Unsupervised Learning of Disentangled Location Embeddings

Kun Ouyang, Yuxuan Liang, Ye Liu, David S. Rosenblum
School of Computing, National University of Singapore
Singapore
email: {ouyangk, yuxliang, liuye, david}@comp.nus.edu.sg

Wenzhuo Yang
BIGO Technology
Singapore
email: yangwenzhuo08@gmail.com

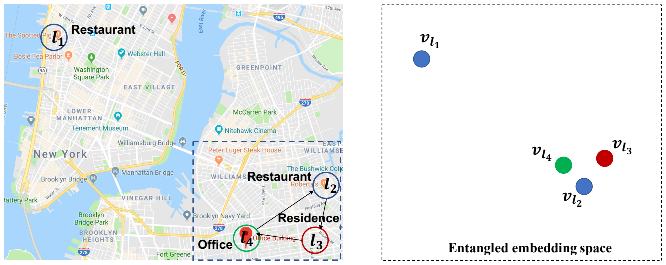
***Abstract*—Learning semantically coherent location embeddings can benefit downstream applications such as human mobility prediction. However, the conflation of geographic and semantic attributes of a location can harm such coherence, especially when semantic labels are not provided for the learning. To resolve this problem, in this paper, we present a novel unsupervised method for learning location embeddings from human trajectories. Our method advances traditional transition-based techniques in two ways: 1) we alleviate the disturbance of geographic attributes on the semantics by disentangling the two spaces; and 2) we incorporate spatio-temporal attributes and regular visiting patterns of trajectories to capture the semantics more accurately. Moreover, we present the first quantitative evaluation on location embeddings by introducing an original query-based metric, and we apply the metric in experiments on two Foursquare datasets, which demonstrate the improvement our model achieves on semantic coherence. We further apply the learned embeddings to two downstream applications, namely next point-of-interest recommendation and trajectory verification. Empirical results demonstrate the advantages of the disentangled embeddings over four state-of-the-art unsupervised location embedding methods.**

***Index Terms*—Representation Learning, Point of Interest**

I. Introduction

The increasing availability of smart devices enables organizations and ISPs to collect massive amounts of human trajectory data, which can support a variety of location-based services, such as point-of-interest (POI) recommendation and mobility prediction. To support such services, one needs to model the locations traversed by users with an appropriate representation (i.e., embedding) in order to facilitate subsequent algorithmic computations. Appropriate location embeddings can achieve broad generalizability and thus benefit multiple downstream applications [1, 2, 3, 4]. Constructing generalizable location embeddings in practice, however, is a challenging problem due to the following reasons:

1) **Limited semantic labels.** The semantics of a location closely correlate with the function of the location. For example, users go to restaurants for food, their offices for work, and their residences for rest, where food, work, and rest can be regarded as the semantics of the respective locations. Location semantics are indispensable in many applications such as mobility prediction, as they indicate the intentions of users. Therefore, location embeddings lacking such semantics limit generalizability to downstream applications. Nonetheless, in many situations, these semantics are not provided when trajectory data are collected (e.g., see Shokri et al. [5]), and manually labeling the functions of locations is highly expensive due to the massive number of locations traversed by users. This limitation prevents the use of supervised embedding techniques that require a large amount of semantic labels.
2) **Discrepancy between semantic and geographic spaces.** A location exhibitsboth semantic and geographic attributes. Taking Figure 1(a) as an example, location l_1 and l_2 are semantically similar (i.e., both are restaurants), but geographically distant from each other. This would induce a discrepancy between the semantic space and the geographic space of the corresponding location representations. Existing studies have failed to account for this discrepancy between the two spaces and instead use a conflated vector to embed both attributes of the location. However, this inevitably results in an entangled hidden space since the geographic attributes *disturb* the semantic attributes of the embedding, which can mislead downstream applications about user intentions and thereby reduce their effectiveness.



(a) (b)

Fig. 1: Illustration of the semantic and geographic attributes of locations (a) and an entangled embedding space (b). See Section III-A for more details.

To resolve the challenges described above, in this paper we present a novel *unsupervised* learning model called *D̲isentangled S̲kip-G̲ram* (DXG) to construct disentangled location embeddings. Our main goal is to improve semantic coherence in learning location embeddings in an unsupervised setting, to provide better generalizability for downstream applications. Note that even though we are unable to identify the *explicit* semantics (i.e., labels) due to the unsupervised nature of the learning, the coherence of the semantic space already reflects the *latent* semantics: Embeddings with similar semantics are close to each other and are far apart otherwise.

More specifically, we present three intuitions that we exploit in the design of DXG, which are derived from the analysis of real-world check-in sequences. The intuitions uncover the underlying mechanism for how geographic factors disturb semantic attributes. To resolve such disturbance, we model the semantic part separately from the geographic part of the location representation to untangle the two subspaces. For the semantic part, we exploit regular visit patterns as a proxy for the explicit semantics of a location. For the geographic part, we employ a manifold-learning scheme to retain the local structure of the Euclidean space of locations in the embedded geographic space. Moreover, we leverage sequential mobility information to unite the two parts, from which we are able to formulate an accurate transition likelihood with spatio-temporal attributes incorporated.

