LMP: Large Language Model Enhanced Intent-aware Mobility Prediction

Anonymous ACL submission

Abstract

Human mobility behavior prediction is essential for applications like urban planning and transportation management, yet it remains challenging due to the complex, often implicit, intentions behind human behavior. Recent advancements in large language models (LLMs) offer a promising alternative research angle for integrating commonsense reasoning into human mobility behavior analysis. However, it is a non-trivial problem because LLMs are not natively built for mobility intention inference, and they also face scalability issues and integration difficulties with spatiotemporal models. To address these challenges, we propose a novel LMP (LLMs for Mobility Prediction) framework. Specifically, LMP integrates a schema learning-based agentic workflow for LLM-driven mobility intention inference, a data-efficient fine-tuning scheme for scalable knowledge distillation, and a transformer-based intent-aware model for final efficient mobility prediction. Evaluated on three real-world datasets, LMP outperforms state-of-the-art baselines on 11 out of 12 metrics and ranks as the second-best method on the remaining one, demonstrating improved accuracy in next-location prediction and effective intention inference. Data and codes are available via https://anonymous.4open.science/r/LMP-1D4B.

1 Introduction

Predicting human mobility behavior is a crucial task with significant implications for various domains, including urban planning, transportation management, and public safety. However, the inherent complexity of human mobility poses substantial challenges, especially the implicit intentions that are often not directly observable. Previous studies have shown that human researchers can infer the intention of human movements with high accuracy by examining their spatiotemporal trajectory (Jiang et al., 2016; Liccardi et al., 2016). However, it is not scalable to ask human researchers to manually label mobility data. Thus, most of existing mobility prediction models (Liu et al., 2016; Feng et al., 2018; Sun et al., 2020; Luo et al., 2021; Yang et al., 2022) focus on capturing spatiotemporal patterns using advanced recurrent network and attention models. While these methods have shown promise, they fail to effectively model the underlying intentions that drive each movement. This limitation highlights the need for new methods that can incorporate a deeper understanding of human behavior.

Recent advancements in LLMs have demonstrated emergent capabilities in commonsense reasoning (Wei et al., 2022a,b), offering a novel research angle for intention-aware mobility prediction. Despite this promise, several challenges remain in leveraging LLMs for mobility prediction. First, LLMs are not inherently optimized for inferring behavioral intentions from spatiotemporal data. Second, the massive size and proprietary nature of state-of-the-art LLMs, such as GPT-4 (Achiam et al., 2023), present practical challenges, including high API costs and the inability to deploy these models locally. Third, the domain-specific nature of spatiotemporal deep learning models and LLMs creates a disconnect, making it unclear how to effectively integrate the two to enhance prediction accuracy.

In response to these challenges, we propose a novel framework, LMP (LLMs for Mobility Prediction), designed to harness the commonsense reasoning abilities of LLMs for intention-aware mobility prediction. The framework comprises three key components. First, we introduce an Schema Learning-based Agentic Workflow that guides LLMs through the process of mobility intention inference in a principal manner, which emulates the methodology of human expert annotators. The workflow enables LLMs to reason through the intentions behind movements step-by-step: analyzing notable features, in-context schema learn-

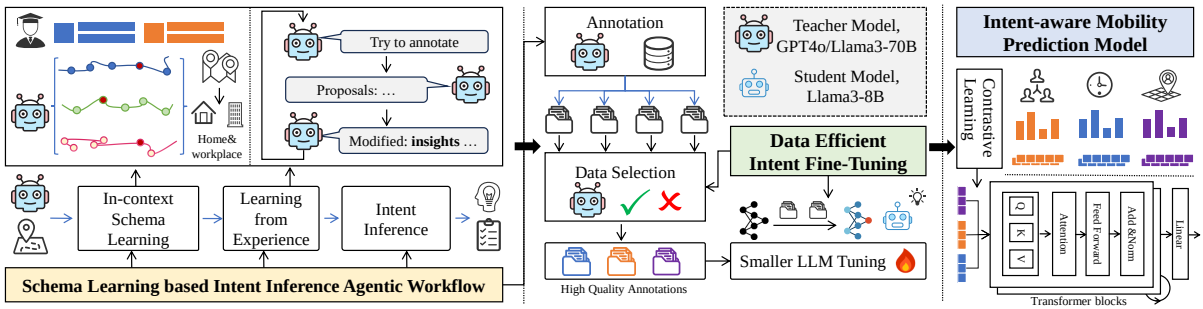


Figure 1: The framework of LMP, including schema learning based agentic workflow for mobility intent annotation, a data-efficient intent fine-tuning approach with data selection, and a transformer based intent-aware mobility prediction model enhanced with contrastive learning.

ing, and inferring the most likely intention from experience. Second, we present a data-efficient fine-tuning scheme with data selection, effectively distilling the reasoning capabilities of large proprietary LLMs, such as GPT-4o, into smaller, locally deployable models like Llama3-8B (Dubey et al., 2024). This approach ensures that our framework can scale to handle millions of mobility records at low cost and high speed. Finally, we design a transformer-based intent-aware mobility prediction model that seamlessly integrates inferred intentions from LLMs through contrastive learning. This approach enables efficient mobility prediction without need for real-time LLM inference, while maintaining minimal computational overhead. In summary, our contributions are fourfold,

- We introduce a novel framework, LMP, that leverages the commonsense reasoning power of LLMs for enhanced mobility prediction, incorporating intention inference to improve both performance and interpretability.
- We propose a schema-learning-based intent inference workflow and in-context inference through experience-driven learning.
- We develop a data-efficient fine-tuning strategy with data selection that obtaining high-performing, cost-effective models by distilling reasoning capabilities from large, proprietary LLMs to smaller, open-source alternatives.
- We conduct extensive empirical evaluations on three real-world datasets, demonstrating LMP's strong robustness and practical applicability.

