



Next Point-of-Interest Recommendation with Inferring Multi-step Future Preferences

Thirty-First International Joint Conference on Artificial Intelligence (IJCAI-22)

汇报人: 吕思索

2022.09.26



1. 研究背景

2. 问题定义

3. CFPRec

4. 实验部分

5. 结论



1. 研究背景

2. 问题定义

3. CFPRec

4. 实验部分

5. 结论

研究背景

- Next Point-of-Interest:

- 本质上是利用用户的历史签到和其他多模态信息，例如POI属性和社交网络，来推荐下一个适合用户的POI
- 目前该问题的影响因素：Sequential Effect, Semantic Effect, Geographical Influence, Temporal Influence和Social Influence

- 本文的写作动机:

- 用户的下一步行为除了受到过去和当前的行为的影响之外，还可能受到未来多步

行为的影响，因为用户可能在开始时就已经考虑了整体的活动计划

- Figure 1

- 已有工作所存在的问题:

- 完全忽略了未来行为对next-POI推荐的潜在影响



Figure 1: An example of Alice's two daily trajectories.



1. 研究背景

2. 问题定义

3. CFPRec

4. 实验部分

5. 结论

问题定义

- Problem Formulation:

- $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ 表示一组用户, $\mathcal{L} = \{l_1, l_2, \dots, l_{|\mathcal{L}|}\}$ 表示一组POI, $\mathcal{C} = \{c_1, c_2, \dots, c_{|\mathcal{C}|}\}$ 表示一组POI的类别, $\mathcal{T} = \{t_1, t_2, \dots, t_{24}, t_{w_0}, t_{w_1}\}$ 是一组时隙 (其中将一天映射为24小时), 工作日为 t_{w_0} , 周末为 t_{w_1}
- 每一次的签到记为 $r = (u, l, c, g, t)$, 其中 g 为签到位置 (longitude, latitude), $\mathbf{t} = \mathbf{t}_k \oplus \mathbf{t}_{w_0} / \mathbf{t}_{w_1}$
- 对每一个用户 u , $S^u = \{S_1^u, S_2^u, \dots, S_n^u\}$ 记为其轨迹 (即签到序列), $S_i^u = \{r_1, r_2, \dots, r_{|S_i^u|}\}$ 表示一天内按时间顺序排列的一组签到记录
- $S_n^u = \{r_1, r_2, \dots, r_k\}$ 认为是用户的最新轨迹 S_{cur}^u (short-term behaviors), $\{S_1^u, \dots, S_{n-1}^u\}$ 被认为是过往的行为 S_{past}^u (long-term behaviors)

- Research Problem:

- 给出 S_{cur}^u 和 S_{past}^u
- 通过推断用户 u 在 t_{k+1} 至 t_T 的多步未来偏好, 从而推荐一组 POI 供用户 u 在 t_{k+1} 访问