In addition, we present an original query-based metric in order to quantify the semantic coherence of the learned embeddings by measuring the local purity. Using this metric, we compare our model with several state-of-the-art methods for learning location representations, including PRME [6], GE [7], POI2Vec [8], and the naive Skip-Gram, based on two open datasets from Foursquare namely NYC and TKY. Empirical results demonstrate that our embedding mechanism outperforms the baselines in preserving semantic coherence. To validate the effectiveness of learned embeddings in downstream tasks, we present results from two additional case studies involving the traditional next POI recommendation task and a novel task regarding trajectory verification. Experimental results demonstrate that embeddings generated by DXG consistently outperform other baselines.

Our key contributions are summarized as follows:

- We present Disentangled Skip-Gram, a novel unsupervised learning model that captures latent location semantics in the embedding space by structurally modeling the spatio-temporal transitions between locations within human trajectories, and by disentangling the semantic and geographic subspaces. Disentanglement provides a further benefit in that downstream tasks can adjust the weights on the geographic and semantic information when needed.
- We present the *first* quantitative investigation of the embedding space of location representations across multiple embedding methods using an original query-based metric that provides a systematic tool for evaluating the quality of the learned location embeddings.
- We evaluate the learned embeddings on two downstream location-based applications using two open trajectory datasets. Empirical results demonstrate the efficacy and superiority of our model.

II. Preliminaries

A. Notation

Let $\mathcal{L} = \{l_1, l_1 \ldots, l_N\}$ be the set of locations in some area of interest and $\mathbf{T}raj$ be a set of trajectories, where each $Traj \in \mathbf{T}raj$ is a sequence $\{(l_1, \tau_1, y_{l_1}), \ldots, (l_m, \tau_m, y_{l_m})\}$. Each tuple (l, τ, y_l) denotes a visit at time τ on a location l with the semantic label y_l. Given a specific time τ, we discretize it into the corresponding hour of the day $t \in \mathcal{T}$, where $\mathcal{T}$ is the set of all hour-wise time intervals.

B. Skip-Gram

Given a sequence of items (e.g., words or locations), Skip-Gram [9] employs a sliding window containing a *current item* and some *context items* to construct representations. The core intuition of Skip-Gram is to reconstruct the transitional relationships between items according to the similarities between their respective representations. The objective of Skip-Gram is to maximize the transition likelihood that the current item l predicts its context items l_c, which can be modeled as

$$O_{Skip-Gram} = \max \prod_{(l,\tau,y_l)\sim\mathbf{T}raj} \prod_{l_c\sim context(l)} p(l_c|l) \quad (1)$$

where the conditional likelihood $p(l_c|l)$ is defined as $p(l_c|l) = e^{<v_l, v_{l_c}>}/Z$, with the normalization term $Z = \sum_{v_{l'} \in \mathcal{V}} e^{<v_{l'}, v_l>}$. v denoting a vector representation, and $< \cdot, \cdot >$ denoting vector inner product.

III. Method

A. Intuitions

(1) Spatio-Temporal Skip-Gram. Following other works in the literature [7, 8, 2], we apply the Skip-Gram model to learn embeddings according to the transitional patterns between locations. The rationale can be drawn from the analogy between sentences in natural language and trajectories in human mobility: just as a word can be inferred from its context [9], a location can be recognized by the locations from which users arrive and to which they depart. However, different from related works, we enhance the basic Skip-Gram model by considering two spatio-temporal attributes that are specific in trajectory transitions: the time interval and the geographic distance. The intuition is that generally a transition is more likely to happen in the near future and to nearby places rather than to distant places or after a long time interval. To describe the transition likelihood more accurately, we incorporate this intuition to reformulate the objective function in Equation 1.

(2) Regular Visiting Patterns. People usually perform regular visiting patterns across different days. For example, restaurants are mainly visited at lunchtime and clubs at night, regardless of the date. To confirm these patterns in real data, we conducted the quantitative analysis of the Foursquare dataset [10] presented in Figure 2. It can be seen that locations with different semantics induce very distinct visiting patterns, which implies a strong correlation between the explicit semantics and the visiting pattern of a location. Such correlations enable us to use the visiting pattern as a proxy for the actual location semantics,

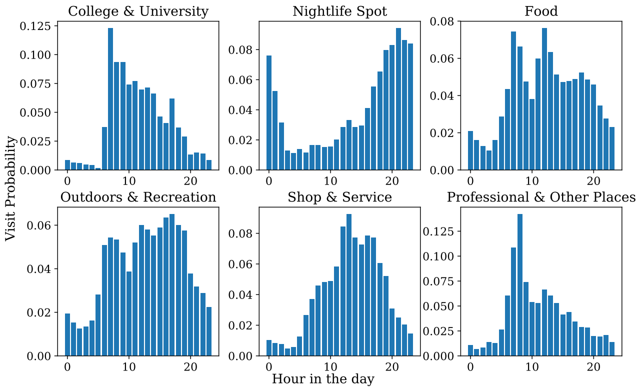


Fig. 2: Visiting distributions averaged over different days at locations with different semantics.

providing extra information to enhance the coherence of the semantic spaces when semantic labels are not available.

(3) Entangled Semantic Space and the Travel Hub Problem. Using a conflated embedding as in traditional methods can provoke an entangled embedding space, especially in unsupervised settings. Using transition-based models such as Skip-Gram might mitigate such discrepancies to some extent, as the training procedure of Skip-Gram actually performs information propagation between items. That is, the transitions between neighbor items provide paths for direct "message passing". As contextual items also connect with their own related contextual items, the message could pass even further to distant items. Assuming the set of training trajectories is ideally large, then distant locations could have a chance (though small) to be connected via an information path (though weak), such as l_1 and l_2 in fig:1. However, this assumption is generally impractical, for two reasons: 1) the large number of locations and cost to collect a large trajectory dataset; and 2) users' tendency to travel locally [10] (e.g., in fig:1 users tend to travel within the bottom right region but not to the distant l_1). This results in a phenomenon called a *travel hub*.

