

# HME: A Hyperbolic Metric Embedding Approach for Next-POI Recommendation

Shanshan Feng  
Inception Institute of Artificial  
Intelligence, UAE  
victor\_fengss@foxmail.com

Lucas Vinh Tran  
Nanyang Technological University  
Institute for Infocomm Research,  
A\*STAR  
trandang001@e.ntu.edu.sg

Gao Cong  
Nanyang Technological University,  
Singapore  
gaocong@ntu.edu.sg

Lisi Chen  
Inception Institute of Artificial  
Intelligence, UAE  
lisi.chen@inceptioniai.org

Jing Li  
Inception Institute of Artificial  
Intelligence, UAE  
jingli.phd@hotmail.com

Fan Li  
Fraunhofer Singapore  
lifan3572@gmail.com

## ABSTRACT

With the increasing popularity of location-aware social media services, next-Point-of-Interest (POI) recommendation has gained significant research interest. The key challenge of next-POI recommendation is to precisely learn users' sequential movements from sparse check-in data. To this end, various embedding methods have been proposed to learn the representations of check-in data in the Euclidean space. However, their ability to learn complex patterns, especially hierarchical structures, is limited by the dimensionality of the Euclidean space. To this end, we propose a new research direction that aims to learn the representations of check-in activities in a *hyperbolic space*, which yields two advantages. First, it can effectively capture the underlying hierarchical structures, which are implied by the power-law distributions of user movements. Second, it provides high representative strength and enables the check-in data to be effectively represented in a low-dimensional space. Specifically, to solve the next-POI recommendation task, we propose a novel hyperbolic metric embedding (HME) model, which projects the check-in data into a hyperbolic space. The HME jointly captures sequential transition, user preference, category and region information in a unified approach by learning embeddings in a shared hyperbolic space. To the best of our knowledge, this is the first study to explore a non-Euclidean embedding model for next-POI recommendation. We conduct extensive experiments on three check-in datasets to demonstrate the superiority of our hyperbolic embedding approach over the state-of-the-art next-POI recommendation algorithms. Moreover, we conduct experiments on another four online transaction datasets for next-item recommendation to further demonstrate the generality of our proposed model.

## CCS CONCEPTS

• Information systems → Recommender systems; • Computing methodologies → Learning latent representations.

## KEYWORDS

Next-POI Recommendation; Hyperbolic Space; Metric Embedding

### ACM Reference Format:

Shanshan Feng, Lucas Vinh Tran, Gao Cong, Lisi Chen, Jing Li, and Fan Li. 2020. HME: A Hyperbolic Metric Embedding Approach for Next-POI Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*, July 25–30, 2020, Virtual Event, China. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3397271.3401049>

## 1 INTRODUCTION

Recent years have witnessed the rapid growth of location-based social networks (LBSNs), such as Foursquare. In LBSNs, users share their locations by checking-in at Points-of-Interest (POIs). With the increasing availability of check-in data, POI recommendation (e.g., [19, 21, 46, 48]) has been extensively investigated, helping users to better explore their surroundings and find interesting locations based on their preferences. Among various types of POI recommendation tasks, next-POI recommendation [7, 11] is one of the most popular recommendation problems, which aims to suggest POIs for a user to visit over the next few hours based on the current location of the user.

The key challenge of next-POI recommendation is to effectively model the personalized sequential transitions from the sparse check-in data. To solve this task, various latent representation models have been proposed by exploiting factorized Markov Chain [7, 53], metric embedding [11, 47], or word2vec-based techniques [10, 27, 55, 56]. However, all of them utilize representation models in the Euclidean space to learn the proximity of different items. Although these Euclidean representation models have proved successful for the next-POI recommendation task, they suffer from one inherent limitation: their capability of learning complex patterns is limited by the dimensionality of the Euclidean space [31].

To better understand user sequential movement patterns, we examine two fundamental relations: POI-POI relation (i.e., sequential transition) and POI-user relation (i.e., individual preference). We find that they follow power-law distributions: a majority of nodes

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).  
*SIGIR '20, July 25–30, 2020, Virtual Event, China*

© 2020 Association for Computing Machinery.  
ACM ISBN 978-1-4503-8016-4/20/07...\$15.00  
<https://doi.org/10.1145/3397271.3401049>

have very few connections, and a few nodes have a huge number of connections. It has been studied that power-law distributions often indicate **implicit hierarchical structures** [1, 31, 34], which can explain many topological properties of graphs [8]. In fact, although there is no clearly defined tree structure, many real world information networks exhibit underlying tree-like structures [1]. Moreover, POIs are often associated with region and category tree information, which intuitively indicate **explicit hierarchical structures**, such as category taxonomy [26] and region hierarchy [10]. These observations motivate us to investigate a specific question: how can we effectively capture the underlying hierarchical structures in user check-in activities?

Recently, hyperbolic representation methods [9, 31, 32] have been developed to model latent hierarchical structures. Inspired by these, we propose to learn latent representations of user movements in a hyperbolic space rather than the conventional Euclidean space. Overall, the hyperbolic space is a better choice than the Euclidean space due to its exponential expansion property [18]. Specifically, there are two main advantages of the hyperbolic space. First, it can effectively capture the tree structures, as a tree can be approximately viewed as a discrete version of the hyperbolic space [18]. This is because the number of children in a tree structure expands exponentially with the distance to the root, and the hyperbolic space also expands exponentially with the radius. Second, it is capable of modeling complex data in a lower-dimensional space, because the hyperbolic space expands exponentially with radius, while the Euclidean space only grows polynomially [31].

We develop a novel hyperbolic metric embedding (HME) approach for the next-POI recommendation task. The basic idea is to represent items with the Poincaré ball model, which is commonly used to describe the hyperbolic space and can be easily visualized [31]. The distance in the Poincaré ball model is used to reflect the relation between items. As the next-POI recommendation is an implicit feedback recommendation task [36], we exploit a Bayesian ranking approach, such that related items could be closer together than unrelated items. In addition, since user’s mobility is influenced by multiple factors, we jointly learn four different relationships (POI-POI, POI-User, POI-Category and POI-Region) by projecting them in a shared hyperbolic space.

The learned hyperbolic representations are then exploited for next-POI recommendation. One key challenge is to integrate the effect of user preferences and sequential transitions. However, due to the hyperbolic geometry, we cannot directly apply the linear interpolation as in the Euclidean space. Therefore, we develop an **Einstein midpoint aggregation** method to combine the user preferences and POI sequential transitions in the Poincaré ball model. The geographical distance is also considered since users are inclined to visit POIs that are close to their current positions [11].

The main contributions of this paper are summarized as follows:

- We observe that user movements exhibit power-law distributions, which indicate the implicit hierarchical structures. Motivated by this observation, we propose a novel research problem of representing the users’ check-in activities in a low-dimensional hyperbolic space. To the best of our knowledge, this is the first study to utilize the hyperbolic space for POI recommendation tasks.

- We develop a hyperbolic metric embedding approach to learn latent representations within the Poincaré ball model. We jointly model the POI sequential transitions, user preferences, regional and categorical information in a unified way. When making a recommendation with the learned hyperbolic representations, we design an Einstein midpoint aggregation method to combine the effect of sequential transitions and user preferences.
- We conduct extensive experiments on three real-world check-in datasets, which demonstrate the significant improvement of the hyperbolic metric embedding against the conventional Euclidean-based embedding methods. In particular, HME in 10-dimensionality is able to outperform all the baselines by more than 120%, which shows the strength of hyperbolic representations in lower-dimensional spaces.
- We conduct next-item recommendation experiments on another four online transaction datasets to further demonstrate the generality of our proposed method. The results show that our HME is not specific for next-POI recommendation and can also be applied to other domains.