2 Methods

We define the mobility prediction task as follows: for a specific user u in the user list, given the historical POI sequence $(l_1, l_2, \ldots, l_n)$, visit times $(t_1, t_2, \ldots, t_n)$, POI category sequence $(c_1, c_2, \ldots, c_n)$, and the next visit time t_{n+1}, predict the next POI l_{n+1} that the user will visit. Our approach aims to assign an intent I to each stay l within the training data trajectories. Subsequently, for each user u, we integrate the intent annotations to obtain the user intent profile $P_u(t)$, and use $P_u(t)$ as a new feature to assist in mobility prediction.

As Figure 1 shows, LMP contains three main steps: Schema Learning based Intent Inference Agentic Workflow, Data Efficient Intent Fine-Tuning, and Intent-Aware Mobility Prediction Model. We divide the training data into three parts: *human labeled data*, a very small dataset manually annotated with intent; *fine-tuning data*, a smaller dataset used for fine-tuning; and *main data*, the remaining majority of the data. The Schema Learning based Intent Inference Agentic Workflow introduces an effective intent inference method using *human labeled data* to guide large-scale models such as GPT-4o. In Data Efficient Intent Fine-Tuning, we fine-tune small-scale models like Llama3-8B by using the inference process of large-scale models on the *fine-tuning data* to achieve efficient intent inference models. And then we use the fine-tuned small-scale models for intent inference on the *main data*. In the Intent-Aware Mobility Prediction Model section, we generate the user intent profile using all the training data and use it to assist in mobility prediction.

2.1 Schema Learning-based Workflow

In the complex task of intent inference, LLMs do not perform well in reasoning and transferring across different datasets. Meanwhile, human experts have accumulated extensive experience in handling trajectory data tasks (Zeng et al., 2017; Chen and Poorthuis, 2021; Liccardi et al., 2016; Jiang et al., 2016). This leads us to consider the schema-based learning approach. Schema-based learning is an important concept in human cognitive psychology. For unfamiliar situations, schemas allow individuals to interpret and reason based on

their generalized knowledge (Lee and Seel, 2012). Based on this, we designed a Schema Learning-based Agentic Workflow, which creatively incorporates human expert knowledge into the reasoning process of LLMs by extracting schemas from human experiences and human-generated expert data. This enhances the LLM's ability for in-context learning and reasoning during intent inference.

To facilitate the following discussion, we define the intent inference task as follows: We use l to represent a point of interest (POI) that a user has visited. The intent inference task is: given a user's trajectory $(l_1, l_2, l_3, \ldots)$, each trajectory point is annotated with its behavioral intent, forming an intent sequence $(I_1, I_2, I_3, \ldots)$ corresponding one-to-one with the trajectory sequence. Each intent is selected from several options, and in our approach, there are six intents: "At Home", "Working", "Running errands", "Eating out", "Leisure and Entertainment", and "Shopping".

2.1.1 In-context Schema Learning

For different users, the characteristics of their routine place, such as home and workplace, may differ significantly, which poses challenges in inferring intents like "At Home" and "Working". In this section, we draw on the experience of human experts to design a schema that classifies POIs into three categories: home, workplace, and other. Through workflow design, this schema is integrated into the in-context learning process of LLMs, effectively addressing the aforementioned challenges. First, based on expert experience, we extract the following features from each user's data to represent their behavior patterns: 1) Percentage, for each POI, we calculate the proportion of visits by the user relative to their total visits; 2) Time Distribution, for a specific POI, we compute the proportion of visits during different time periods relative to the total visits to that POI. Using these key features, the LLM distinguishes the user's home and workplace POIs to assist with intent inference.

2.1.2 Learning from Experience

LLM can learn from historical experiences to better guide task completion (Zhao et al., 2024). We enable the LLM to learn the schema of human reasoning intent from *human-labeled data* and represent it through several insights. Through this approach, we transfer humans' efficient reasoning capabilities to the LLM in the form of a schema, allowing the LLM's performance in intent inference tasks to approximate that of humans. This workflow consists of the following iterative steps:

- Intent Annotation Attempt: We provide the LLM with the current insights, a sequence of user visits to POIs, and the user's home and workplace (i.e., the POIs most frequently associated with "At Home" and "Working" intents in the manually labeled data). The LLM is tasked with labeling the intent of each visit in the trajectory sequence.
- Correction Suggestion Generation: We provide the LLM with the current insights and several groups of the user's POI visit sequence, the user's home and workplace, the intent results from the LLM's previous step, and the manual intent labels. The LLM analyzes the differences between its annotations and the human annotations to generate suggestions for correcting the insights.
- Insights Correction: The LLM extracts commonalities from several correction suggestions and uses these to update the insights accordingly.

Finally, we used in-context learning (ICL) to synthesize the schemas into a cohesive intent inference process. For each stay within the user's daily trajectory, LLM infers the user's intent by considering the identified home and workplace locations, the sequence of POIs, the time of day and the insights. By using ICL, the model could apply its understanding of typical behaviors to infer intents in novel situations.