To confirm the travel hub problem, we analyzed the TKY and NYC datasets from Foursquare [10]. Specifically, we first formulated travel hubs as the isolated connected components from the global transition graph. Then we constructed a location-location transition graph $G = (\mathcal{L}, \mathcal{E})$ according to the check-in sequences, where $\mathcal{E}$ is the set of edges between locations. For any consecutive check-in pair $\{(l_i, \tau_i), (l_j, \tau_j) | \tau_i < \tau_j\}$ in a check-in trajectory, if the interval $\delta = (\tau_j - \tau_i) < \Delta T$ for a predefined threshold ΔT, we consider it to be a valid pair and add an edge $e_{i,j}$ between nodes l_i and l_j. The edge weight $e_{i,j}$ is the number of times l_i and l_j are observed as a valid check-in pair. Given a constructed graph G, we computed the corresponding connected components. Table I presents the results of this analysis, with ΔT set to 6 hours.

Component Cardinality	1	2	3	4	5	6	29992	47987
# Components (NYC)	7520	345	32	6	1	1	1	0
# Components (TKY)	12933	413	31	1	3	0	0	1

TABLE I: Transition graph analysis with Foursquare data.

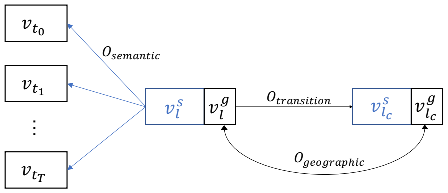


Fig. 3: The structure of our model.

Table I shows that transition graphs from both datasets are not fully connected, and thus many isolated components can be observed. The existence of these isolated components thus confirms the existence of travel hubs.

Take fig:1 as an example. If we simply apply the Skip-Gram scheme, the embedding v_{l_2} will be trained to be close to v_{l_3} and v_{l_4} since a sequence can be observed among l_2, l_3 and l_4. However, as shown in Figure 1(b), v_{l_2} will be separated from v_{l_1}, since there exists no observed sequence between l_1 and the hub region, even though l_1 has semantics similar to l_2's. Thus the information from the hub area cannot propagate to l_1 during training. As a result, this would undermine the semantic coherence of the learned embeddings.

Overall, the geographic attributes of a location influence the semantic purity of its embedding. Therefore, we aim to alleviate such geographic disturbance by partitioning the embedding space into semantic and geographic subspaces, in order to achieve better semantic coherence.

B. The DXG Model

Our model architecture consists of three training objectives, as depicted in Figure 3. According to our *third intuition* described above, we first split each embedding $v_l \in \mathcal{R}^L$ into two parts, v_l^s and v_l^g, where the first s entries encode the semantic information of location l and the last $g = L - s$ entries encode the geographic information.

1) Semantic Part: For the semantic part v^s, we assume it can be approximately learned through the regular visiting pattern based on our *second intuition*. Given a visit tuple (l, τ), we convert τ to its respective hour slot t and formulate the probability that l is visited at time τ as $p(\tau|l) = \frac{e^{<v_t, v_l^s>}}{\sum_{t' \in T} e^{<v_{t'}, v_l^s>}}$, where $v_t \in \mathcal{R}^L$ is the vector representation at the t-th time slot. The semantic objective function is then defined as

$$O_{semantic} = \max \prod_{\tau, l} p(\tau|l), \qquad (2)$$

which models the correlation between the locations and the corresponding visiting time.

2) Geographic Part: For the geographic part, we aim to learn embeddings that preserve the local structure of locations in the Euclidean space. One way is to simply use the Cartesian coordinates as the embedding, but that approach has two major problems: 1) the vector is hardcoded instead of learnable, which can induce inconsistency in the model; and 2) more importantly, we aim to use a consistent arithmetic operation for computing the distances between vectors (i.e., the cosine similarities). We therefore use a trainable vector in each embedding to correlate the Euclidean distances $eud(.,.)$ of locations in the geographic space with their cosine similarities $cos(.,.)$ in the embedding space, such that the inner product $< v_{l_i}, v_{l_j} >$ can encode the geographic closeness between location l_i and l_j. To achieve this, we design the following geographic objective function:

$$O_{geographic} = \min \sum_{i,j} D[f_{cos}(v^g_{l_i}, v^g_{l_j}), f_{eud}(l_i, l_j)]. \quad (3)$$

$O_{geographic}$ represents the cost of aligning the two spaces (Euclidean distance space and Cosine similarity space), with a divergence function D measuring the distance between pairwise correlations in the two spaces. In order to bridge the two spaces effectively, we require three common properties of f_{cos} and f_{eud}, namely that they should have 1) the same output space (which can be either similarity or distance), 2) the same range, and 3) the same monotonicity.