## 2 RELATED WORK

### 2.1 Next-POI Recommendation

POI recommendation problems have been extensively studied [6, 19, 21, 23, 40, 49, 54], in which the main objective is to model user preferences and geographical influences. Among the various types of problems, next-POI recommendation, which additionally exploits sequential transitions between POIs, has recently attracted significant attention, and various methods have been proposed. Some studies directly utilize the Markov Chain [50] or Hidden Markov Chain model [45] to model sequential movements. However, most proposed methods exploit latent representation models to learn the personalized sequential movements. For instance, the factorized Markov chain [36] has been exploited to model the personalized POI transitions [7, 15, 22, 53], and the metric embedding [11, 47] has been used to model the user preferences and POI transitions. Recently, the word2vec technique [30] has also been explored for next-POI recommendation: some studies [10, 29] first build a tree structure and then exploit the hierarchical softmax [30] to learn the embeddings of different items; other studies [5, 27, 33, 43, 55, 56] utilize the negative sampling strategy [30] to jointly learn the embeddings of POIs as well as other factors. However, all of these methods are designed for Euclidean spaces. Therefore, they cannot effectively model the latent hierarchical structures of user mobility as discussed in Introduction. Different from conventional Euclidean representation models, we propose a hyperbolic representation model to learn the representations of user check-in activities in a non-Euclidean space.

A variant of next-POI recommendation problem is also considered, where a sequence of historical POIs are available for recommending the next POI. For the variant problem, recurrent neural network (RNN) methods have been utilized to model check-in trajectories [2, 20, 25, 52]. Since RNN models capture a sequence of check-ins through hidden states, they are not designed to model first-order relationships between POIs. Additionally, the RNN models require a sequence of denser check-ins to capture higher-order

dependencies. In our problem setting, only the current POI is available, but not the previous POI trajectory. In other words, we only consider first-order sequential transitions, and thus these RNN-based methods cannot be used.

## 2.2 Next-Item Recommendation

The embedding models are not only utilized in next-POI recommendation, but also adopted in various next-item recommendation tasks. Rendle *et al.* [36] propose the factorized personalized Markov Chain for next-basket products recommendation. Wang *et al.* [41] propose a hierarchical representation model for next-product recommendation by applying different aggregation operations. He *et al.* [16] develop a translation based recommendation method for modeling sequential behaviors. Similar to the next-POI recommendation task, most of the existing solutions for next-item recommendation are based on Euclidean spaces. In this work, we also explore the feasibility of using the hyperbolic representations for next-item recommendation and empirically compare with the state-of-the-art methods. We are aware of the recent convolutional neural networks [37] and self-attentive methods [51], which are developed for a variant of next-item recommendation, where there exist a sequence of previous items. Hence, these methods cannot be directly used for the next-item recommendation problem we consider in this work.

## 2.3 Hyperbolic Embedding

An increasing number of studies have shown that many types of data exhibit non-Euclidean structures [3]. Recently, hyperbolic embedding approaches have been proposed to learn the latent representation of complex networks, especially hierarchical networks [9, 31, 32, 42]. In fact, hyperbolic geometry has been integrated into recent advanced deep learning frameworks such as the recurrent neural network [12], attention network [13], and graph neural network [24]. The hyperbolic space has also been investigated in recommender systems [4, 39], demonstrating the viability of hyperbolic geometry in capturing individual preferences. Different from the two studies for the traditional recommendation problem, we jointly learn multiple factors for the next-POI recommendation, such as sequential transitions and user preferences, which is more challenging.

## 3 PRELIMINARY ANALYSIS

In this section, we first describe some important factors in user check-in activities. Then we examine the properties of these factors. Based on real world check-in data, we observe that they all exhibit underlying hierarchical structures, which motivates us to explore hyperbolic spaces to model user movements.

### 3.1 Data Description

In LBSNs, users share their locations by checking-in at POIs. We denote the set of users by  $\mathcal{U}$  and the set of POIs as  $\mathcal{L}$ . Each check-in  $\langle u, l, t \rangle$  denotes that user  $u$  visited POI  $l$  at time  $t$ . Each POI  $l$  is associated with a geographical coordinate: latitude and longitude. Based on the coordinate of each POI, we can obtain the regional information. Each POI also has categorical information,

e.g., restaurant or library. The set of regions is denoted by  $\mathcal{R}$  and set of categories is denoted by  $\mathcal{C}$ .

Notably, human movement patterns are very complex and various factors have been studied in POI recommendation tasks [28]. In this work, we mainly consider the following four factors, which have been shown effective in next-POI recommendation [10, 11, 29].

*POI-POI Relation.* Users' check-in behaviors exhibit strong sequential patterns [7]. For example, if a user is having dinner at a restaurant, he would possibly go to a cinema after that. Following previous studies [10, 11], if the time interval between two consecutive check-ins  $l_i$  and  $l_j$  of a user is smaller than  $\tau = 6$  hours, a POI-POI edge  $\langle l_i, l_j \rangle$  exists. These extracted POI-POI edges reflect the sequential patterns of user movements.

*POI-User Relation.* As different users may have different preferences, personalized interests play an important role in check-in behaviors [46, 48]. If a user  $u$  has visited a POI  $l$ , there is a POI-User  $\langle l, u \rangle$  edge.

*POI-Region Relation.* Each POI is associated with latitude and longitude information, which reflects the geographical information. If a POI  $l$  is located in a region  $r$  (we will discuss how to define regions later), there exists an edge  $\langle l, r \rangle$  between them. POIs in the same region may have strong relations as users are more likely to visit nearby places [7, 10].

*POI-Category Relation.* POIs are commonly labeled with categories that specify their types and functions. If a POI  $l$  is associated with a category  $c$ , there is a POI-Category edge  $\langle l, c \rangle$ . Intuitively, POIs in a same category share similar intrinsic features. Additionally, categories are also beneficial for predicting next POIs [14].

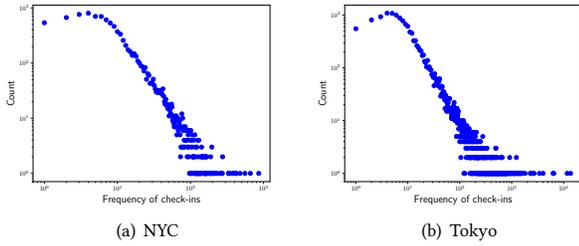
The difficulty of modeling check-in behaviors lies in effectively learning all these factors. To better understand user mobility patterns, we examine characteristics of these factors.

### 3.2 Observation

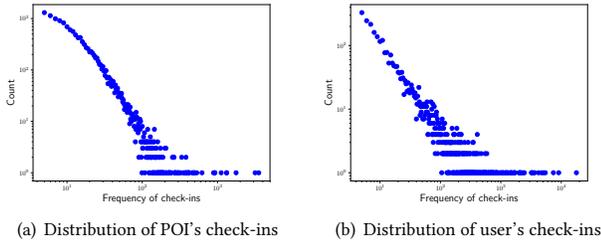
We investigate properties of user movements on real-world datasets: **NYC** and **Tokyo** datasets contain *Foursquare* check-ins within New York City and Tokyo [44], and **Houston** dataset contains *Gowalla* check-ins within Houston [26]. Details of these datasets are presented in Section 6.

To study the sequential transition factor, we show the distributions of POI-POI relations on the NYC and Tokyo datasets in Figure 1. As can be seen, POI-POI transition pairs follow a power-law distribution. Some POIs are more likely to be involved in sequential transitions with other POIs, while most POIs are only connected to a small number of POIs. A similar distribution is also observed on the Houston dataset and not presented here.

To investigate the user preference factor, we present the distribution of frequency of check-ins for users and POIs in Figure 2. As shown in Figure 2(a), some POIs attract many users while the majority of POIs are only visited by a small number of users. Similarly, Figure 2(b) shows that a small portion of users have many check-ins while most of users only have a few check-ins. The check-in activities of POIs and users both follow the power-law distribution, which is consistent with the results reported in [33].



**Figure 1: Distributions of POI-POI relations on NYC and Tokyo. The X-axis presents the POIs’ number of POI-POI transitions and the Y-axis shows the count of such POIs.**



**Figure 2: Distributions of the frequency of check-ins for users and POIs. The X-axis presents the number of check-ins associated with a POI or user, and the Y-axis shows the count of such POIs or users.**

Since power-law distributions often suggest underlying hierarchical structures [31, 34], we claim that the check-in data exhibits *implicit hierarchical structures*, which also widely exist in various real world information networks [1].

In addition to implicit hierarchical structures, the POIs may also exhibit some *explicit hierarchical structures*, such as the category taxonomy [26] and region hierarchy [10, 29]. The three check-in datasets used in this work contain the categories of POIs, which can be intuitively organized in a category tree. Based on the geographical coordinates, POIs can also be hierarchically split into different regions. Following [10], we recursively divide the whole geographical space into sub-regions. In this way, a region tree is constructed. Based on the category tree and region tree, we can obtain POI-Category relations and POI-Region relations that reflect the explicit hierarchical structures.