2.2 Data-Efficient Intent Instruction Tuning

For large datasets, using large-scale models such as GPT-4o for intentions inference incurs excessive costs and lower speed, making it inefficient to annotate the entire dataset directly. However, smaller models such as the original Llama3-8B perform poorly on related tasks and do not meet the expected performance. To address this issue, we designed a *data-efficient intent instruction tuning* method to distill the intent inference abilities from large-scale LLMs. In this method, large-scale LLMs act as *teacher model* and small-scale LLMs act as *student model*. The *teacher model* annotates intent on *fine-tuning data* and generates detailed reasoning steps, which is then further processed through a data selection module to improve the data quality. The selected high-quality reasoning data is used to fine-tune the *student model*, enabling it to achieve or even surpass the intention inference performance of the *teacher model* on specific tasks through learned reasoning strategies, while maintaining low inference costs.

2.2.1 Intent Data Synthesis and Selection

We utilized Schema Learning-based Agentic Workflow to annotate *fine-tuning data* using *teacher model*. The results are saved as the potential reasoning data for following fine-tuning. Experiments show that *teacher model* can also make errors during the annotation process, which can affect the distilling results. To address this, we added a data selection module to screen the synthetic reasoning data for intention inference, aiming to reduce the interference of incorrect annotations in the synthetic data. Given LLM's excellent performance in simulating human reasoning and judgment abilities, it excels in many judgment tasks (Gu et al., 2025). Therefore, we use *teacher model* for this data selection process, utilizing the reflection of *teacher model* to determine which data to use for further fine-tuning. Specifically, we leverage LLM's in-context learning capability by feeding the context information provided during the User Intentions Inference phase back to the model along with the model's labeling results. After a chain of thought (CoT) process, the model outputs "yes" or "no" to judge whether the annotation is correct. We remove the data judged as "no" and collect the "yes" data to include in the final fine-tuning dataset with reasoning steps on various tasks.

2.2.2 Intent Instruction Tuning

In the Schema Learning-based Agentic Workflow, we selected the in-context Schema Learning and intentions inference tasks for fine-tuning and directly incorporated the experience generated in the Schema Learning-based Agentic Workflow into the fine-tuning prompts. To further reduce inference costs, we omitted the CoT part from the original workflow and used the final results directly for fine-tuning. We use the Low-Rank Adaptation (LoRA) method to fine-tune the *student model*. After LoRA tuning, we ultimately obtained the *student model* with both high inference performance and low inference cost. And then we use the *student model* for intent inference on the *main data*.

2.3 Intent-Aware Mobility Prediction Model

In this section, we designed a model that effectively utilizes intent information to assist with mobility prediction. Specifically, we developed a method for generating user intent profiles. These user intent profiles can infer a user's potential intentions at specific times during the day. By annotating intent on the training data and generating these intent profiles, we can use them directly during inference. This approach eliminates the need for LLMs to annotate intents in post-training applications, further reducing the computational cost of our method. To better learn the user intent profiles, we designed a user intent profile-location contrastive learning module, enabling the model to better grasp the relationship between user intents and specific locations. Finally, we developed an intent-aware mobility prediction model, using a transformer architecture to perform the mobility prediction task.

2.3.1 User Intent Profile Generation

For each user u, we define their user intent profile function P_u as $P_u(t) = \mathbf{p}$, where t is the time of day, and $\mathbf{p}$ is a vector of length equal to the number of intent categories, representing the probability of each intent occurring at time t.

We developed a method using expert knowledge for this generating user intent profile, based on the idea that people often follow the same daily behavioral patterns. Specifically, for the intent sequence of a user u corresponding to all of his movements in the train dataset, $(I_1, I_2, I_3, \dots I_N)$, and their corresponding time sequence $(t_1, t_2, t_3, \dots t_N)$, we consider that an intent I_i recorded over a period provides a likelihood of this intent occurring. The influence period is defined as $t_{begin,i} = \max(t_{i-1}, t_i - T)$ and $t_{end,i} = \min(t_{i+1}, t_i + T)$, where T is a parameter representing the maximum influence time range. For each intent type $\mathbb{I}_j$, we construct the function $f_{\mathbb{I}_j}(t)$ as follows to represent the effect of the intent:

$$eff_i = \min\left(\frac{t - t_{begin,i}}{t_i - t_{begin,i}}, \frac{t_{end,i} - t}{t_{end,i} - t_i}\right),$$

$$f_{\mathbb{I}_j}(t) = \sum_{I_i = \mathbb{I}_j} \max(0, eff_i).$$

For a specific time t_0 within a day, the user intent profile function of user u and intent $\mathbb{I}_j$ is calculated as:

$$P_{u,\mathbb{I}_j}(t_0) = \frac{\sum_k f_{\mathbb{I}_j}(t_0 + k\Delta t)}{\sum_{k,j} f_{\mathbb{I}_j}(t_0 + k\Delta t)},$$

where Δt represents a time interval of a day. Then $P_u(t)$ is calculated as:

$$P_u(t) = [P_{u,\mathbb{I}_1}(t), P_{u,\mathbb{I}_2}(t) ...].$$

2.3.2 User Intent Profile Contrastive Learning

We propose a user intent profile-location contrastive learning module. This module uses contrastive learning to initialize part of the network parameters of the mobility prediction model, thereby

enhancing the model's attention to possible locations given a user's intent profile.

Specifically, we generate an embedding vector for each user and each intent. For a particular visit, we calculate the probability estimate $\mathbf{p} = P(t)$ for each intent at the time of the visit. To effectively handle the predicted probability information, we then use $\mathbf{p}$ to weight the embedding vectors of each intent, this means that the higher the probability of a certain intent, the closer the overall intent embedding is to the individual embedding of that intent. This makes it easier for the model to determine the extent to which it should rely on the intent information, resulting in a more effective combination of intent information and trajectory information. This can be expressed as $\mathbf{e}_I = \sum_{i=1}^{n} p_i \cdot \mathbf{e}_{I_i}$, where $\mathbf{e}_{I_i}$ represents the embedding vector for each intent, $\mathbf{e}_I$ is the resulting weighted vector, and p_i is the weight for each intent. The weighted intent embedding and the user embedding are fed into a user interest generator, composed of a single-layer fully connected neural network, which outputs a visit interest vector representing a user's interest under a specific intent probability distribution.