As the distance between two locations usually exhibits a long-tail distribution [11, 12], we use the reciprocal of the exponential function in f_{eud} to perform normalization, with a coefficient ϕ_G to adjust its slope:

$$f_{eud} = e^{-\phi_G(l_i, l_j)} \quad (4)$$

Note that f_{eud} computes *Euclidean similarity* due to the reciprocal form. We then formulate f_{cos} as

$$f_{cos} = \frac{1}{2}(1 + cos(v^g_{l_j}, v^g_{l_j})), \quad (5)$$

which satisfies the three properties mentioned above. By using f_{eud} and f_{cos}, we normalize both cosine and Euclidean similarity to the range $[0, 1]$, effectively making them the probability that two points are neighbors in their respective spaces. To measure the distance between the distributional manifolds formed by f_{eud} and f_{cos}, we use *cross entropy* as the divergence function D and reformulate Equation 3 as

$$\begin{aligned} O_{geographic} = \min \sum_{i,j} & -f_{eud}(l_i, l_j) \log f_{cos}(v^g_{l_i}, v^g_{l_j}) \\ & - (1 - f_{eud}(l_i, l_j)) \log(1 - f_{cos}(v^g_{l_i}, v^g_{l_j})). \quad (6) \end{aligned}$$

KL-Divergence [13] is an alternative for D, differing from cross entropy with only an additive constant, but in this work we use cross entropy for simplicity.

3) The United Embedding: In a human trajectory, a location is visited due to both of its semantic attributes and geographic attributes. Therefore, we need to combine v^s_l and v^g_l to form a comprehensive representation of the respective location $v_l = [v^s_l; v^g_l]$, and to accurately formulate the transition likelihood in trajectories. As per our *first intuition*, we define the spatio-temporal Skip-Gram likelihood function as follows:

$$O_{transition} = \max \prod_{l \in Traj} \prod_{l_c \in context(l)} p(l_c|l)^{w_\Delta}, \quad (7)$$

where the exponential decay factor $w_\Delta = e^{-\phi_\Delta ||\tau_{l_c} - \tau_l||_1}$ indicates that the larger the time interval between two visits, the less likely the current visit is able to predict the context visit. ϕ_Δ is the temperature term that controls the sensitivity of the time interval. Note that the temporal impact is accounted for by w_Δ, while the spatial impact has been considered by design into the transition likelihood. Recall that $p(l_c|l) = e^{<v_l, v_{l_c}>}/Z$. Since

$$< v_l, v_{l_c} >=< v^s_l, v^s_{l_c} > + < v^g_l, v^g_{l_c} > \quad (8)$$

where the inner product $< v^g_l, v^g_{l_c} >$ is determined by the geographic distance between l and l_c according to the objective function (6), the resulting transition likelihood is also determined by this geographic distance.

So far we have incorporated all three intuitions into DXG, a comprehensive spatio-temporal model that disentangles the semantic and geographic subspaces. Recall l_1 and l_2 in fig:1. Since we model the semantic and geographic parts separately, v_{l_1} will remain distant from v_{l_2} because $v^g_{l_1}$ and $v^g_{l_2}$ are distant, while $v^s_{l_1}$ and $v^s_{l_2}$ can remain close since they share the similar semantics and exhibit similar regular visiting patterns.

4) Optimization: We jointly optimize the three objectives of DXG, with $O_{geographic}$ plus the logarithms of $O_{semantic}$ and $O_{transition}$. The overall training objective is defined as:

$$\begin{aligned} V &= \arg\max_V \{\log O_{semantic} - O_{geographic} + \log O_{transition}\} \\ &= \arg\max_V \{\sum_{\tau,l} \log p(\tau|l) \\ &\quad - \sum_{i,j} D[f_{cos}(v^g_{l_i}, v^g_{l_j}), f_{eud}(l_i, l_j)] \\ &\quad + \sum_{l \in Traj} \sum_{c \in context(l)} w_\Delta \log p(l_c|l)\} \end{aligned} \quad (9)$$

where $V = \{\mathcal{V}_L, \mathcal{V}_T\}$ denotes the set of all location embeddings plus all time slot embeddings.

Note that directly computing the normalization term Z in eq:skipgram is very expensive due to the large cardinality of the set of locations to be normalized. Therefore, we replace the regular softmax function with the sampled softmax proposed by Jean et al. [14] to accelerate the training process. We choose a negative sample size $N_{neg} = 64 << N$ for negative sampling in all experiments. During the training procedure, we use the gradient descent optimizer Adam [15] to search the optimal embeddings and renormalize the embedding vector to be a unit vector to further boost training speed. Our pseudocode is provided in the supplementary materials[1].

IV. Evaluation

In this section, we present a novel query-based metric to quantify semantic coherence in the semantic space, and an empirical evaluation of the metric on embeddings learned by DXG and several baselines. We then present results from case studies on a traditional task (next POI recommendation) and a novel task (trajectory verification) to evaluate the learned embeddings under both non-parametric and parametric scenarios.

[1] https://www.dropbox.com/s/towfdwhjdkdhyeo/supplementary.pdf

A. Query-Based Metric

Existing works [16, 1] use visualization tools such as t-SNE [13] to judge learned location embeddings via eyeballing, which is imprecise and subjective. To improve objectivity, traditional metrics for objective evaluation of the quality of unsupervised embeddings can be used. These metrics usually involve a clustering task using off-the-shelf clustering algorithms to examine the purity of the formed clusters with respect to given ground-truth labels [17]. However, the results can be biased by the clustering algorithms, since different algorithms favor different geometric structures of the data. We therefore designed a novel query-based metric that provides a quantitative measure of the coherence in the semantic space by examining the local purity of the neighbors of the embeddings.