We investigate four kinds of relations: POI-POI, POI-User, POI-Region, and POI-Category. We find that they exhibit either implicit or explicit hierarchical structures. These common characteristics motivate us to study how to effectively capture the hierarchical structures in user mobility.

### 3.3 Discussion

To learn the representations of user check-in activities, various latent representation models have been proposed. They exploit factorized Markov Chain [7, 53], metric embedding [11, 47], or word2vec-based techniques [10, 27, 55, 56]. All of them utilize representation models in Euclidean spaces to learn the proximity of different items.

Although Euclidean representation models have proved successful for the POI recommendation task, they suffer from one inherent

limitation. Their capability of learning complex patterns, especially latent hierarchical structures, is limited by the dimensionality of the embedding space [31]. To this end, we propose to learn representations in a hyperbolic space. Hyperbolic spaces are spaces of constant negative curvature, while Euclidean spaces have zero curvature. Therefore, hyperbolic spaces exhibit an exponential expansion property [18], which leads to two additional advantages. First, it can effectively model hierarchical structures [31]. Second, it enables complex data to be modeled with small dimensionality, thus reducing the number of parameters in the representation model.

## 4 PROBLEM STATEMENT

Based on the above observations and analysis, we propose a novel research problem. To overcome the limitations of conventional Euclidean representation methods, we attempt to model check-in activities in a hyperbolic space. The problem is formulated below.

**DEFINITION 1. (Hyperbolic Representation of User Mobility)** Given 1) four types of nodes: POI set  $\mathcal{L}$ , user set  $\mathcal{U}$ , region set  $\mathcal{R}$ , and category set  $\mathcal{C}$ ; and 2) four types of relations: POI-POI edge set  $\mathcal{E}_{LL}$ , POI-user edge set  $\mathcal{E}_{LU}$ , POI-region edge set  $\mathcal{E}_{LR}$ , and POI-category edge set  $\mathcal{E}_{LC}$ ; we aim at learning the representation of each node  $v$  in a  $d$ -dimensional hyperbolic space:  $\mathbf{x}(v) \in \mathbb{R}^d$ .

To the best of our knowledge, this is the first study to represent check-in data in the hyperbolic space. With the latent representations of different items, we can solve the next-POI recommendation task. Given a user and his current location, we aim at recommending a set of new POIs to visit in the next few hours. Following previous studies [7, 11], we formally define the research task below.

**DEFINITION 2. (Next-POI Recommendation Task)** Given the current check-in  $\langle u, l^c, t \rangle$  of a user  $u$ , and the set of POIs that user  $u$  has visited  $\mathcal{L}^u$ , the next-POI recommendation task is to recommend a set of unvisited POIs  $\mathcal{L}^{u, l^c} = \{l \in \mathcal{L} \setminus \mathcal{L}^u\}$  for user  $u$  to visit in the time period  $[t, t + \tau]$ .

Here,  $\tau$  is the time interval and commonly set to 6 hours [11]. Note that in this work, we only utilize the first-order sequential transitions by following [11, 47].

## 5 HYPERBOLIC REPRESENTATIONS FOR NEXT-POI RECOMMENDATION

In this section, we first introduce the proposed hyperbolic metric embedding. Then we present a method to exploit the learned representations for making personalized next-POI recommendation.

### 5.1 Hyperbolic Metric Embedding

To describe the hyperbolic space, there are several commonly used models, such as the Poincaré model, Klein model, and hyperboloid model [13]. These models can be converted into each others. We use the Poincaré ball model, which can be better visualized [31].

The basic idea of our hyperbolic metric embedding (HME) is to represent items with the Poincaré ball model, such that the related items are close to each other.

**5.1.1 Optimization Criterion.** The Poincaré ball  $\mathcal{B}^d = \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\| < 1\}$  describes a hyperbolic space, in which the points are in a  $d$ -dimensional unit ball. Here  $\|\cdot\|$  denotes the Euclidean norm. To

learn the latent representations of nodes, we exploit the distances in the Poincaré ball model to measure their proximity. Given an edge  $\langle a, b \rangle$ , the representations  $\mathbf{x}_a$  and  $\mathbf{x}_b$  in the Poincaré ball should be close to each other. Different from the intuitive Euclidean distance, the distance in the Poincaré ball is stated as follows:

$$\mathcal{D}_{ab} = \operatorname{arcosh} \left( 1 + 2 \frac{\|\mathbf{x}_a - \mathbf{x}_b\|^2}{(1 - \|\mathbf{x}_a\|^2)(1 - \|\mathbf{x}_b\|^2)} \right), \quad (1)$$

where  $\operatorname{arcosh}(x) = \ln(x + \sqrt{x^2 - 1})$  is an inverse hyperbolic cosine function. One interesting feature is that the distance varies with the location of  $\mathbf{x}_a$  and  $\mathbf{x}_b$ . For example, when  $\mathbf{x}_a$  and  $\mathbf{x}_b$  are near to the boundary of the ball, i.e.  $(1 - \|\mathbf{x}_a\|^2) \rightarrow 0$  and  $(1 - \|\mathbf{x}_b\|^2) \rightarrow 0$ , their distance is much larger than when they are near to the center of the ball. As the next-POI recommendation is an implicit feedback recommendation task [36], we utilize the Bayesian personalized ranking (BPR) method [35] to learn the likelihood of training pairs. For each given positive pair  $\langle a, b \rangle$ , we randomly sample a small number  $k$  of negative nodes, each denoted by  $n$ , following the negative sampling strategy in [30]. The node  $b$  is supposed to be ranked higher than the negative node  $n$ , which can be reflected by a ranking probability  $P(b > n|a)$ . By utilizing the distance in the hyperbolic space, the ranking probability can be written as:

$$P(b > n|a) = \sigma(\mathcal{D}_{an} - \mathcal{D}_{ab}), \quad (2)$$

where  $\sigma(z) = \frac{1}{1+e^{-z}}$  is a logistic function, and  $\mathcal{D}_{an}$  is the distance between  $\mathbf{x}_a$  and  $\mathbf{x}_n$  in the Poincaré ball model. Equation (2) reflects the intuition that the distance between a negative pair should be larger than the distance between a positive pair  $\mathcal{D}_{an} > \mathcal{D}_{ab}$ .

By maximizing the log-likelihood, the optimization criterion of the HME model can be derived as:

$$\begin{aligned} \Theta &= \operatorname{argmax}_{\Theta} \sum_{(a,b) \in \mathcal{E}} \sum_{n \in \mathcal{N}_{ab}} \log P(b > n|a) \\ &= \operatorname{argmax}_{\Theta} \sum_{(a,b) \in \mathcal{E}} \sum_{n \in \mathcal{N}_{ab}} \log \sigma(\mathcal{D}_{an} - \mathcal{D}_{ab}). \end{aligned} \quad (3)$$

Here  $\mathcal{N}_{ab}$  is the negative nodes sampled for each pair  $\langle a, b \rangle$  in the training dataset; and  $\Theta$  is the hyperbolic representations of node set  $\mathcal{V}$ . In this paper, we set  $k = |\mathcal{N}_{ab}| = 5$ .

**5.1.2 Learning Procedure.** Different from Euclidean embedding methods, we cannot directly use the Stochastic Gradient Descent (SGD) due to the Riemannian manifold structure of the Poincaré ball. Following [31], we first calculate Euclidean gradients and then combine them with the Riemannian gradient to update parameters.