1. 研究背景

2. 问题定义

3. CFPRec

4. 实验部分

5. 结论

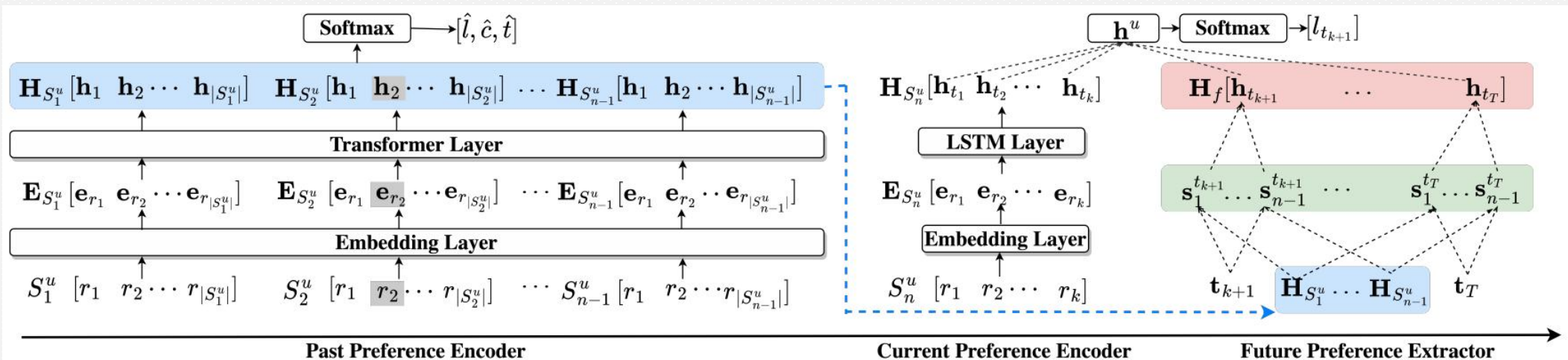
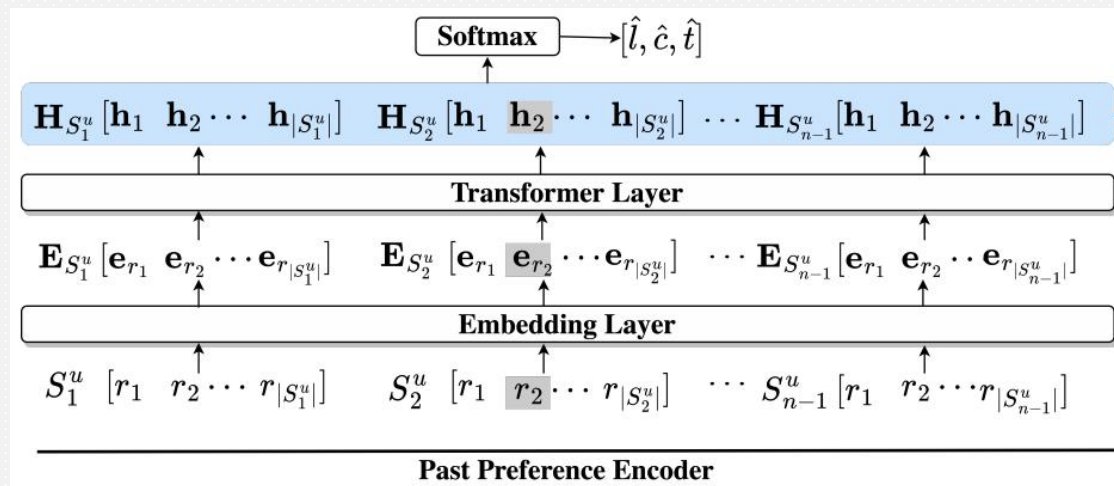


Figure 2: The overall architecture of our proposed CFPRec, which consists of past and current preference encoders as well as the future preference extractor; and the rectangles in grey, pink and green color represent the masked record, inter-sequence attention aggregation and intra-sequence attention aggregation, respectively.

CFPRec:

- Figure 2
- Past Preference Encoder: 对用户过去的长期轨迹 (S_{past}^u) 进行建模得 $\{H_{S_1^u}, \dots, H_{S_{n-1}^u}\}$, 以捕捉非连续签到行为之间的上下文感知相关性
- Current Preference Encoder: 通过对最近的顺序签到 (S_{cur}^u) 建模来学习用户当前的偏好
- Future Preference Extractor: 将过去偏好的隐藏状态矩阵 $\{H_{S_1^u}, \dots, H_{S_{n-1}^u}\}$ 输入两层 Attention Aggregation, 用于推断多步未来偏好

Past Preference Encoder



- **Past Preference Encoder :**

- 采用bidirectional Transformer

- Transformer 可以对目标 POI 的前后上下文进行建模, 使得对POI的学习更具表现力
 - 可以捕获非连续签到之间的上下文相关性, 并在轨迹自动聚合最相关的行为

- **EmbeddingLayer :**

- 给定用户 u 的过去轨迹: $S_i^u = \{r_1, r_2, \dots, r_{|S_i^u|}\}$
 - 对每个签到行为 r_i 表示为: $\mathbf{e}_{r_i} = \mathbf{u} \oplus \mathbf{l} \oplus \mathbf{c} \oplus \mathbf{t}, \mathbf{e}_{r_i} \in \mathbb{R}^{5D}$
 - S_i^u 从而被嵌入层转化为: $\mathbf{E}_{S_i^u} = [\mathbf{e}_{r_1}, \mathbf{e}_{r_2}, \dots, \mathbf{e}_{r_{|S_i^u|}}]$