Simultaneously, we generate an embedding vector for each POI. The visit interest vector and the POI embedding vector are fed into a comparator, constructed with an multi-layer perceptron network, forming our contrastive learning architecture. For each visit in the training set, we form a positive sample using the user, intent probability, and POI, with the comparator output label as 1. We then form a negative sample by keeping the user and intent probability unchanged and randomly selecting a POI from other visits made by the same user, with the comparator output label as 0. Through learning with this module, we obtain user, intent, and POI embeddings, as well as the user interest generator module, which carry user intent profile attention information to POIs. These module parameters will be further utilized in the mobility prediction model.

2.3.3 Transformer-based Prediction Model

Past research has shown that the transformer architecture is effective for mobility prediction tasks (Yang et al., 2022). Inspired by this, we propose a transformer-based mobility model that effectively utilizes user intent profile information while incorporating data associations learned during the contrastive learning phase. This integration allows the model to achieve excellent predictive accuracy.

Our model's input sequence unit is defined as $(u, l, c, t, P_u(t))$, where u is the user ID, l is the current POI of the user, c is the category of the POI, t is the time of day of the next movement, and $P_u(t))$ is the user intent profile function. To fully leverage the data relationships learned during contrastive learning, we incorporate the visit interest vector generation model and the POI embedding model from the contrastive learning phase as part of our model, using the weights trained during this phase as initialization. Additionally, we generate an embedding for each c and use time2vector (Kazemi et al., 2019) to embed t. These two embeddings are then processed through a single-layer fully connected neural network to learn their interaction. Finally, we concatenate the visit interest vector, the fused embedding of c and t, and the POI embedding, and input them into a transformer encoder structure. We use a single-layer fully connected neural network as the decoder layer, which includes three output heads for POI, category, and time.

3 Experiments

3.1 Datasets

To evaluate our model, we conducted experiments on three public datasets (Feng et al., 2018; Yabe et al., 2024). For the first two representative datasets (Feng et al., 2018): one comprising mobile application location data from a popular social network vendor, referred to as the Beijing dataset, and the other consisting of call detail records (CDR) data from a major cellular network operator, referred to as the Shanghai dataset. Additionally, we also conducted tests on a widely used synthetic dataset, Yjmob100K (Yabe et al., 2024). We selected a portion of this dataset, using one month's worth of data from some users, for our experiments. When we use this dataset, the POI name is replaced by a number, and the category is left blank.The number of users, locations, and check-ins for each dataset are detailed in Table 1. We follow the license of original paper to use the data.

As mentioned earlier, we divided the data into *human labeled data*, *fine-tuning data*, and *main data*. Specifically, we manually labeled 670 daily trajectories from 135 users in the Beijing training set to form the *human labeled data*. We randomly extracted 500 users from the training sets in Beijing, using 5 daily trajectories for each user to construct the *fine-tuning data*.

Following the common practice of mobility prediction (Feng et al., 2018; Sun et al., 2020; Yang

et al., 2022), we segmented trajectories into fixed-length sessions and applied a sliding window over the dataset to make full use of the data when training the Transformer-based Mobility Prediction Model. Specifically, for a user with m check-ins, and a fixed length n, the processed trajectories will contain $(m - n + 1)n$ check-ins, where each sequence of n check-ins forms a trajectory, and consecutive trajectories overlap by $n - 1$ point. This strategy is also applied in all the baselines using deep-learning to ensure the fair comparison.

Table 1: Basic statistics of three mobility datasets.

	Duration	Users	POIs	Records
Beijing	3 months	1566	5919	744813
Shanghai	1 month	841	6955	215379
Yjmob100K	1 month	997	17704	506022

3.2 Baselines

We consider the RNN (Graves and Graves, 2012), DeepMove (Feng et al., 2018), STAN (Luo et al., 2021), LSTPM (Sun et al., 2020), GETNext (Yang et al., 2022), LLM-Mob (Wang et al., 2023a) and LLM-Zero-Shot-NL (Beneduce et al., 2024) as baselines to benchmark the performance of our model. Detailed introduction to these baselines are in appendix. Previous deep learning-based methods do not provide information about the next visit during prediction. To ensure a fair comparison, we modify the time in the input sequence from the current visit time to the next visit time, which is exactly the same as the setting of our method, to provide information about the next visit time.

3.3 Main Results

In this section, we present the performance of our method on three datasets and compare it with other baselines. We have chosen four evaluation metrics: Acc@1, Acc@5, Acc@10, and MRR@5. This allows for a comprehensive evaluation of the model's performance. The experimental results on the three datasets are shown in Table 2. Compared to the best deep learning baseline, our method achieved relative improvements of 8.36%, 10.35% and 11.48% in the Acc@1 metric for the three datasets. Compared to the LLM-based baseline, our method was only 1.70% relatively lower in the Acc@1 metric on the Beijing dataset, but it outperformed the LLM-based baseline across all other metrics. On the Shanghai and Yjmob100K datasets, the Acc@1 relative improvements reached 14.15% and 11.83%, respectively, and our method significantly reduced computational costs compared to using LLM alone. Overall, our method consistently delivers high performance across all datasets. Although it is slightly inferior to some other methods on one certain individual metric, it still outperforms any baseline when considering all metrics across the three datasets. Compared to other LLM baselines, we not only achieved superior performance but also significantly reduced inference costs.