For each ranking pair  $(b > n|a)$ , we use  $E$  to denote the log-likelihood  $E = \log(\sigma(\mathcal{D}_{an} - \mathcal{D}_{ab}))$ . We calculate the Euclidean derivatives for variables as follows:

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{x}_a} &= (1 - \sigma(\mathcal{D}_{an} - \mathcal{D}_{ab})) \left( \frac{\partial \mathcal{D}_{an}}{\partial \mathbf{x}_a} - \frac{\partial \mathcal{D}_{ab}}{\partial \mathbf{x}_a} \right) \\ \frac{\partial E}{\partial \mathbf{x}_n} &= (1 - \sigma(\mathcal{D}_{an} - \mathcal{D}_{ab})) \left( \frac{\partial \mathcal{D}_{an}}{\partial \mathbf{x}_n} \right) \\ \frac{\partial E}{\partial \mathbf{x}_b} &= (1 - \sigma(\mathcal{D}_{an} - \mathcal{D}_{ab})) \left( -\frac{\partial \mathcal{D}_{ab}}{\partial \mathbf{x}_b} \right). \end{aligned} \quad (4)$$

The corresponding gradients can be further derived as follows:

$$\begin{aligned} \frac{\partial \mathcal{D}_{ab}}{\partial \mathbf{x}_a} &= \frac{4}{\beta_a \sqrt{\gamma_{ab}^2 - 1}} \left( \frac{\|\mathbf{x}_b\|^2 - 2 \langle \mathbf{x}_a, \mathbf{x}_b \rangle + 1}{\alpha^2} \mathbf{x}_a - \frac{\mathbf{x}_b}{\alpha} \right) \\ \frac{\partial \mathcal{D}_{ab}}{\partial \mathbf{x}_b} &= \frac{4}{\alpha \sqrt{\gamma_{ab}^2 - 1}} \left( \frac{\|\mathbf{x}_a\|^2 - 2 \langle \mathbf{x}_a, \mathbf{x}_b \rangle + 1}{\beta_b^2} \mathbf{x}_b - \frac{\mathbf{x}_a}{\beta_b} \right) \\ \frac{\partial \mathcal{D}_{an}}{\partial \mathbf{x}_a} &= \frac{4}{\beta_n \sqrt{\gamma_{an}^2 - 1}} \left( \frac{\|\mathbf{x}_n\|^2 - 2 \langle \mathbf{x}_a, \mathbf{x}_n \rangle + 1}{\alpha^2} \mathbf{x}_a - \frac{\mathbf{x}_n}{\alpha} \right) \\ \frac{\partial \mathcal{D}_{an}}{\partial \mathbf{x}_n} &= \frac{4}{\alpha \sqrt{\gamma_{an}^2 - 1}} \left( \frac{\|\mathbf{x}_a\|^2 - 2 \langle \mathbf{x}_a, \mathbf{x}_n \rangle + 1}{\beta_n^2} \mathbf{x}_n - \frac{\mathbf{x}_a}{\beta_n} \right), \end{aligned} \quad (5)$$

where  $\alpha = 1 - \|\mathbf{x}_a\|^2$ ,  $\beta_b = 1 - \|\mathbf{x}_b\|^2$ ,  $\beta_n = 1 - \|\mathbf{x}_n\|^2$ ,  $\gamma_{ab} = 1 + \frac{2}{\alpha\beta_b} \|\mathbf{x}_a - \mathbf{x}_b\|^2$  and  $\gamma_{an} = 1 + \frac{2}{\alpha\beta_n} \|\mathbf{x}_a - \mathbf{x}_n\|^2$ .

Combining with the Riemannian gradient [31], we obtain the update procedure for the variables as:

$$\begin{aligned} \mathbf{x}_a^{t+1} &\leftarrow \operatorname{norm} \left( \mathbf{x}_a^t + lr \frac{(1 - \|\mathbf{x}_a^t\|^2)^2}{4} \frac{\partial E}{\partial \mathbf{x}_a} \right) \\ \mathbf{x}_b^{t+1} &\leftarrow \operatorname{norm} \left( \mathbf{x}_b^t + lr \frac{(1 - \|\mathbf{x}_b^t\|^2)^2}{4} \frac{\partial E}{\partial \mathbf{x}_b} \right) \\ \mathbf{x}_n^{t+1} &\leftarrow \operatorname{norm} \left( \mathbf{x}_n^t + lr \frac{(1 - \|\mathbf{x}_n^t\|^2)^2}{4} \frac{\partial E}{\partial \mathbf{x}_n} \right), \end{aligned} \quad (6)$$

where  $lr$  is the learning rate,  $\mathbf{x}^t$  is the embedding  $\mathbf{x}$  at iteration  $t$ ; and  $\operatorname{norm}(\cdot)$  is a normalization function to constrain representations in the Poincaré unit ball:

$$\operatorname{norm}(\mathbf{x}) = \begin{cases} \mathbf{x}/\|\mathbf{x}\| - \epsilon, & \text{if } \|\mathbf{x}\| \geq 1 \\ \mathbf{x}, & \text{otherwise.} \end{cases} \quad (7)$$

Here  $\epsilon = 10^{-5}$  is a small value to make sure that embeddings are always within the Poincaré ball.

In this paper, the four bipartite relations, i.e., POI-POI, POI-User, POI-Region, and POI-Category, can be jointly learned with the proposed hyperbolic metric embedding method. We learn the representations of all nodes in the same Poincaré ball.

The time complexity of the HME algorithm is  $O(I \cdot k \cdot d \cdot |\mathcal{E}|)$ , where  $I$  is the number of iterations,  $k$  is the number of negative samples for each edge,  $d$  is the number of dimensions, and  $|\mathcal{E}|$  is the number of edges extracted from a given check-in dataset.

## 5.2 Recommending with Hyperbolic Embeddings

With the learned hyperbolic embeddings of users and POIs, we can provide personalized POI recommendations for a user to visit in the next few hours. Given a user  $u$  and current location  $l_c$ , we need to calculate a list of POIs that are closely related to the query  $(u, l_c)$ , i.e., those POIs with the shortest distances to the query.

Since both user preferences and sequential transitions are important for next-POI recommendation [7, 11], we need to combine these two factors by computing the aggregation of  $u$  and  $l_c$ . In conventional Euclidean methods, we can easily get the linear interpolation of these points as:  $\mathbf{x} = w \cdot \mathbf{x}_u + (1 - w) \cdot \mathbf{x}_{l_c}$ , where  $w \in [0, 1]$  controls the weight of different embeddings. However, due to the hyperbolic geometry, we cannot simply apply the linear interpolation of two points in the Poincaré ball model.

Therefore, it is necessary to design a new way to calculate the aggregation. We resort to the **Einstein midpoint aggregation** [13, 38] in the Klein model, since it exhibits a weighted average manner and thus provides an efficient aggregation operation. Note

that the Poincaré model and the Klein model describe the same hyperbolic space using different coordinates. Thus, we first convert Poincaré ball coordinates to Klein coordinates, and then calculate the aggregated point. After that, we transfer the Klein coordinates back to the Poincaré ball model.

Given two  $d$ -dimensional embeddings  $\mathbf{x}_u \in \mathcal{B}^d$  and  $\mathbf{x}_{l_c} \in \mathcal{B}^d$  in the Poincaré ball model, we convert them to the Klein model by conducting the following transformation:

$$\mathbf{x}_u^{\mathcal{K}} = \frac{2 \cdot \mathbf{x}_u}{1 + \|\mathbf{x}_u\|^2}, \quad \mathbf{x}_{l_c}^{\mathcal{K}} = \frac{2 \cdot \mathbf{x}_{l_c}}{1 + \|\mathbf{x}_{l_c}\|^2}. \quad (8)$$

Here,  $\mathbf{x}_u^{\mathcal{K}} \in R^d$  and  $\mathbf{x}_{l_c}^{\mathcal{K}} \in R^d$  are the corresponding points in the  $d$ -dimensional Klein model. The Einstein midpoint aggregation  $\mathbf{x}_{ag}^{\mathcal{K}}$  is then calculated as:

$$\mathbf{x}_{ag}^{\mathcal{K}} = \frac{w \cdot \psi_u}{w \cdot \psi_u + (1-w) \cdot \psi_{l_c}} \cdot \mathbf{x}_u^{\mathcal{K}} + \frac{(1-w) \cdot \psi_{l_c}}{w \cdot \psi_u + (1-w) \cdot \psi_{l_c}} \cdot \mathbf{x}_{l_c}^{\mathcal{K}}, \quad (9)$$

where  $\psi_u = \frac{1}{\sqrt{1 - \|\mathbf{x}_u^{\mathcal{K}}\|^2}}$  and  $\psi_{l_c} = \frac{1}{\sqrt{1 - \|\mathbf{x}_{l_c}^{\mathcal{K}}\|^2}}$  are Lorentz factors of  $\mathbf{x}_u^{\mathcal{K}}$  and  $\mathbf{x}_{l_c}^{\mathcal{K}}$ ; and  $w \in [0, 1]$  denotes the component weight.