Given some set of location queries $Q = \{l_0, \dots, l_M\}$, we first project each query location l into the semantic space v_l and then compute the cosine similarity $s(l, l_j)$ between l and all other embeddings l_j: $s(l, l_j) = \frac{<v_l, v_{l_j}>}{|v_l||v_{l_j}|}$, $\forall l_j \in \mathcal{L} \backslash l$. We rank l_j according to the cosine similarity scores and select the Top-K locations, which form the level-K Nearest Neighbor set of l, $NN@K(l)$. Given ground truth semantic labels $\{y_l | l \in \mathcal{L}\}$, we can treat this as a Top-K ranking task [18], which allows us to compute the aggregate scores $\mathit{Accuracy}@K$, $\mathit{Recall}@K$, $\mathit{Precision}@K$, and $F_1\mathit{score}@K$ on $NN@K(l)$, for each query location l and a given K level. The aggregate scores are computed as follows:

$$\begin{aligned}
\mathit{Accuracy}@K &= \frac{1}{M} \sum_{l_i \in Q} |\mathbb{1}\{y_{l_i} \in \{y_{l_j} | l_j \in NN@K(l)\}\}| \\
\mathit{Precision}@K &= \frac{1}{M} \sum_{l_i \in Q} \frac{|\mathbb{1}\{y_{l_i} = y_{l_j}\} | l_j \in NN@K(l)|}{K} \\
\mathit{Recall}@K &= \frac{1}{M} \sum_{l_i \in Q} \frac{|\mathbb{1}\{y_{l_i} = y_{l_j}\} | l_j \in NN@K(l)|}{|\mathbb{1}\{y_{l_i} = y_{l_j}\} | l_j \in \mathcal{L}|} \\
F_1\mathit{score}@K &= 2 \times \frac{\mathit{Precision}@K \times \mathit{Recall}@K}{\mathit{Precision}@K + \mathit{Recall}@K}
\end{aligned} \tag{10}$$

The aggregate scores over all queries reflect the local closeness of embeddings having similar semantics, and thus can be used as a tool to evaluate the semantic coherence of the learned embedding space. We use M=N in our experiments to evaluate embeddings of all locations.

B. Model Selection

Model selection (i.e., validation) in unsupervised settings is an important yet challenging task. In DXG, the key hyperparameters are the sizes of the semantic part s and the geographic part g. Typical selection methods requiring ground truth labels (e.g., AIC, BIC, and cross-validation) are not suitable for our unsupervised setting. We therefore heuristically treat the learning of the semantic embedding as an implicit clustering process. After performing preliminary studies[2], we employ the average closeness of the K-nearest neighbors across all points as the hyperparameter selection criterion.

C. Experimental Settings

1) Dataset: We conducted our experiments with two publicly available datasets, which contain check-ins in New York City (NYC) and Tokyo (TKY) collected from Foursquare [10]. Each data record contains a user ID, POI ID, check-in timestamp, GPS coordinates of the POI and semantic label (e.g., "park"). We consider locations with more than five check-in records as valid entities and label the remaining locations as "UNK". We also remove trajectories of length less than two. From this preprocessed data (see Table II for statistics), we use the first 80% of the trajectories of each user as the training set and the remaining 20% as the test set. Since our target is unsupervised learning for location embeddings, we ignore the semantic labels in the training set and use this set to train location embeddings and the parametric model in the trajectory verification task. The test set with semantic labels is used for the query-based metric for evaluation of semantic coherence and for the application case studies.

Dataset	#Users	#POIs	#Semantic Categories	#POIs per category
NYC	1071	5342	249	112.28
TKY	2242	9541	240	190.29

TABLE II: Statistics of the preprocessed datasets.

2) Hyperparameters: We set the embedding dimension to $L = 100$ for all models. As per the model selection criterion of Section IV-B, we set the dimension of the semantic part in DXG to be $s = 75$ and the geographic part to be $g = 25$ for the NYC dataset, and $s = 50$ and $g = 50$ for the TKY dataset. The temperature terms are set as $\phi_\Delta = 0.001$ and $\phi_G = 50$. To guarantee convergence, we train all models for 40 epochs with batch size 256. The learning rate is set to 0.001.

3) Baseline Models: We use the following four unsupervised models as strong baselines for our evaluation. We experimented with various hyperparameter settings for the baselines and then used the optimal settings for the baselines.

- **Skip-Gram**. Due to its simplicity and popularity in NLP, many recent studies [1, 4, 19] directly use the embeddings learned from the Skip-Gram model.
- **PRME-G**. PRME-G [6] stands for the Personalized Ranking Metric, and it considers the embeddings in both the transition space and the user ranking space. It also considers the temporal impacts within a temporal threshold. However, it does not consider geographic impacts.
- **GE**. GE [7] is a graph embedding method that is an extension of LINE [20]. It constructs POI-POI, POI-region, POI-time slot and POI-word graphs and applies the network embedding scheme as LINE does. Since our focus is on unsupervised learning, we discard the POI-word graphs while keeping the others.
- **POI2Vec**. POI2Vec [8] incorporates geographic distances between POIs into the hierarchical softmax function of the classic CBOW [9] model by dividing POIs into sub-regions. No temporal effects are considered.

[2] Due to space limitations, please refer to the supplementary materials (see footnote 1) for more detail.