3.4 Intent Inference Performance Analysis

We conducted experiments to compare the accuracy of different intent annotation methods on the test set relative to manual annotations. The experimental results are shown in Figure 2. To verify the effectiveness of our LLM intent annotation workflow and fine-tuning, we used data of 55 users in the *human labeled data* as test set and data of 80 users in the *human labeled data* as training set.

As show in Figure 2a, for models that have not been fine-tuned, if they are directly tasked with intent annotation without using our workflow, neither GPT-4o-mini nor the smaller Llama-3-8B-Instruct achieves an annotation accuracy above 0.4, indicating poor performance. However, after applying our annotation workflow, the accuracy of intent annotation significantly improved, with GPT-4o-mini reaching an accuracy of 0.767 and Llama-3-8B-Instruct reaching 0.596. This demonstrates the effectiveness of our workflow. Nonetheless, it is important to note that the performance of Llama-3-8B-Instruct still lags behind that of GPT-4o-mini, which justifies the subsequent use of GPT-4o-mini to generate data to supervise the fine-tuning of Llama-3-8B-Instruct.

For the fine-tuned models, we conducted four ablation experiments to fine-tune the Llama-3-8B-Instruct model:

- **GroundTruth:** This method involves directly fine-tuning with the training set of manually annotated data.
- **w/o Workflow:** We randomly selected 500 users from the Beijing dataset, with each user having 5 daily trajectories annotated using the GPT-4o-mini workflow. However, the fine-tuning process did not incorporate the workflow, and the testing uses direct trajectory annotation.
- **Without Selection:** Based on the w/o Workflow experiment, this method added workflow-generated data during fine-tuning and utilized the complete workflow for annotation during testing.

Table 2: Performance comparison in Acc@k and MRR@5 on three datasets.

	Beijing				Shanghai				Yjmob100K			
	Acc@1	**Acc@5**	**Acc@10**	**MRR**	**Acc@1**	**Acc@5**	**Acc@10**	**MRR**	**Acc@1**	**Acc@5**	**Acc@10**	**MRR**
RNN	0.2290	0.3667	0.3941	0.2846	0.2530	0.3899	0.4232	0.3084	0.0945	0.1425	0.1144	0.1532
DeepMove	0.3129	0.5202	0.5482	0.3999	0.2713	0.3893	0.4123	0.3199	0.1119	0.1833	0.1945	0.1413
STAN	0.3270	0.6532	0.7419	0.4548	0.2566	0.5411	0.6544	0.3641	0.1365	0.3263	0.4012	0.2070
LSTPM	0.4291	0.7910	0.8202	0.5826	0.4489	0.7018	0.7422	0.5518	0.2518	0.4842	0.5308	0.3455
GETNext	0.4547	0.8175	0.8596	0.6065	0.4177	0.6782	0.7363	0.5265	0.2352	0.5180	0.5873	0.3518
LLM-Mob	**0.5012**	0.8211	0.8592	0.6338	0.4340	0.7183	0.7558	0.5522	0.2510	0.5022	0.5579	0.3518
LLM-zero-shot-NL	0.4163	0.8228	0.8630	0.5853	0.4068	0.7168	0.7604	0.5347	0.2399	0.5045	0.5691	0.3442
Ours	0.4927	**0.8352**	**0.8743**	**0.6350**	**0.4954**	**0.7654**	**0.8240**	**0.6048**	**0.2807**	**0.5518**	**0.6239**	**0.3939**
vs. DL baseline	8.36%	2.17%	1.71%	4.70%	10.35%	8.14%	7.92%	9.60%	11.48%	6.53%	6.23%	11.97%
vs. LLM baseline	-1.70%	1.72%	1.31%	0.19%	14.15%	6.56%	8.36%	9.53%	11.83%	9.37%	9.63%	11.97%

- **Full Method:** This is the complete fine-tuning method, which incorporates the data selection process based on the Without Selection approach.

As shown in Figure 2b, when using GPT-generated data for fine-tuning, the inclusion of the workflow without selection results in a significantly higher accuracy of 0.776 compared to 0.553 without the workflow. This underscores the importance of the workflow during the fine-tuning phase. Additionally, the accuracy improves further to 0.793 with data selection, compared to 0.776 without it, highlighting the effectiveness of data selection in enhancing fine-tuning performance. Fine-tuning solely with GroundTruth achieves an accuracy of 0.726, which is inferior to our comprehensive model's 0.793. This demonstrates that our approach of using GPT-generated data for fine-tuning is superior to directly utilizing ground truth data for fine-tuning.

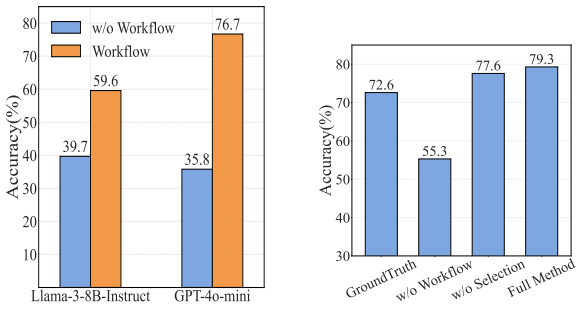


Figure 2: (a) The performance of the un-finetuned model with and without the workflow, and (b) The performance of models finetuned with different data.