Finally, the  $d$ -dimensional aggregated point  $\mathbf{x}_{ag}^{\mathcal{K}}$  in the Klein model can be converted into the Poincaré ball model as:

$$\mathbf{x}_{ag} = \frac{\mathbf{x}_{ag}^{\mathcal{K}}}{1 + \sqrt{1 - \|\mathbf{x}_{ag}^{\mathcal{K}}\|^2}}. \quad (10)$$

With this Einstein midpoint aggregation method, we obtain the aggregated point  $\mathbf{x}_{ag}$  of the given query  $(u, l_c)$ . For each POI candidate  $l$ , we calculate the distance  $\mathcal{D}_{l,ag}$  between  $\mathbf{x}_l$  and  $\mathbf{x}_{ag}$  using Equation (1). In addition, the geographical distance can be accommodated, which has been demonstrated beneficial for next-POI recommendation. For a fair comparison, we exploit the same strategy as [11]. Specifically, we compute the fused distance score

$$\mathcal{D}_{l,ag}^{Geo} = (1 + d_{l,l_c})^{0.25} * \mathcal{D}_{l,ag}, \quad (11)$$

where  $d_{l,l_c}$  is the distance calculated by the geographical coordinates of POI  $l$  and current POI  $l_c$ . By sorting the fused distance scores  $\mathcal{D}_{l,ag}^{Geo}$  of POI candidates, a list of POIs with the smallest fused distance scores are returned as the recommendation result.

## 6 EXPERIMENTS

### 6.1 Experimental Settings

We evaluate the performance of our hyperbolic metric embedding model on three research tasks:

- **RT1: Sequential Transition.** Given a location, we predict the most likely successive locations. This task is to evaluate the quality of capturing POI sequential transitions. Here, we only consider POI-POI relations.
- **RT2: Next-POI Recommendation.** Given a user and his/her current location, we suggest a list of POIs for the user to check-in next.
- **RT3: Next-Item Recommendation.** Although HME is proposed for next-POI recommendation, it can be applied to other domains. To demonstrate the generality of the proposed HME model, we evaluate the empirical results on the next-item recommendation task, where we recommend products for a user to purchase next based on the last item that the user purchased.

For the first two research tasks, i.e., **RT1** and **RT2**, we use three publicly available real-world check-in datasets: **NYC** and **Tokyo** from the *Foursquare* check-in dataset [44], and **Houston** from the *Gowalla* check-in dataset [26]. The basic statistics of these datasets are summarized in Table 1.

Dataset	Platform	#User	#POI	#Check-ins
NYC	Foursquare	1,083	9,989	227,428
Tokyo	Foursquare	2,293	15,177	573,703
Houston	Gowalla	4,627	15,135	362,783

**Table 1: Statistics of the three check-in datasets**

Our model is able to incorporate POI categories and regions. All the check-in datasets contain category taxonomy information, and the categories in the top-two levels of the category taxonomy are considered. In total, there are 209 categories in the NYC dataset, 206 categories in the Tokyo dataset, and 140 categories in the Houston dataset. We recursively divide the whole region into four equal size sub-regions until the depth reaches 4. Each POI is associated with all the sub-regions (including the leaf sub-region and its antecedents) that cover the POI.

For the last research task, i.e., **RT3**, we use the Amazon datasets [17]. Four sub-categories are chosen with diverse domains and densities: **APP** (Apps for Android), **Music** (Digital Music), **Game** (Video Games), and **Health** (Health and Personal Care). Each transaction is a tuple of  $\langle \text{user}, \text{item}, \text{time} \rangle$ . Following [51], for each dataset, we remove the users and items with less than 10 transactions. The statistics of the datasets are reported in Table 2.

Dataset	#User	#Item	#Transactions
APP	22,693	14,542	357,150
Health	16,181	36,194	242,776
Game	8,055	15,718	141,608
Music	5,729	9,267	65,344

**Table 2: Statistics of the four online transaction datasets**

For each dataset, we split the first 80% chronological records as the training set, the 80-90% records as the tuning set, and the last 10% as the test set.

### 6.2 Experimental Results

**6.2.1 RT1: Sequential Transition.** This task aims to evaluate the results of capturing sequential transitions. The HME model is compared with several widely used embedding models:

- **FMC** [7, 36]: The factorized Markov chain model, which is commonly used in modeling POI transitions [7, 15, 22, 53].
- **ME**: The Metric Embedding model [11], which utilizes Euclidean distance to learn sequential transitions.
- **Skip-gram**: By treating a user’s check-in sequence as a ‘sentence’ and each POI in a sequence as a ‘word’, the skip-gram technique [30] is widely used to learn POI representations [5, 27, 55, 56].

FMC, ME and Skip-gram are all built upon the Euclidean space: FMC and Skip-gram utilize the inner product of two vectors, and ME exploits the Euclidean distance to learn the relations. To evaluate the quality of capturing sequential transitions, we try to infer new transition pairs in the test dataset. We use mean average precision

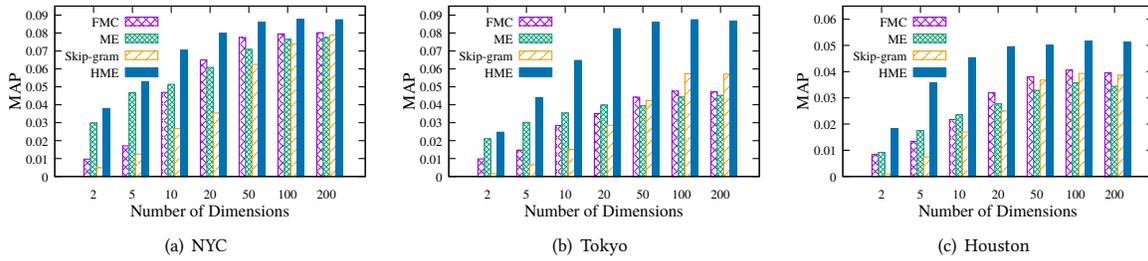


Figure 3: The performance of predicting new sequential transitions with different numbers of dimensions.

(MAP) to measure the capability of predicting new transition pairs.

The MAP results of different embedding methods with different numbers of dimensions are depicted in Figure 3. We can see that HME significantly outperforms the baselines, especially on the sparse Tokyo and Houston datasets, achieving improvements of 50% and 30%, respectively. This result indicates that the hyperbolic space has a stronger capability for capturing the transition relations. For all the methods, the performance first increases with the increasing of number of dimensions  $d$ , and then stops increasing. HME saturates at  $d = 20$ , while the other methods saturate at  $d = 100$ , which shows that HME can effectively learn sequential transitions with smaller  $d$ . All reported improvements over baseline methods are statistically significant with  $p$ -value  $< 0.01$ .

In addition, the embedding model is expected to reflect the hierarchical structure as implied by the power-law distribution of POI-POI relations. In Figure 4, we plot each POI’s degree (i.e., number of POI-POI relations) against the L2-norm of its embedding on the NYC dataset. Similar results are also observed on the other two check-in datasets and not reported in this paper. In these figures, each point represents a POI. For better visualization, the ten most frequent POIs are highlighted by red squares. As indicated in Figure 4(d), the high-degree POIs have a smaller L2-norm. This is because the frequent POIs have more connections with other POIs and are more likely to be driven close to the origin. This observation is reasonable since nodes close to the origin have smaller distances to other nodes due to the hyperbolic geometry. Overall, with the HME, the high-degree POIs are located near the center, while low-degree POIs are far away from the center. We do not observe similar results with the Euclidean embedding models (FMC, ME, and Skip-gram), as shown by Figure 4.

**6.2.2 RT2: Next-POI Recommendation.** We have shown that HME can model the sequential transitions well. In this task, we evaluate its ability to incorporate multiple factors. For next-POI recommendation, sequential transitions and user preferences are the essential factors, and most of the existing algorithms focus on them. For this setting, we evaluate the following methods:

- **FPMC** [7]: The personalized factorized Markov chain [36] with localized region constraints for solving successive POI recommendation.
- **PRME** [11]: The personalized ranking-based metric embedding approach, which models the user preferences and sequential transitions in two separate latent Euclidean spaces.