Past Preference Encoder

- Transformer Layer:

- 将 $E_{S_i^u}$ 馈入 Transformer 层, 得到由隐藏向量组成的输出矩阵:

$$\begin{aligned}\mathbf{H}_{S_i^u} &= [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_{|S_i^u|}] = \text{TransLayer}(\mathbf{E}_{S_i^u}) \\ &= \text{TransLayer}(\mathbf{E}_{S_i^u} \mathbf{W}_Q, \mathbf{E}_{S_i^u} \mathbf{W}_K, \mathbf{E}_{S_i^u} \mathbf{W}_V, \Delta^{dist}) \\ \text{TansLayer}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \Delta^{dist}) &= \left(F\left(\frac{\mathbf{Q}\mathbf{K}^\top + \Delta^{dist}}{\sqrt{5D}}\right) \right) \mathbf{V}\end{aligned}$$

where $F(\cdot)$ is the softmax function; $\text{TransLayer}(\cdot)$ is a Transformer layer; $\mathbf{H}_{S_i^u}$ is the output matrix consisting of the hidden vectors and $\mathbf{h}_i \in \mathbb{R}^{5D}$; $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{5D \times 5D}$ are transformation weight matrices of query, key and value, respectively; $\Delta^{dist} \in \mathbb{R}^{|S_i^u| \times |S_i^u|}$ is the spatial relation matrix; and given the distance $\text{dist}(l_i, l_j)$ derived from g_{l_i} and g_{l_j} , $\Delta_{i,j}^{dist} = (1 + \text{dist}(l_i, l_j))^{-1}$ measures the spatial influence between locations l_i and l_j .

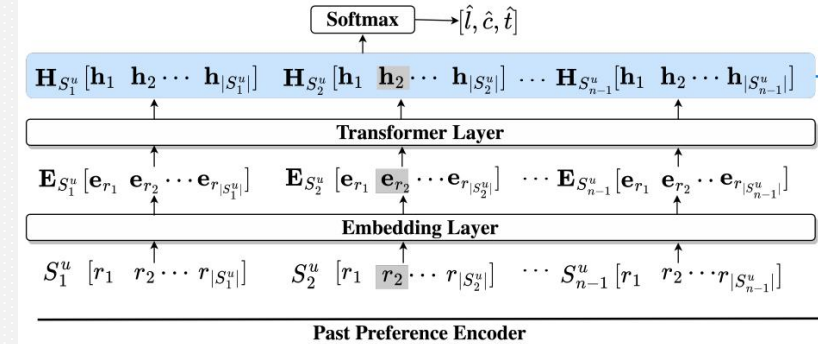
- 掩码训练: 受最近构建自我监督训练目标以实现掩码标记预测的研究的启发, 设计了三个辅助任务来进一步监督过去的偏好学习并帮助捕获更准确的偏好表示:

- 给定一个序列 S_i^u , 随机屏蔽 20% 的签到记录 $r_m = (\bar{u}, l_m, c_m, \bar{g}_m, t_m)$, (对应的嵌入向量和隐藏状态分别用 e_m 和 h_m 表示)
- 通过 softmax 层解码隐藏向量 h_m , 对原始位置 l 、类别 c 和时间 t 进行多任务预测:

$$\hat{l} = F(\mathbf{h}_m \mathbf{W}_l), \hat{c} = F(\mathbf{h}_m \mathbf{W}_c), \hat{t} = F(\mathbf{h}_m \mathbf{W}_t), \quad (\text{其中 } W_l, W_c, W_t \text{ 是这三个任务对应的变换矩阵})$$

- 辅助任务的损失函数 (负对数似然函数) 是:

$$L^{aux} = L_l^{aux} + L_c^{aux} + L_t^{aux} = - \sum_{k \in N_m} \log(\hat{l}_k) + \log(\hat{c}_k) + \log(\hat{t}_k), \quad (N_m \text{ 是所有被屏蔽记录的集合})$$



Current Preference Encoder

- Current Preference Encoder:

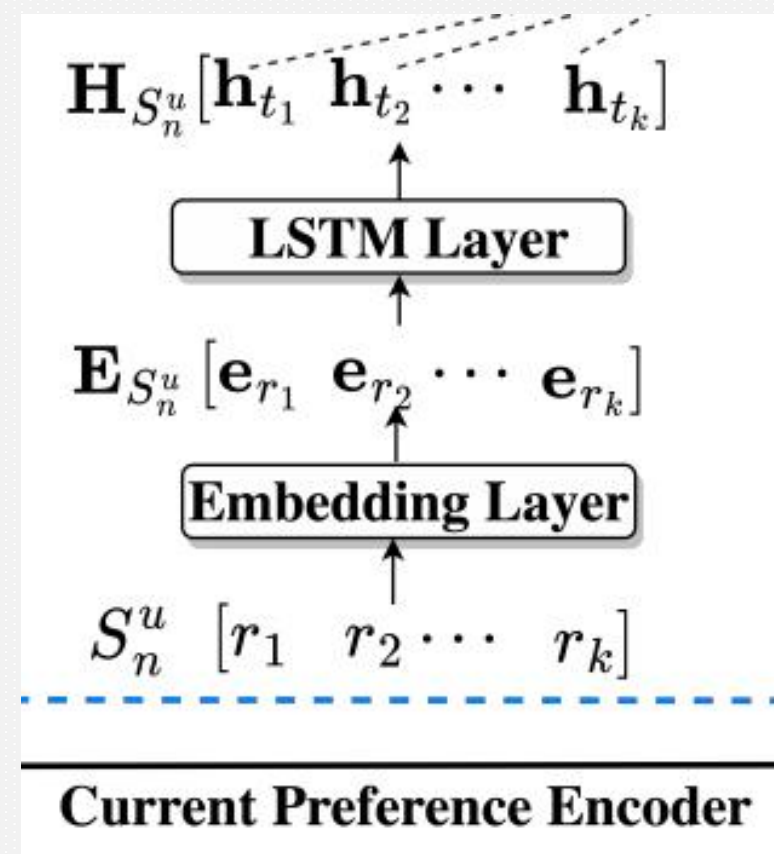
- 使用 LSTM 单元来表征用户对当前签到行为的时间感知顺序依赖性

- Embedding Layer:

- 给定用户 u 的当前轨迹 S_n^u
- 对每个签到行为 r_i 表示为: $\mathbf{e}_{r_i} = \mathbf{u} \oplus \mathbf{l} \oplus \mathbf{c} \oplus \mathbf{t}$, $\mathbf{e}_{r_i} \in \mathbb{R}^{5D}$
- 得到嵌入的当前轨迹: $\mathbf{E}_{S_n^u} = [\mathbf{e}_{r_1}, \mathbf{e}_{r_2}, \dots, \mathbf{e}_{r_k}]$

- LSTM Layer:

- 输入LSTM层: $\mathbf{h}_{t_i} = \text{LSTM}(\mathbf{e}_{r_i}, \mathbf{h}_{t_{i-1}})$, $i \in \{1, 2, \dots, k\}$, 其中 \mathbf{h}_{t_i} 和 $\mathbf{h}_{t_{i-1}}$ 分别是对应于时间戳 t_i 和 t_{i-1} 的隐藏状态
- 最终LSTM 层输出针对用户当前偏好的隐藏状态序列: $\mathbf{H}_{S_n^u} = [\mathbf{h}_{t_1}, \mathbf{h}_{t_2}, \dots, \mathbf{h}_{t_k}]$