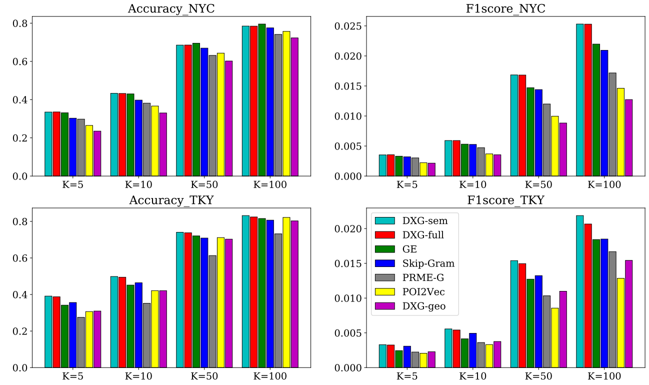


Fig. 4: Semantic coherence measured by the query-based metric. We report only the accuracy and F_1-score due to space limitations. All graph share the same legend.

	Acc.@100		Prec.@100		Rec.@100		F_1@100	
DXG\geo	0.79	0.83	7.41	7.34	1.49	1.24	2.49	2.13
DXG\sem	0.78	0.81	6.00	6.47	1.37	1.15	2.23	1.96
DXG\tran	0.73	0.79	5.46	6.12	0.85	0.77	1.47	1.36
DXG\w_Δ	0.79	0.83	7.55	7.35	1.54	1.25	2.55	2.13
DXG	**0.79**	**0.83**	**7.81**	**7.46**	**1.55**	**1.25**	**2.58**	**2.15**

TABLE III: The ablation experiment results. Results of NYC are reported on the left and TKY on the right in each column. The precision, recall, and F_1 score values are times 10^{-2}.

In order to understand the semantics captured by different parts of the embeddings, we also compare three variants of the trained embeddings: **DXG-full** uses the whole embedding vector; **DXG-sem** uses only the semantic part from **DXG-full**; and **DXG-geo** uses only the geographic part from **DXG-full**.

D. Semantic Coherence

1) Model Comparison: Figure 4 depicts the accuracy and F_1 score from applying the query-based metric to all models. Accuracy is similar among all models since it is a coarse-grain metric. However, for the F_1 score, DXG-sem achieves the highest among all models. For instance, for F_1 score@100, it outperforms GE, POI2Vec, PRME-G, DXG-full, DXG-geo and Skip-Gram by 15%, 73%, 47%, 0.1%, 99%, 21%, respectively, on the NYC dataset and by 18%, 70%, 31%, 5.8%, 41%, 18%, respectively, on the TKY dataset. As would be expected, DXG-geo performs the worst, since it captures geographic information only and no semantics. Notice that despite its simplicity, Skip-Gram performs close to or even outperforms the other baselines, demonstrating that the transitions between visits does reveal the intrinsic semantics of the locations.

2) Variant Comparison: In order to understand the impact of different components of DXG, we conducted ablation experiments. We use DXG\geo, DXG\sem, DXG\tran, DXG\w_Δ to indicate the deprivation of geographic loss, semantic loss, transition loss and temporal decay, respectively. We measured the performance by evaluating the semantic part only. The results in Table III demonstrate that the model with full setting learns the best semantic space, which confirms our intuition that a proper location embedding model should incorporate all three of our intuitions. Of all the components, the deprivation of the transition loss degrades the performance the most, followed by the semantic loss. The time decay factor w_Δ plays the least role in the model.

	Acc.@100		Prec.@100		Rec.@100		F_1@100	
STSG-1	0.69	0.75	2.86	2.81	0.51	0.34	0.86	0.61
STSG-25	0.79	0.82	6.99	7.46	1.25	1.22	2.13	2.10
STSG-50	0.77	**0.83**	7.38	**7.46**	1.36	**1.25**	2.29	**2.15**
STSG-75	**0.79**	0.82	**7.81**	6.91	**1.55**	1.21	**2.58**	2.06
STSG-100	0.77	0.82	7.61	6.78	1.52.	1.19	2.53	2.02

TABLE IV: The dimension sensitivity experiment results. NYC results are presented to the left and TKY to the right in each column, and the precision, recall and F_1-score values are times 10^{-2}.

3) Effects of dimensions s versus g: To understand the trade-off between dimensions of the geographic part and semantic part of the embedding, we set different sizes for s to $[1, 25, 50, 75, 100]$ and set $g = 100 - s$. When $s = 1$, the model learns mostly from the geographic information, and when $s = 100$ there is no geographic influence on the model. The results in Table IV demonstrate that to achieve the best performance, we need to maintain a relative trade-off between the effect of the semantic part and the geographic part. The main reason why different dataset favors different settings is that the geographic structure of Tokyo and New York City are different, including the scale of their urban areas and density of their POI distributions. Elements such as cultural factors are possible additional factors that influence the mobility patterns of people, which also impacts the observed transition patterns.

V. Case Studies

A. Next POI Recommandation

1) Description: This case study examines the learned location embeddings under a non-parametric setting. We follow a recommendation protocol similar to that of [7] to evaluate the effectiveness of learned embeddings in the next POI recommendation task. Given a visit history up to time τ, Xie et al. model the user preference $\overrightarrow{u_\tau}$ as a weighted average of the visited locations up to τ, where the weight is computed using an exponential decay according to the time difference between a previous visit τ_i and τ. Formally,

$$\overrightarrow{u_\tau} = \sum_{\{i:\tau_i<\tau\}} e^{-(\tau-\tau_i)} \cdot \overrightarrow{v_{l_i^u}}, \tag{11}$$

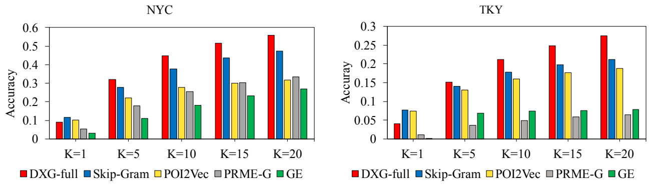
where l_i^u denotes a visit of user u at location l and time τ_i.