- **MEAP** [47]: A metric embedding method with asymmetric projections. It extends PRME by utilizing two projection matrices to model asymmetric sequential transitions.
- **JRLM** [55]: The joint representation learning model, which exploits the word2vec framework [30] to jointly learn embeddings of users and POIs.
- **Transrec** [16]: The state-of-the-art embedding model for next-item recommendation. It utilizes a translation-based approach to model sequential behaviors. It adopts the triangle inequality to make recommendation  $\mathbf{u} + \mathbf{l}_c \approx \mathbf{l}$ , where  $\mathbf{u}$  and  $\mathbf{l}_c$  are the embedding of a user and current location  $l_c$ .
- **HME**: Our proposed hyperbolic metric embedding method, which only exploits two bipartite graphs (POI-POI and POI-User) to learn the representations of users and POIs.

In addition, some algorithms are able to incorporate the POI’s regional and categorical information to learn embeddings of items, i.e., utilizing four kinds of relations (POI-POI, POI-User, POI-Region, and POI-Category). We evaluate the following methods:

- **GraphEmb** [43]: A graph-based POI embedding method, which models the first-order proximity between nodes. We adapt the method to learn the four bipartite relations
- **GLR** [29]: The state-of-the-art graph-based latent representation method for next-POI recommendation. It utilizes the word2vec framework [30] to learn representations. We modify it to consider the four bipartite relations.
- **HHNE** [42]: The state-of-the-art hyperbolic embedding for heterogeneous networks. To use the same settings of other baselines, we directly employ this algorithm to learn the four types of relations.
- **HME+**: Our proposed model that jointly models all four types of relations.

For fair comparisons, the geographical distance influence is incorporated accordingly for all the methods. Note that all the Euclidean embedding models achieve their best performance when the number of dimensions is around 100, as reported in these studies. Hence, we set the default number of dimensions as 100 for all the evaluated methods. In addition to MAP, we employ two widely used metrics, namely Precision@K and Recall@K (denoted by Pre@K and Rec@K, respectively). K is the number of POIs to be recommended, and set to 5 and 10. Given the current location of a user, we calculate the scores of POIs that this user has not visited, and return the top-K POI candidates according to their scores.

The experimental results are reported in Table 3. Based on the factors used, the results are divided into two groups. The first group uses POI-POI relations and POI-User relations. When comparing

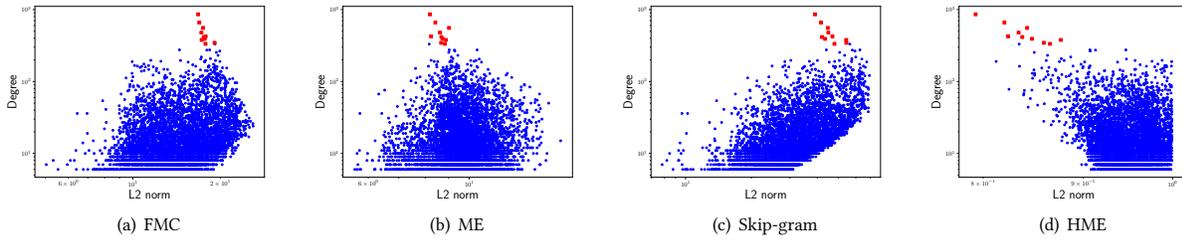


Figure 4: The relationships between a POI’s degree and L2-norm of embedding on the NYC dataset (Best viewed in color).

Dataset	Metric	FPMC	PRME	MEAP	JRLM	Transrec	HME	GraphEmb	GLR	HHNE	HME+	
NYC	MAP	0.0250	0.0483	0.0457	0.0482	0.0460	<b>0.0615</b>	0.0537	0.0546	0.0553	<b>0.0642</b>	
	Pre@5	0.0067	0.0142	0.0137	0.0166	0.0143	<b>0.0193</b>	0.0159	0.0167	0.0166	<b>0.0210</b>	
	Rec@5	0.0274	0.0654	0.0638	0.0764	0.0667	<b>0.0863</b>	0.0727	0.0776	0.0750	<b>0.0962</b>	
	Pre@10	0.0058	0.0103	0.0112	0.0115	0.0102	<b>0.0134</b>	0.0105	0.0121	0.0115	<b>0.0148</b>	
	Rec@10	0.0492	0.0934	0.1033	0.1064	0.0929	<b>0.1223</b>	0.0966	0.1109	0.1035	<b>0.1371</b>	
Tokyo	MAP	0.0855	0.0852	0.0902	0.0898	0.0817	<b>0.1063</b>	0.0729	0.0914	0.0686	<b>0.1115</b>	
	Pre@5	0.0264	0.0297	0.0274	0.0302	0.0271	<b>0.0329</b>	0.0241	0.0313	0.0209	<b>0.0343</b>	
	Rec@5	0.1192	0.1329	0.1223	0.1364	0.1211	<b>0.1469</b>	0.1087	0.1418	0.0923	<b>0.1527</b>	
	Pre@10	0.0170	0.0219	0.0191	0.0218	0.0205	<b>0.0238</b>	0.0173	0.0223	0.0156	<b>0.0242</b>	
	Rec@10	0.1543	0.1950	0.1721	0.1979	0.1833	<b>0.2133</b>	0.1572	0.2033	0.1379	<b>0.2172</b>	
Houston	MAP	0.0629	0.0779	0.0802	0.0729	0.0699	<b>0.0964</b>	0.0780	0.0783	0.0626	<b>0.1006</b>	
	Pre@5	0.0205	0.0288	0.0239	0.0249	0.0251	<b>0.0305</b>	0.0290	0.0272	0.0213	<b>0.0325</b>	
	Rec@5	0.0942	0.1343	0.1128	0.1137	0.1180	<b>0.1441</b>	0.1355	0.1272	0.0994	<b>0.1533</b>	
	Pre@10	0.0142	0.0210	0.0192	0.0196	0.0199	<b>0.0236</b>	0.0226	0.0212	0.0178	<b>0.0246</b>	
	Rec@10	0.1315	0.1954	0.1815	0.1815	0.1853	<b>0.2219</b>	0.2126	0.1980	0.1659	<b>0.2318</b>	
Used factors		POI-POI, POI-User						POI-POI, POI-User, POI-Region, POI-Category				

Table 3: The results of next-POI recommendation on three check-in datasets. All the methods use 100-dimensional representations. Best results are in boldface.

the results of the Euclidean embedding methods (FPMC, PRME, MEAP, JRLM, and Transrec), we find that HME significantly outperforms these Euclidean embedding models on all three check-in datasets. For the second group, which uses all the four bipartite graphs, HME+ can get better results than GraphEmb and GLR. For instance, the MAP of HME+ is 18-28% higher than GraphEmb and GLR on all three datasets. This is because GraphEmb and GLR are based on the Euclidean space, which limits their representative power. In addition, we observe that the performance of HHNE is not satisfying. The reason is that HHNE needs an extensive amount of training samples generated by numerous random walks, and thus is not suitable for directly solving this problem. All reported improvements over baseline methods in this task are also statistically significant with  $p$ -value  $< 0.01$ .

Dataset	GraphEmb	GLR	HHNE	HME+
NYC	0.0136	0.0248	0.0202	<b>0.0599</b>
Tokyo	0.0090	0.0413	0.0419	<b>0.1062</b>
Houston	0.0268	0.0432	0.0467	<b>0.0981</b>

Table 4: The MAP results of next-POI recommendation with 10-dimensional representations. Best results are in boldface.

To further demonstrate the strength of modeling complex data within a low-dimensional space, we present the MAP results with number of dimensions  $d = 10$  in Table 4. We find that HME outperforms the baselines by more than 120% on all three datasets.

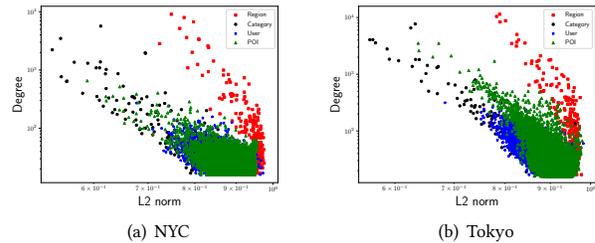


Figure 5: The relationships between degree and L2-norm of embedding for different nodes (Best viewed in color).