Future Preference Extractor

- Future Preference Extractor

- 由用户日常签到行为的周期性特性从而推导出一个提取器：该提取器配备两层注意力聚合，以推断多步隐式的未来偏好，从而在用户下次访问之前隐式地模仿用户的活动计划

- Intra-sequence Attention Aggregation (序列内) -Green

- 由用户的周期性属性，设计了一种时间感知注意力来识别过去轨迹中与未来时间背景最相关的行为

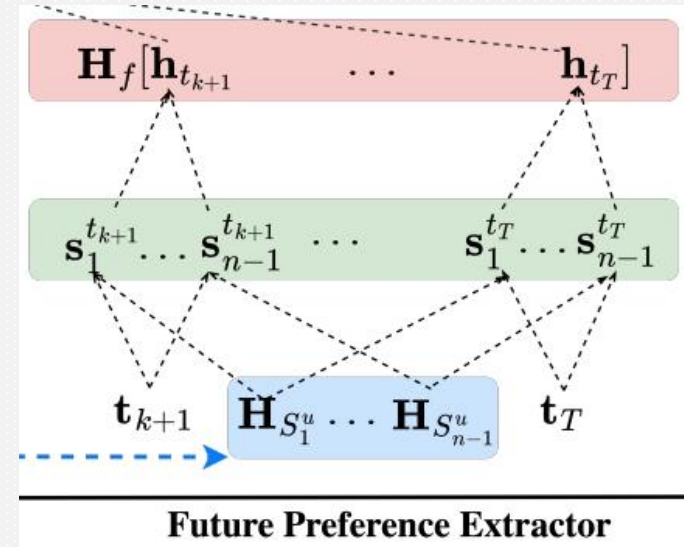
- 由过去偏好编码器已获得： $\mathbf{H}_{S_i^u} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_{|S_i^u|}]$ (隐藏向量 \mathbf{h}_i 表征用户的时空感知动态POI偏好和静态活动偏好)

- 未来时间戳 t_f 用作查询向量，由过去轨迹的每个隐藏状态，可得未来时间轨迹嵌入 $(S_i^{t_f})$ 为：

$$\mathbf{s}_i^{t_f} = \sum_{i=1}^{|S_i^u|} \alpha_i \mathbf{h}_i, t_f \in \{t_{k+1}, t_{k+2}, \dots, t_T\}$$

$$\alpha_i = \frac{\exp(\mathbf{t}_f^\top \mathbf{h}_i)}{\sum_{i'=1}^{|S_i^u|} \exp(\mathbf{t}_f^\top \mathbf{h}_{i'})}$$

- 即 $S_i^{t_f}$ 通过从过去的轨迹 S_i^u 中提取与未来时间戳 t_f 最相关的行为来表示潜在的未来行为



Future Preference Extractor

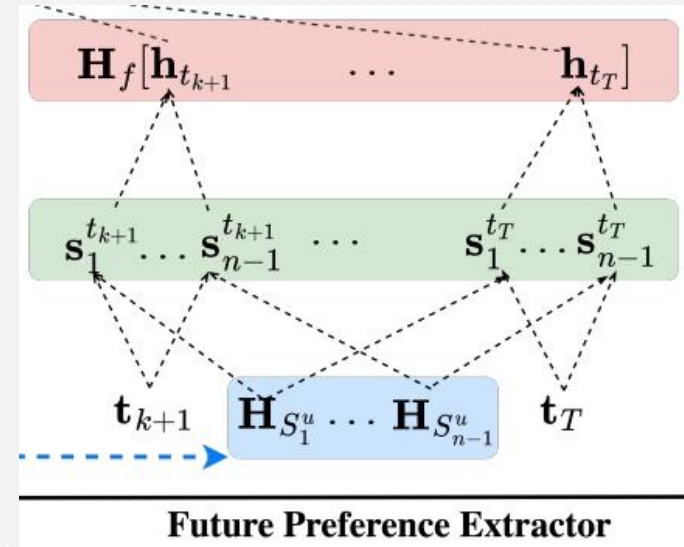
- Inter-sequence Attention Aggregation (序列间) -Pink
 - 提出了一个序列间注意层来模拟跨不同轨迹的顺序偏好进化过程
 - 上述获得多个关于 t_f 的未来时间轨迹嵌入 $\{\bar{s}_1^{t_{k+1}}, \dots, \bar{s}_{n-1}^{t_{k+1}}\}$
 - 使用动态用户嵌入 u 作为查询向量来关注相关的轨迹偏好:

$$\mathbf{h}_{t_f} = \sum_{i=1}^{n-1} \beta_i \mathbf{s}_i^{t_f}, t_f \in \{t_{k+1}, t_{k+2}, \dots, t_T\}$$

$$\beta_i = \frac{\exp(\mathbf{u}^\top \mathbf{s}_i^{t_f})}{\sum_{i'=1}^{n-1} \exp(\mathbf{u}^\top \mathbf{s}_{i'}^{t_f})},$$