3.5 Mobility Prediction Performance Analysis

To demonstrate the effectiveness of the Intent-Aware Mobility Model, we designed the following experiments using the Beijing dataset which are shown in Table 3:

- **w/o pretrain:** We removed the contrastive learning pretraining step and used random initialization for the visit interest vector generation model and the POI embedding model.
- **Train-Real:** We use the intents labeled in the training set trajectories as training data. In the test set, since the real intent of the next visit cannot be directly obtained, we use the most likely intent provided by the User Intent Profile as the corresponding intent for the test set visits.
- **w/o intent:** We removed the contrastive learning pretraining part and replaced the intent input with a uniform value to simulate the scenario without any intent information.

Table 3: Ablation experiment on mobility prediction performance using the Beijing dataset.

Experiments	**Acc@1**	**Acc@5**	**Acc@10**	**MRR@5**
LMP	0.4927	0.8352	0.8743	0.6350
w/o pretrain	0.4854	0.8299	0.8712	0.6311
Train-Real	0.4971	0.7100	0.7781	0.5810
w/o intent	0.4607	0.8289	0.8689	0.6166

The experiments demonstrate that, across various metrics, the Full Model outperforms the w/o pretrain, which in turn surpasses the w/o intent. Taking Acc@1 as an example, the w/o intent experiment, which does not include intent information, achieves only 0.4607. Incorporating intent information in the w/o pretrain experiment increases this to 0.4854, and further adding the pretraining process allows the full model to reach 0.4927. This indicates that both intent information and the contrastive learning pretraining process are effective.

The Train-Real experiment achieves the best Acc@1 score but underperforms on other metrics, indicating that using exact intent labels during training creates a train-test gap, as future intents in test data cannot be accurately known.

3.6 Efficiency Analysis

We conduct a quantitative analysis of the inference efficiency improvements resulting from using the fine-tuned *student model* (*Qwen-2.5-7B-Instruct*) to replace *teacher model* (*Llama-3.1-70B-Instruct*). We deployed models using vllm and recorded the memory consumption of deploying both models. Additionally, we tested the throughput of these tasks using 32 concurrent requests: distinguishing home and workplace in In-context Schema Learning(Task1), and intent annotation(Task2). As shown in Table 4, due to its smaller parameter size, the *student model* has a significant advantage in memory consumption, with the *teacher model* consuming approximately nine times more memory. Furthermore, due to simpler computations and output texts from the fine-tuned *student model* not containing complex reasoning, the output text length is significantly reduced. This results in a substantial increase in throughput for the *student model* compared to the *teacher model*, with ratios of 16.8 times and 9.2 times for the two tasks, respectively.

Table 4: Efficiency of *teacher model* and *student model*.

Model	Memory	Task1	Task2
Teacher Model	298496 MB	1.98/s	1.93/s
Student model	33198 MB	33.23/s	17.81/s

3.7 User Intent Profiles Analysis

We analyzed the computed User Intent Profiles, and Figure3 shows two typical examples. In the left graph, the user's "Working" curve reaches a very high value during the day, while the "At Home" curve peaks at night, with other intent curves showing slight fluctuations. This corresponds to the behavior pattern of a person with a stable job. In the right graph, the "At Home" curve consistently remains higher than the other intent curves, with only slight increases in other intents in the morning. This corresponds to the behavior pattern of people who are unemployed and tend to remain near their home address for extended periods. Overall, the phenomena exhibited by the User Intent Profiles align with our understanding.

4 Related Work

Mobility Prediction. In the past decades, various deep learning methods (Liu et al., 2016; Feng et al., 2018; Sun et al., 2020; Luo et al., 2021; Yang et al., 2022) have been proposed to predict the human mobility. While these methods succeed in

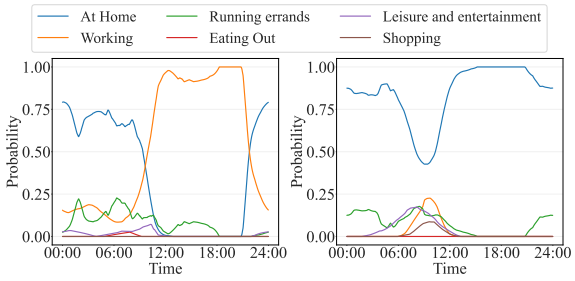


Figure 3: Two typical user intent profiles, representing people with stable jobs and people who stay at home for extended periods.

modelling the sequential patterns in the trajectory, they cannot capture the shared mobility patterns between users effectively. To solve this problem, graph neural networks (Lim et al., 2020; Yan et al., 2023; Xu et al., 2020; Wang et al., 2023b; Yin et al., 2023) are introduced into the mobility prediction modelling. However, due to the absence of large scale mobility intent dataset, these works ignore the intent modelling behind the mobility. In this work, we employ the agentic workflow to enable the large-scale automatic annotation of mobility intent and further propose an effective intent-aware mobility prediction model.

LLMs and Agents. LLMs (Achiam et al., 2023; Dubey et al., 2024) have achieved rapid development in the past few years. Recently, LLM based agent framework (Wang et al., 2024; Xi et al., 2023; Shao et al., 2024) are proposed to complement the deficiencies of LLMs on specific domain knowledge and unleash the power of LLMs in real-world tasks, e.g., ChatDev (Qian et al., 2023) for project programming and WebAgent (Gur et al., 2023) for autonomous web tasks. Researchers try to directly apply LLMs in the mobility modelling (Wang et al., 2023a; Beneduce et al., 2024) and achieve promissing results. Different from these works, we use LLMs as the mobility data annotator to augment mobility data for training a stronger small domain-specific model.

5 Conclusion

In this paper, we investigate the problem of intent enhanced mobility prediction problem. We propose LMP, an agentic workflow based framework to harness the commonsense reasoning abilities of LLMs for intention-aware mobility prediction. Extensive experiments on three real-world datasets demonstrate the effectiveness of proposed framework. In the future, we plan to extend the framework to more spatial-temporal applications.