Impressively, the HME with 10 dimensionality even outperforms all the baselines with  $d = 100$ , as can be seen by comparing the results in Table 3 and Table 4. This finding demonstrates that our proposed HME can effectively learn good representations even with a small  $d$ . Consequently, the number of parameters can be greatly reduced compared to the conventional Euclidean embedding models.

To examine the embeddings of HME+, we show the distribution of L2-norm with degree in Figure 5. Note that the embeddings of four different types of nodes are jointly learned in the HME+ model. We observe that the higher the degree of a node, the smaller the L2-norm. This observation indicates that multiple relationships can be simultaneously captured.

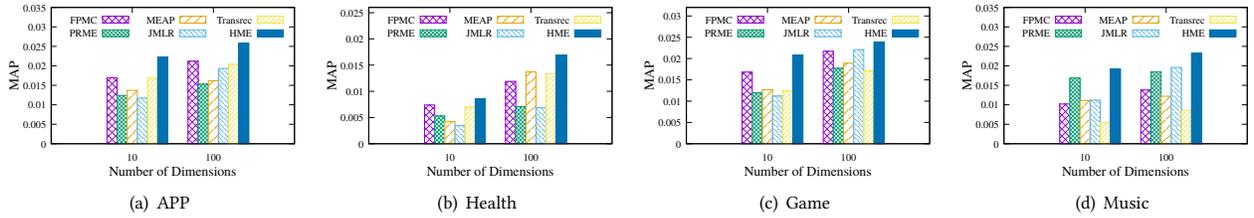


Figure 6: The performance of next item recommendation task on four transaction datasets.

6.2.3 RT3: Next-Item Recommendation. We apply HME to the next-item recommendation problem, and compare with the state-of-the-art solutions to the problem using the same setting discussed earlier. FPMC and Transrec are originally developed for next-item recommendation. PRME, MEAP, JRLM, and HME can be easily adapted for next-item recommendation as follows. Based on the online transaction datasets, we can extract item-item and item-user relations. For each user, we sort his/her transactions by timestamps. If a user purchases product  $p_i$  and then purchases product  $p_j$ , an item-item sequential transition  $\langle p_i, p_j \rangle$  exists. For this task, we use the MAP score as the evaluation metric.

The results of next-item recommendation are presented in Figure 6. We observe that the proposed HME again consistently outperforms the five Euclidean embedding baselines (improvements over baseline methods are statistically significant with p-value  $< 0.01$ ). Particularly, when  $d = 10$ , HME still achieves competitive performance compared to the baselines. The results on the four datasets demonstrate the superiority and generality of our HME model.

### 6.3 Effects of Parameters

We conduct experiments to study the effects of parameters on the performance of HME. To investigate the effect of the weight of different components  $w$  in the next-POI recommendation task, we report the experimental results on the tuning set. As shown in Figure 7, HME is able to achieve remarkable results when  $w \in [0.2, 0.6]$ . The default weight  $w$  is set to 0.5 in the experiments.

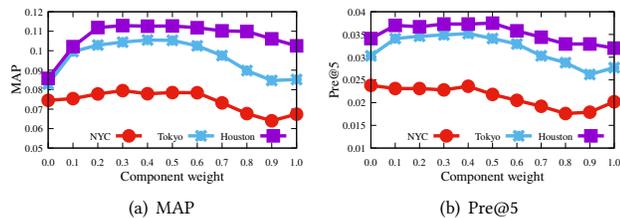


Figure 7: The effect of component weight

Moreover, we also investigate the effect of the number of dimensions  $d$  and report the results in Figure 8. Overall, we observe an increase in performance when  $d$  increases, but the improvement becomes marginal when  $d > 20$ . As shown in Figure 8, the hyperbolic metric embedding can achieve good results even when  $d$  is small. Empirically, the number of dimensions  $d$  can be set to 10-20, which achieves a satisfying trade-off between recommendation quality and running time.

### 6.4 Visualization

To better understand the hyperbolic metric embedding, we show 2-dimensional hyperbolic embeddings on the NYC dataset. We learn

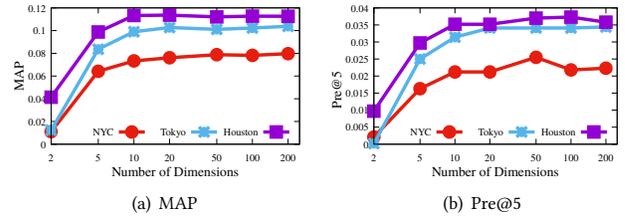
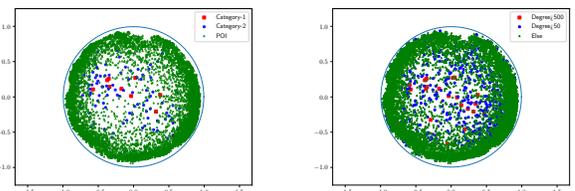


Figure 8: The effect of number of dimensions

the embeddings of different nodes with the HME+ algorithm and then present them in a unit disk. We first visualize the explicit hierarchical structure. We show the embeddings of POIs and categories in Figure 9(a). We observe that the high-level categories (marked by red squares) are closer to the origin, whereas the POIs are located near the boundary. This result indicates that the POI-category hierarchy can be well modeled by the hyperbolic metric embedding. This result is consistent with previous studies [31, 42].

We further study the visualizations of all nodes. Figure 9(a) shows 2-dimensional embeddings of the nodes grouped by their degrees. Overall, the high-degree nodes are located near the center while low-degree nodes are placed around the boundary. This visualization demonstrates that the hyperbolic metric embedding can also effectively reflect the implicit hierarchical structure.



(a) The embeddings of POIs and categories (b) The embeddings of all nodes

Figure 9: Visualization of 2-dimensional embeddings on the NYC dataset (Best viewed in color).

## 7 CONCLUSION

In this paper, we investigate a novel non-Euclidean representation problem for the next-POI recommendation task. We develop a hyperbolic metric embedding approach that projects the items into a Poincaré unit ball model, and hence is capable of capturing the underlying hierarchical structures in the check-in data. We jointly consider multiple factors in the same hyperbolic metric embedding model: POI-POI, POI-User, POI-Region, and POI-Category relations. With the learned hyperbolic embeddings, we can provide personalized next-POI recommendations given a user and his/her current location. To combine user preferences and sequential transitions, we further derive an Einstein midpoint aggregation method for

hyperbolic representations. Based on three real-world check-in datasets, extensive experiments are conducted to demonstrate the significant superiority of our hyperbolic metric embedding against various conventional Euclidean competitors. Empirical results also demonstrate the generality of HME for next-item recommendation.

## ACKNOWLEDGMENT

Gao Cong is supported in part by a MOE Tier-1 grant RG114/19.