其中 β_i 是attention score, \mathbf{h}_{t_f} 指的是时间戳 t_f 处的未来偏好

- 综上, 未来偏好提取器实现了多步未来偏好 (即 $\mathbf{H}_f = [\mathbf{h}_{t_{k+1}}, \mathbf{h}_{t_{k+2}}, \dots, \mathbf{h}_{t_T}]$) 的推断, 同时也继承了用户过去的偏好



Model Training

• Model Training

- 获得表征用户偏好的隐藏状态 $\mathbf{H}_{S_n^u, f} = [\mathbf{H}_{S_n^u}, \mathbf{H}_f]$
- 通过聚合 $H_{S_n^u}$ 中的每个隐藏状态来制定最终的用户偏好 (h^u):

$$\mathbf{h}^u = \sum_{i=1}^T \omega_i \mathbf{h}_{t_i}, \mathbf{h}_{t_i} \in \mathbf{H}_{S_n^u, f}$$

$$\omega_i = \frac{\exp(\mathbf{u}^\top \mathbf{h}_{t_i})}{\sum_{i'=1}^T \exp(\mathbf{u}^\top \mathbf{h}_{t_{i'}})},$$

其中 ω_i 是动态用户嵌入 \mathbf{u} 和每个隐藏状态 \mathbf{h}_{t_i} 之间的attention score

- 使用softmax函数从学习到的用户偏好 h^u 解码 $|L|$ 上POI的概率分布

$$\hat{\mathbf{y}} = F(\mathbf{h}^u \mathbf{W}_o)$$

其中 $\hat{\mathbf{y}}$ 是在 t_{k+1} 的预测概率分布, \mathbf{w}_o 是一个变换矩阵

- 目标 POI 预测的目标函数为: $L^{poi} = -\sum_{i \in \mathcal{N}} \log(\hat{y}_i),$

其中 \mathcal{N} 是训练样本集; \hat{y}_i 是第 i 个训练样本对应的 ground truth POI 的预测概率

- 总损失函数为 $L = L^{poi} + \eta L^{aux},$

Algorithm 1: Training of the CFPRec

Input: $\mathcal{U}, \{S^{u_1}, S^{u_2}, \dots, S^{u_{|\mathcal{U}|}}\}, max_iter, \eta, D, T, \gamma$

Output: Trained model $\{\Theta\}$

```

1 Initialize model parameters  $\Theta, \mathcal{N} = \emptyset;$ 
  // Training instance construction
2 foreach  $u \in \mathcal{U}$  do
3   foreach  $S_i^u \in S^u$  do
4      $S_{past}^u = \{S_1^u, S_2^u, \dots, S_{n-1}^u\}, S_{cur}^u = S_n^u$ 
5      $\{S_{past}^u, S_{cur}^u\} \mapsto \mathcal{N}$ 
  // Update parameters  $\Theta$ 
6 foreach  $iter \in \{1, 2, \dots, max\_iter\}$  do
7   Select instance  $\{S_{past}^u, S_{cur}^u\}$  from  $\mathcal{N}$ 
8   Update  $\Theta$  by minimizing the objective in Eq.(12)
```




1. 研究背景
2. 问题定义
3. CFPRec
- 4. 实验部分**
5. 结论

Experiments

- 本文的实验主要研究了三个问题:

- 与baseline进行对比
- 消融实验
- 超参数分析

- 数据集 (Table 1), 根据时间戳将每个用户的轨迹按照 8:1:1 的比例拆分

- 评价指标:

- HitRate@K (HR@K): 是一种常用的衡量召回率的指标
- NDCG@K: 用作排序结果的评价指标, 评价排序的准确性

- Baseline

- MostPop
- BPRMF: Bpr:Bayesian personalized ranking from implicit feedback. In UAI, 2009
- ST-RNN:Predicting the next location: A recurrent model with spatial and temporal contexts. In AAAI, 2016
- ATST-LSTM:An attention-based spatiotemporal lstm network for next poi recommendation. TSC, 2019
- MCARNN:Predicting activity and location with multi-task context aware recurrent neural network. In IJCAI, 2018
- PLSPL:Personalized long-and short-term preference learning for next poi recommendation. TKDE, 2020
- iMTL:An interactive multi-task learning framework for next poi recommendation with uncertain check-ins. In IJCAI, 2021
- CTLE:Pre-training context and time aware location embeddings from spatial-temporal trajectories for user next location prediction. In AAAI, 2021.