6 Limitations

Despite the significant improvements achieved by the LMP framework in intention-aware mobility prediction, several limitations need to be acknowledged:

6.1 Dynamic Nature of Intentions

The current model presumes that human intentions follow a daily cycle, with similar intentions at the same time each day. However, this assumption is an oversimplification. Human intentions can be more complex due to factors such as holidays or ad-hoc schedule changes. Future work should consider more sophisticated models that account for these dynamic variations in human intentions.

6.2 Labeling of Home and Work Locations

Currently, home and work locations are identified through statistical analysis of time distribution and frequency, which is more applicable to individuals with stable and singular home and work locations. For those with multiple residences or workplaces, the effectiveness of the LMP method may diminish. Enhancing the model to better accommodate such variability could improve its robustness and applicability.

6.3 Intent Configuration

The current six intents—"At Home," "Running Errands," "Working," "Eating Out," "Leisure and Entertainment," and "Shopping"—exhibit imbalances in frequency distribution and predictive accuracy distribution, which somewhat affect the model's performance. Identifying more balanced and easily predictable intents presents a worthwhile challenge for consideration.

6.4 Dataset Bias

The performance of the current model has been evaluated using real-world datasets from Beijing, Shanghai and Yjmob100K. These datasets may be subject to biases in recording frequency and distribution due to the data collection methods employed. As a result, there may be discrepancies between the experimental outcomes and actual user mobility behaviors. Addressing these biases in future studies could lead to more accurate and generalizable results.

7 Ethics Statement

The development and application of the LMP framework adhere to the following ethical principles:

7.1 Respect for Privacy

We prioritize the protection of individual privacy. The data used in this study consists of aggregated and anonymized mobility data, ensuring that no personal information can be traced back to individual users.

7.2 Transparency

We strive to ensure transparency in the operation of the LMP framework. Our methodologies, data sources, and limitations are clearly documented to facilitate peer review and reproducibility. This openness not only supports academic scrutiny but also builds trust with stakeholders and the wider community.

7.3 Beneficence

The primary aim of the LMP framework is to enhance the accuracy of mobility predictions, thereby aiding urban planning and management. Moreover, it seeks to improve the interpretability of existing machine learning-based prediction methods, enabling stakeholders to make informed decisions based on the predictive insights. By promoting responsible and beneficial use of technology, we aim to contribute positively to societal needs.

7.4 Ethical Use of LMP

We recognize the potential risk that the LMP framework could be utilized to process personal trajectory data, which might lead to privacy breaches. However, our method does not explicitly include personal information beyond user trajectories, significantly minimizing the risk of privacy leaks. We are firmly opposed to any misuse that jeopardizes privacy security, and we will closely monitor the ethical application of our research methods in subsequent implementations.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Ciro Beneduce, Bruno Lepri, and Massimiliano Luca. 2024. Large language models are zero-shot next location predictors. *arXiv preprint arXiv:2405.20962*.

Qingqing Chen and Ate Poorthuis. 2021. Identifying home locations in human mobility data: an open-source r package for comparison and reproducibility. *International Journal of Geographical Information Science*, 35(7):1425–1448.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jie Feng, Yong Li, Chao Zhang, Funing Sun, Fanchao Meng, Ang Guo, and Depeng Jin. 2018. Deepmove: Predicting human mobility with attentional recurrent networks. In *Proceedings of the 2018 World Wide Web Conference*.

Alex Graves and Alex Graves. 2012. *Supervised sequence labelling*. Springer.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. A survey on llm-as-a-judge. *Preprint*, arXiv:2411.15594.

Izzeddin Gur, Hiroki Furuta, Austin Huang, Mustafa Safdari, Yutaka Matsuo, Douglas Eck, and Aleksandra Faust. 2023. A real-world webagent with planning, long context understanding, and program synthesis. *arXiv preprint arXiv:2307.12856*.

Shan Jiang, Yingxiang Yang, Siddharth Gupta, Daniele Veneziano, Shounak Athavale, and Marta C González. 2016. The timegeo modeling framework for urban mobility without travel surveys. *Proceedings of the National Academy of Sciences*, 113(37):E5370–E5378.

Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupart, and Marcus Brubaker. 2019. Time2vec: Learning a vector representation of time. *arXiv preprint arXiv:1907.05321*.

JungMi Lee and Norbert M. Seel. 2012. *Schema-Based Learning*, pages 2946–2949. Springer US, Boston, MA.

Ilaria Liccardi, Alfie Abdul-Rahman, and Min Chen. 2016. I know where you live: Inferring details of people's lives by visualizing publicly shared location data. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1–12.

Nicholas Lim, Bryan Hooi, See-Kiong Ng, Xueou Wang, Yong Liang Goh, Renrong Weng, and Jagannadan Varadarajan. 2020. STP-UDGAT: Spatial-temporal-preference user dimensional graph attention network for next poi recommendation. In *Proceedings of the 29th ACM International conference on information & knowledge management*, pages 845–854.

Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the next location: A recurrent model with spatial and temporal contexts. In *AAAI*.

Yingtao Luo, Qiang Liu, and Zhaocheng Liu. 2021. Stan: Spatio-temporal attention network for next location recommendation. In *Proceedings of the web conference 2021*, pages 2177–2185.

Chen Qian, Xin Cong, Cheng Yang, Weize Chen, Yusheng Su, Juyuan Xu, Zhiyuan Liu, and Maosong Sun. 2023. Communicative agents for software development. *arXiv preprint arXiv:2307.07924*, 6.