## REFERENCES

- [1] Aaron B. Adcock, Blair D. Sullivan, and Michael W. Mahoney. 2013. Tree-Like Structure in Large Social and Information Networks. In *ICDM*. 1–10.
- [2] Basmah Altaf, Lu Yu, and Xiangliang Zhang. 2018. Spatio-Temporal Attention based Recurrent Neural Network for Next Location Prediction. In *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, 937–942.
- [3] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. 2017. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Process. Mag.* 34, 4 (2017), 18–42.
- [4] Benjamin Paul Chamberlain, Stephen R Hardwick, David R Wardrop, Fabon Dzugang, Fabio Daolio, and Saúl Vargas. 2019. Scalable Hyperbolic Recommender Systems. *arXiv preprint arXiv:1902.08648* (2019).
- [5] Buru Chang, Yonggyu Park, Donghyeon Park, Seongsoo Kim, and Jaewoo Kang. 2018. Content-Aware Hierarchical Point-of-Interest Embedding Model for Successive POI Recommendation. In *IJCAI*. 3301–3307.
- [6] Chen Cheng, Haiqin Yang, Irwin King, and Michael R Lyu. 2012. Fused Matrix Factorization with Geographical and Social Influence in Location-Based Social Networks. In *AAAI*. 17–23.
- [7] Chen Cheng, Haiqin Yang, Michael R Lyu, and Irwin King. 2013. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*. 2605–2611.
- [8] Aaron Clauset, Christopher Moore, and Mark EJ Newman. 2008. Hierarchical structure and the prediction of missing links in networks. *Nature* (2008).
- [9] Christopher De Sa, Albert Gu, Christopher Ré, and Frederic Sala. 2018. Representation tradeoffs for hyperbolic embeddings. *ICML* (2018).
- [10] Shanshan Feng, Gao Cong, Bo An, and Yeow Meng Chee. 2017. Poi2vec: Geographical latent representation for predicting future visitors. In *AAAI*.
- [11] Shanshan Feng, Xutao Li, Yifeng Zeng, Gao Cong, Yeow Meng Chee, and Quan Yuan. 2015. Personalized ranking metric embedding for next new POI recommendation. In *IJCAI*. 2069–2075.
- [12] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic neural networks. In *NeurIPS*. 5345–5355.
- [13] Caglar Gulcehre, Misha Denil, Mateusz Malinowski, Ali Razavi, Razvan Pascanu, Karl Moritz Hermann, Peter Battaglia, Victor Bapst, David Raposo, Adam Santoro, et al. 2019. Hyperbolic attention networks. *ICLR* (2019).
- [14] Jing He, Xin Li, and Lejian Liao. 2017. Category-aware Next Point-of-Interest Recommendation via Listwise Bayesian Personalized Ranking. In *IJCAI*.
- [15] Jing He, Xin Li, Lejian Liao, Dandan Song, and William K Cheung. 2016. Inferring a Personalized Next Point-of-Interest Recommendation Model with Latent Behavior Patterns. In *AAAI*. 137–143.
- [16] Ruining He, Wang-Cheng Kang, and Julian McAuley. 2018. Translation-based Recommendation: A Scalable Method for Modeling Sequential Behavior. In *IJCAI*.
- [17] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *WWW*.
- [18] Dmitri V. Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguñá. 2010. Hyperbolic Geometry of Complex Networks. *Physical Review E* (2010).
- [19] Huayu Li, Yong Ge, Richang Hong, and Hengshu Zhu. 2016. Point-of-interest recommendations: Learning potential check-ins from friends. In *SIGKDD*.
- [20] Ranzhen Li, Yanyan Shen, and Yanmin Zhu. 2018. Next Point-of-Interest Recommendation with Temporal and Multi-level Context Attention. In *ICDM*.
- [21] Xutao Li, Gao Cong, Xiao-Li Li, Tuan-Anh Nguyen Pham, and Shonali Krishnaswamy. 2015. Rank-GeoFM: A ranking based geographical factorization method for point of interest recommendation. In *SIGIR*. 433–442.
- [22] Xin Li, Dongcheng Han, Jing He, Lejian Liao, and Mingzhong Wang. 2019. Next and Next New POI Recommendation via Latent Behavior Pattern Inference. *TOIS* 37, 4 (2019), 1–28.
- [23] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. 2014. GeoMF: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In *SIGKDD*. 831–840.
- [24] Qi Liu, Maximilian Nickel, and Douwe Kiela. 2019. Hyperbolic graph neural networks. In *NeurIPS*. 8228–8239.
- [25] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts. In *AAAI*.
- [26] Xin Liu, Yong Liu, Karl Aberer, and Chunyan Miao. 2013. Personalized point-of-interest recommendation by mining users' preference transition. In *CIKM*.
- [27] Xin Liu, Yong Liu, and Xiaoli Li. 2016. Exploring the Context of Locations for Personalized Location Recommendations. In *IJCAI*. 1188–1194.
- [28] Yiding Liu, Tuan-Anh Nguyen Pham, Gao Cong, and Quan Yuan. 2017. An experimental evaluation of point-of-interest recommendation in location-based social networks. *VLDB* (2017), 1010–1021.
- [29] Yi-Shu Lu and Jiun-Long Huang. 2020. GLR: A graph-based latent representation model for successive POI recommendation. *Future Generation Computer Systems* 102 (2020), 230–244.
- [30] T Mikolov and J Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NeurIPS*. 3111–3119.
- [31] Maximilian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *NeurIPS*. 6338–6347.
- [32] Maximilian Nickel and Douwe Kiela. 2018. Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry. *ICML* (2018).
- [33] Yaogong Qiao, Xiangyang Luo, Chenliang Li, Hechan Tian, and Jiangtao Ma. 2020. Heterogeneous graph-based joint representation learning for users and POIs in location-based social network. *Information Processing & Management* (2020).
- [34] Erzsébet Ravasz and Albert-László Barabási. 2003. Hierarchical organization in complex networks. *Physical review E* 67, 2 (2003), 026112.
- [35] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*.
- [36] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized Markov chains for next-basket recommendation. In *WWW*.
- [37] Jiayi Tang and Ke Wang. 2018. Personalized top-n sequential recommendation via convolutional sequence embedding. In *WSDM*.
- [38] Abraham A Ungar. 2005. *Analytic hyperbolic geometry: Mathematical foundations and applications*. World Scientific.
- [39] Lucas Vinh Tran, Yi Tay, Shuai Zhang, Gao Cong, and Xiaoli Li. 2020. HyperML: A Boosting Metric Learning Approach in Hyperbolic Space for Recommender Systems. *WSDM* (2020).
- [40] Hao Wang, Huawei Shen, Wentao Ouyang, and Xueqi Cheng. 2018. Exploiting POI-Specific Geographical Influence for Point-of-Interest Recommendation. In *IJCAI*. 3877–3883.
- [41] Pengfei Wang, Jiafeng Guo, Yanyan Lan, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2015. Learning hierarchical representation model for nextbasket recommendation. In *SIGIR*. 403–412.
- [42] Xiao Wang, Yiding Zhang, and Chuan Shi. 2019. Hyperbolic Heterogeneous Information Network Embedding. *AAAI*.
- [43] Min Xie, Hongzhi Yin, Hao Wang, Fanjiang Xu, Weitong Chen, and Sen Wang. 2016. Learning graph-based poi embedding for location-based recommendation. In *CIKM*. 15–24.
- [44] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. 2014. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2014).
- [45] Jihang Ye, Zhe Zhu, and Hong Cheng. 2013. What's Your Next Move: User Activity Prediction in Location-based Social Networks. In *SDM*. 171–179.
- [46] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee. 2011. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*.
- [47] Haochao Ying, Jian Wu, Guandong Xu, Yanchi Liu, Tingting Liang, Xiao Zhang, and Hui Xiong. 2019. Time-aware metric embedding with asymmetric projection for successive POI recommendation. *World Wide Web* 22, 5 (2019), 2209–2224.
- [48] Quan Yuan, Gao Cong, Zongyang Ma, Aixun Sun, and Nadia Magnenat Thalmann. 2013. Time-aware Point-of-interest Recommendation. In *SIGIR*. 363–372.
- [49] Quan Yuan, Gao Cong, and Aixun Sun. 2014. Graph-based point-of-interest recommendation with geographical and temporal influences. In *CIKM*. 659–668.
- [50] Jia-Dong Zhang, Chi-Yin Chow, and Yanhua Li. 2014. LORE: Exploiting sequential influence for location recommendations. In *SIGSPATIAL*. 103–112.
- [51] Shuai Zhang, Yi Tay, Lina Yao, Aixun Sun, and Jake An. 2019. Next Item Recommendation with Self-Attentive Metric Learning. In *AAAI*, Vol. 9.
- [52] Pengpeng Zhao, Haifeng Zhu, Yanchi Liu, Zhixu Li, Jiajie Xu, and Victor S Sheng. 2019. Where to Go Next: A Spatio-temporal LSTM model for Next POI Recommendation. *AAAI* (2019).
- [53] Shenglin Zhao, Tong Zhao, Irwin King, and Michael R Lyu. 2016. STELLAR: spatial-temporal latent ranking for successive point-of-interest recommendation. In *AAAI*.
- [54] Shenglin Zhao, Tong Zhao, Irwin King, and Michael R Lyu. 2017. Geo-teaser: Geo-temporal sequential embedding rank for point-of-interest recommendation. In *WWW*. 153–162.
- [55] Wayne Xin Zhao, Feifan Fan, Ji-Rong Wen, and Edward Y. Chang. 2018. Joint Representation Learning for Location-Based Social Networks with Multi-Grained Sequential Contexts. *TKDD* 12, 2 (2018), 22:1–22:21.
- [56] Wayne Xin Zhao, Ningnan Zhou, Aixun Sun, Ji-Rong Wen, Jialong Han, and Edward Y Chang. 2018. A time-aware trajectory embedding model for next-location recommendation. *Knowledge and Information Systems* (2018), 1–21.