	# User	# POI	# Check-in	Density
SIN	8,648	33,712	355,337	0.12%
NYC	16,387	56,252	511,431	0.06%
PHO	2,946	7,247	47,980	0.22%

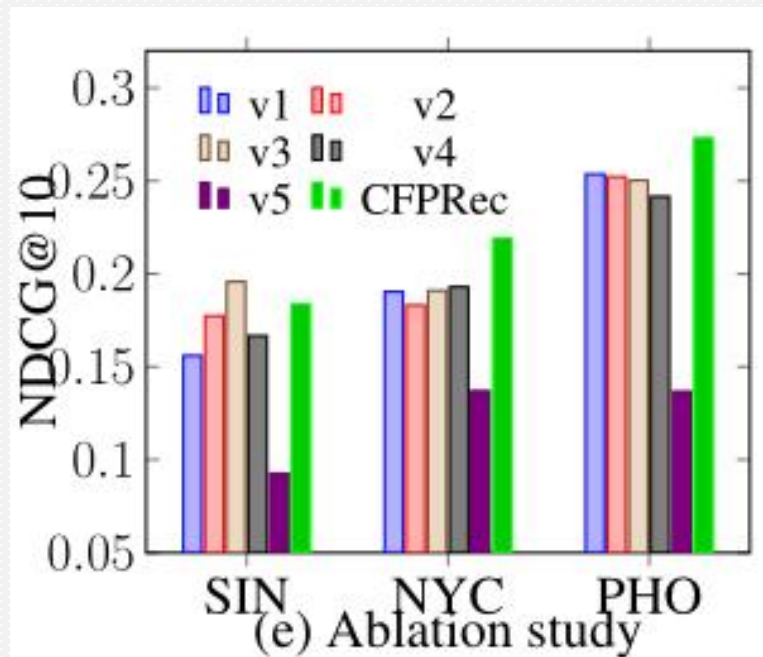
Table 1: Statistics of the three real-world datasets.

Compare with Baseline

	SIN				NYC				PHO			
	HR@5	HR@10	NDCG@5	NDCG@10	HR@5	HR@10	NDCG@5	NDCG@10	HR@5	HR@10	NDCG@5	NDCG@10
MostPop	0.0125	0.0293	0.0106	0.0151	0.0139	0.0235	0.0101	0.0145	0.0160	0.0223	0.0114	0.0131
BPRMF	0.0352	0.0525	0.0222	0.0380	0.0381	0.0468	0.0239	0.0339	0.0487	0.0585	0.0256	0.0304
ST-RNN	0.0959	0.1370	0.0655	0.0794	0.1347	0.1826	0.0593	0.1303	0.1240	0.2028	0.0802	0.1229
ATST-LSTM	0.1296	0.1933	0.1027	0.1476	0.1667	0.2031	0.0912	0.1638	0.1579	0.2377	0.1033	0.1385
MCARNN	0.1608	0.1862	0.1169	<u>0.1591</u>	0.1835	0.2397	0.1036	<u>0.1870</u>	0.1905	0.2726	0.1264	0.1617
iMTL	0.1505	0.1801	0.1051	0.1423	0.1798	0.2422	0.0989	0.1861	0.1830	0.2747	0.1301	0.1632
PLSPL	0.1447	0.1719	0.1126	0.1384	0.1741	0.2413	0.0961	0.1825	0.1775	0.2569	0.1285	0.1538
CTLE	<u>0.2041</u>	<u>0.2784</u>	<u>0.1315</u>	0.1556	<u>0.2421</u>	<u>0.3205</u>	<u>0.1513</u>	0.1841	<u>0.2632</u>	<u>0.3605</u>	<u>0.1995</u>	<u>0.2068</u>
CFPRec	0.2310	0.3085	0.1588	0.1836	0.2771	0.3606	0.1971	0.2190	0.3421	0.4253	0.2432	0.2730
Improve	13.2%	10.8%	11.2%	15.4%	14.5%	12.5%	30.3%	19.0%	30.0%	17.5%	21.9%	32.0%