Given a query $q = (l_u, \tau)$, we first convert τ to its corresponding time slot t. Then the likelihood s that the user will visit a specific location l is computed as:

$$s_l(q) = \overrightarrow{u_\tau} \cdot v_l + v^g_{l^u_\tau} \cdot v^g_l + v_t \cdot v^s_l. \tag{12}$$

Once we obtain the likelihood scores over all candidate locations $S = \{s_l | \forall l \in \mathcal{L}\}$, we sort the scores from largest to smallest and then recommend the top-K locations. We conducted experiments on the test sets of Section IV-C and used accuracy as the evaluation metric.

2) Evaluation: The accuracy results reported in Figure 5 demonstrate that DXG outperforms all baselines. For instance, at K=20 on the NYC dataset, DXG outperforms Skip-Gram, POI2Vec, PRME and GE by by 18%, 67%, 75% and 106%, respectively. Note that at K=1, the accuracies of all models are just too low to be informative (similar to random), which can be attributed to the simple form of the non-parametric protocol. But results with larger K (5 to 20) demonstrate that embeddings generated by DXG significantly outperform those of the other baseline models.



(a) Accuracies in NYC. (b) Accuracies in TKY.

Fig. 5: Result from the next POI recommendation case study.

B. Trajectory Verification

1) Description: The task of trajectory verification is to determine whether two trajectories $traj_a$ and $traj_b$ were produced by the same person. We formalize this as a classification problem, where the aim is to train a binary classifier $f_\theta(traj_a, traj_b) \in \{0, 1\}$ that takes as input a pair of trajectories and outputs 1 if the pair is *genuine* (i.e., generated by the same person) and 0 otherwise. In privacy research, this task is used to evaluate the anonymity afforded by a synthesized trajectory $traj_b$ based on an actual trajectory $traj_a$.

For this task, we adopt the Siamese architecture widely used for face recognition [21] (see fig:Siamese). Specifically, given a trajectory $traj = \{l_0, l_1, \ldots, l_t\}$, we first project the associated locations to the vector space using the learned embeddings from our model and other baselines, and then encode the whole trajectory using a neural network-based encoder. We finally compute the probability of having a genuine pair from the distance between the trajectory's latent codes. To concretely relate the performance of the verification model to the quality of the learned embeddings, we fix the embedding vectors and use a simple model architecture containing a one-layer RNN with 100 hidden units, which is a small number compared to the massive amount of data. Better embeddings that capture more useful features would, therefore, outperform others after the convergence of the training process.

2) Evaluation: For data preparation, we follow a scheme similar to that of Cho et al. [22].[3] For evaluation, in tb:verification we report both average accuracy and F_1-score for five runs with different initializations. It can be seen that our DXG models outperform the others. For instance, for the F_1-score, our models relatively outperform Skip-Gram, PRME, GE and POI2Vec by 6.5%, 37.3%, 21%, and 8%, respectively, on the NYC dataset and 11.3%, 22.9%, 22.9%, and 20.4% respectively, on the TKY dataset. An important by-product of our disentanglement is that it allows the parametric model to weight the semantic and geographic information differently, which is why the best models in these experiments are different.

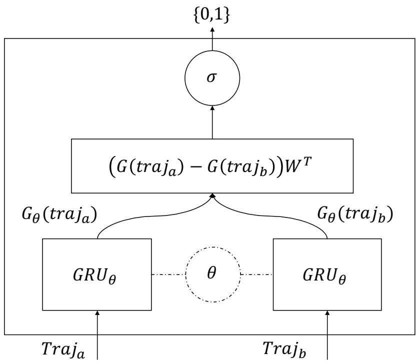


Fig. 6: The Siamese architecture. Two trajectories are encoded to hidden features by the RNN encoders sharing the same weights θ. The model computes the distance between the latent codes and projects the distance to a probability using the output weight vector W and the sigmoid function σ. A threshold of 0.5 is applied to map the probability to a binary decision. We applied the Gated Recurrent Unit (GRU) [22] as the basic unit for the RNN to achieve better performance.

	NYC		TKY	
	Accuracy	F_1 score	Accuracy	F_1 score
Skip-Gram	0.95	0.76	0.86	0.53
PRME-G	0.91	0.59	0.85	0.48
GE	0.93	0.67	0.84	0.48
POI2Vec	0.95	0.75	0.87	0.49
DXG-full	**0.96**	**0.81**	0.88	0.55
DXG-sem	0.94	0.75	**0.90**	**0.59**

TABLE V: Results from the trajectory verification case study.

VI. Related Work

In recent years, researchers have applied location embedding techniques in the development of various location-based systems [4, 23, 24], because a location embedding is a comprehensive representation capturing location semantics, allowing the models to capture human intentions.

