang D (2013) Where is the largest market: ranking areas by popularity from location based social networks. In: Ubiquitous intelligence and computing, 2013 IEEE 10th international conference on and 10th international conference on autonomic and trusted computing (UIC/ATC), pp 157–162
- Yu Z, Tian M, Wang Z, Guo B, Mei T (2016) Shop-type recommendation leveraging the data from social media and location-based services. *ACM Trans Knowl Discov Data (TKDD)* 11(1):1
- Yuan J, Zheng Y, Xie X (2012) Discovering regions of different functions in a city using human mobility and pois. In: The 18th ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD). ACM, New York, pp 186–194
- Zhao S, King I, Lyu MR, Zeng J, Yuan M (2017) Mining business opportunities from location-based social networks. In: The 40th international ACM SIGIR conference on research and development in information retrieval (SIGIR), pp 1037–1040
- Zhao T, McAuley J, King I (2014) Leveraging social connections to improve personalized ranking for collaborative filtering. In: Proceedings of the 23rd ACM international conference on conference on information and knowledge management, CIKM '14

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.