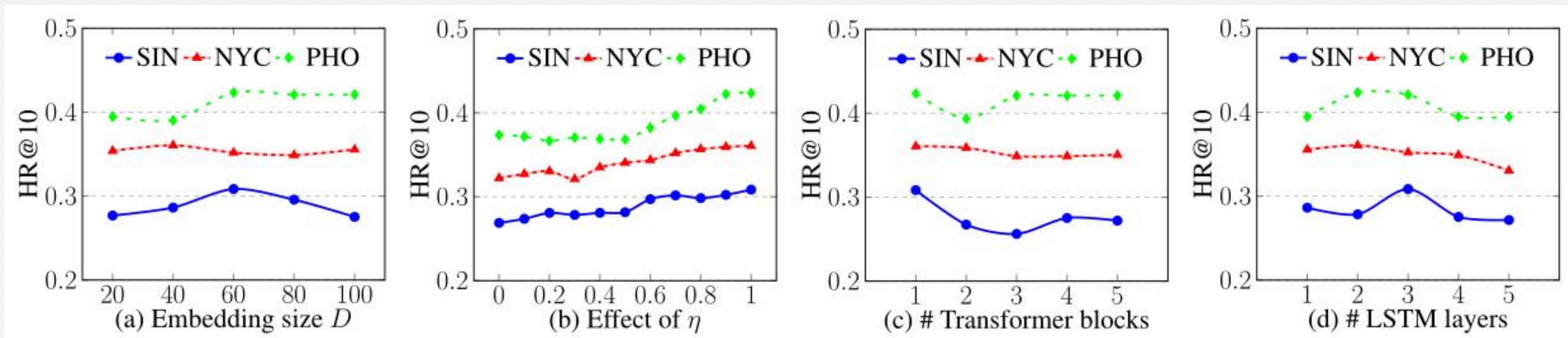
Table 2: Performance of all methods on three datasets, where the best performance is boldfaced; the runner up is underlined; the row ‘Improve’ indicates the improvements achieved by CFPRec relative to the runner up, whose significance is determined by a paired t-test with $p < 0.01$.

Ablation Study



- 通过与CFPRec 的五种变体进行比较来探索不同组件对 CFPRec 的功效：
 - (v1) 去除过去偏好编码器
 - (v2) 去除未来偏好提取器
 - (v3) 仅在 t_{k+1} 采用一步未来偏好
 - (v4) 将未来偏好提取器中的两层Attention Aggregation替换为平均池化
 - (v5) 去除过去的偏好编码器和未来的偏好提取器，即只采用 LSTM 来建模用户的顺序行为以进行推荐

Parameter Sensitivity Analysis



- 检查关键参数对 CFPRec 的影响：
 - 随着嵌入大小 D 的增加，CFPRec 的性能先上升，然后达到峰值，最后在所有数据集上保持稳定或略有下降
 - CFPRec 在所有数据集上的性能随着 η 的增加而逐渐提高，最佳设置为1
 - CFPRec 的最佳性能是在所有数据集上仅使用一个transformer块实现的，这表明更多的transformer块可能并不能保证更好的结果
 - 所有数据集上 LSTM 层数对性能的影响



1. 研究背景

2. 问题定义

3. CFPRec

4. 实验部分

5. 结论

Conclusion

- 提出了一种新方法 CFPreC，用于联合建模用户过去、当前和未来偏好对next-POI的影响
- 具体来说：
 - 设计了过去偏好编码器来通过 Transformer 层对用户过去的偏好进行建模，该层进一步采用三个辅助目标来帮助监督从过去轨迹中学习的用户偏好
 - 当前偏好编码器旨在通过 LSTM 层对用户的最新顺序行为进行建模
 - 通过配备两层注意力聚合来设计未来偏好提取器，以推断多步未来偏好



Thank you!