Despite the widespread use of location embedding, only a limited number of works specifically focus on the embedding mechanism itself. Feng et al. [6] introduced metric embedding, which learns embeddings by exploiting transitional patterns and location-user relationships. Liu et al. [2] and Zhao et al. [25] considered the temporal cyclic effects of check-ins and further introduced the latent vectors for time slots to assist the learning of location embeddings. Feng et al. [8] proposed POI2Vec, which adopts the word2vec mechanism by incorporating geographic closeness between POIs into the learning objective. Xie et al. [7] extended the graph embedding introduced by Tang et al. [20] to heterogeneous graph settings, which consists of a POI-POI graph, POI-word graph, etc. Wang and Li [26] advanced the graph embedding method by

[3] See the supplementary materials (see footnote 1) for details.

introducing time-specific location embeddings, which inherently increases the number of parameters to learn.

The main limitation of the related works is that they only consider a part of the spatio-temporal effects, but fail to capture all in a consistent, comprehensive way. In addition, none of them analyzes the resulting embedding space in the unsupervised setting, and they fail to address the problem of entanglement. Our work addresses these weaknesses in a systematic manner by incorporating spatio-temporal attributes and visiting patterns into the model, with improved semantic coherence achieved through disentanglement.

VII. Conclusion

Semantics is the key for learning embeddings. With this in mind, we have presented the Disentangled Skip-Gram (DXG) model, which disentangles the geographic subspace from the semantic subspace and thus is able to improve semantic coherence in the embedding space. Using our query-based metric, we show that our method outperforms four baselines by 39% and 34.25% on average on the NYC and TKY datasets, respectively, at F_1@100. By applying the learned embeddings to two case studies, our model outperforms the baselines by 66.5% (for the next POI recommendation task) and 25% (for the trajectory verification task) on average.

In the future, we plan to apply our technique to more location-based services to further study the capabilities of our method. We also can extend our method to semi-supervised learning settings when a small number of labels are available.

References

[1] H. Wu, Z. Chen, W. Sun, B. Zheng, and W. Wang, “Modeling trajectories with recurrent neural networks,” in *IJCAI*, 2017.

[2] X. Liu, Y. Liu, and X. Li, “Exploring the context of locations for personalized location recommendations.” in *IJCAI*, 2016.

[3] J. He, X. Li, L. Liao, D. Song, and W. K. Cheung, “Inferring a personalized next point-of-interest recommendation model with latent behavior patterns.” in *AAAI*, 2016.

[4] Q. Gao, F. Zhou, K. Zhang, G. Trajcevski, X. Luo, and F. Zhang, “Identifying human mobility via trajectory embeddings,” in *IJCAI*, 2017.

[5] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, “Quantifying location privacy,” in *Security and Privacy (S&P), IEEE Symposium on*, 2011.

[6] S. Feng, X. Li, Y. Zeng, G. Cong, Y. M. Chee, and Q. Yuan, “Personalized ranking metric embedding for next new poi recommendation.” in *IJCAI*, 2015.

[7] M. Xie, H. Yin, H. Wang, F. Xu, W. Chen, and S. Wang, “Learning graph-based poi embedding for location-based recommendation,” in *CIKM*, 2016.

[8] S. Feng, G. Cong, B. An, and Y. M. Chee, “Poi2vec: Geographical latent representation for predicting future visitors.” in *AAAI*, 2017.

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013.

[10] D. Yang, D. Zhang, V. W. Zheng, and Z. Yu, “Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2015.

[11] Y. Liang, Z. Jiang, and Y. Zheng, “Inferring traffic cascading patterns,” in *Proceedings of the 25th ACM SIGSPATIAL*, 2017.

[12] M. Gomez-Rodriguez, D. Balduzzi, and B. Schölkopf, “Uncovering the temporal dynamics of diffusion networks,” in *ICML*, 2011.

[13] L. v. d. Maaten and G. Hinton, “Visualizing data using t-sne,” *Journal of machine learning research*, 2008.

[14] S. Jean, K. Cho, R. Memisevic, and Y. Bengio, “On using very large target vocabulary for neural machine translation,” *arXiv preprint arXiv:1412.2007*, 2014.

[15] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[16] C. Zhang, K. Zhang, Q. Yuan, H. Peng, Y. Zheng, T. Hanratty, S. Wang, and J. Han, “Regions, periods, activities: Uncovering urban dynamics via cross-modal representation learning,” in *WWW*, 2017.

[17] T. Schnabel, I. Labutov, D. Mimno, and T. Joachims, “Evaluation methods for unsupervised word embeddings,” in *EMNLP*, 2015.

[18] C. D. Manning, P. Raghavan, and H. Schütze, *An Introduction to Information Retrieval.* Cambridge University Press, 2008.

[19] B. Chang, Y. Park, D. Park, S. Kim, and J. Kang, “Content-aware hierarchical point-of-interest embedding model for successive poi recommendation.” in *IJCAI*, 2018.

[20] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, “LINE: Large-scale information network embedding,” in *WWW*, 2015.

[21] S. Chopra, R. Hadsell, and Y. LeCun, “Learning a similarity metric discriminatively, with application to face verification,” in *CVPR*, 2005.

[22] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.

[23] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, and D. Jin, “Deepmove: Predicting human mobility with attentional recurrent networks,” in *WWW*, 2018.

[24] C. Cheng, H. Yang, M. R. Lyu, and I. King, “Where you like to go next: Successive point-of-interest recommendation.” in *IJCAI*, 2013.

[25] S. Zhao, T. Zhao, I. King, and M. R. Lyu, “Geo-teaser: Geo-temporal sequential embedding rank for point-of-interest recommendation,” in *WWW*, 2017.

[26] H. Wang and Z. Li, “Region representation learning via mobility flow,” in *WWW*, 2017.