Chenyang Shao, Fengli Xu, Bingbing Fan, Jingtao Ding, Yuan Yuan, Meng Wang, and Yong Li. 2024. Beyond imitation: Generating human mobility from context-aware reasoning with large language models. *arXiv preprint arXiv:2402.09836*.

Ke Sun, Tieyun Qian, Tong Chen, Yile Liang, Quoc Viet Hung Nguyen, and Hongzhi Yin. 2020. Where to go next: Modeling long-and short-term user preferences for point-of-interest recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 214–221.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, and 1 others. 2024. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345.

Xinglei Wang, Meng Fang, Zichao Zeng, and Tao Cheng. 2023a. Where would i go next? large language models as human mobility predictors. *arXiv preprint arXiv:2308.15197*.

Zhaobo Wang, Yanmin Zhu, Chunyang Wang, Wenze Ma, Bo Li, and Jiadi Yu. 2023b. Adaptive graph representation learning for next poi recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 393–402.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, and 1 others. 2022a. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022b. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, and 1 others. 2023. The rise and potential of large language model based agents: A survey. *arXiv preprint arXiv:2309.07864*.

Fengli Xu, Zongyu Lin, Tong Xia, Diansheng Guo, and Yong Li. 2020. Sume: Semantic-enhanced urban mobility network embedding for user demographic inference. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(3):1–25.

Takahiro Yabe, Kota Tsubouchi, Toru Shimizu, Yoshihide Sekimoto, Kaoru Sezaki, Esteban Moro, and Alex Pentland. 2024. Yjmob100k: City-scale and longitudinal dataset of anonymized human mobility trajectories. *Scientific Data*, 11(1):397.

Xiaodong Yan, Tengwei Song, Yifeng Jiao, Jianshan He, Jiaotuan Wang, Ruopeng Li, and Wei Chu. 2023. Spatio-temporal hypergraph learning for next poi recommendation. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*, pages 403–412.

Song Yang, Jiamou Liu, and Kaiqi Zhao. 2022. Getnext: trajectory flow map enhanced transformer for next poi recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on research and development in information retrieval*, pages 1144–1153.

Feiyu Yin, Yong Liu, Zhiqi Shen, Lisi Chen, Shuo Shang, and Peng Han. 2023. Next poi recommendation with dynamic graph and explicit dependency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4827–4834.

Wei Zeng, Chi-Wing Fu, Stefan Müller Arisona, Simon Schubiger, Remo Burkhard, and Kwan-Liu Ma. 2017. Visualizing the relationship between human mobility and points of interest. *IEEE Transactions on Intelligent Transportation Systems*, 18(8):2271–2284.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19632–19642.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *ArXiv*, abs/2403.13372.

A Appendix

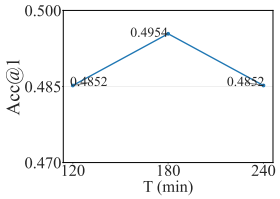
A.1 Parameter Analysis

A.1.1 Model Parameter Analysis

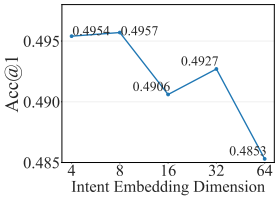
In the "Intent-Aware Mobility Model" section, we focused on examining the impact of parameter T during user intent profile generation, the impact of the Intent Embedding Dimension, the relationship between Pretrain Epochs during the contrastive learning phase and predictive performance, as well as the relationship between the trajectory length used for training and predictive performance. Specifically, Acc@1 on the Beijing dataset was used to evaluate the predictive performance for the Intent Embedding Dimension and Pretrain Epochs, while Acc@1 on the Shanghai dataset was used to evaluate the impact of parameter T and trajectory length. The experimental results are shown in Figure 4.

The model achieves optimal performance when parameter T is set to 180 minutes. When T is smaller, the model may overfit historical data, and when T is too large, the temporal influence range becomes excessive, leading to distortion in the model's representation. Acc@1 reaches its maximum when the Intent Embedding Dimension is 8, and performance declines as the Intent Embedding Dimension continues to increase. This may be due to overfitting of the intents caused by the data. There is a noticeable positive correlation between Pretrain Epochs and Acc@1, indicating that increasing the number of contrastive learning pretraining epochs within the studied parameter range helps capture more data relationships. Additionally, the overall performance of the model improves as the training trajectory length increases, suggesting that the model's training relies to some extent on the length of historical trajectories.

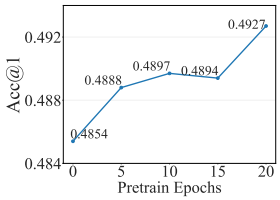
Simultaneously, we also investigated the effects of using different ranks during LoRA fine-tuning. We employed GPT4O-mini as the teacher model and Qwen2.5-7B-Instruct as the student model, using the annotation accuracy on the test set of *human labeled data* as the evaluation metric. The experimental results are shown in Table 5. The accuracy of our method across different ranks consistently falls between 0.79 and 0.8, indicating that our approach exhibits high robustness against different fine-tuning parameter settings.



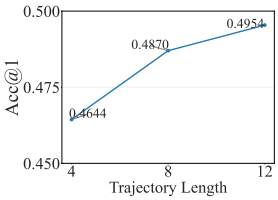
(a) Relation between parameter T and performance.



(b) Relation between intent embedding dimension and performance.



(c) Relation between pretrain epochs and performance.



(d) Relation between trajectory length and performance.

Figure 4: Parameter analysis of parameter T, intent embedding dimension, pretrain epochs, and trajectory length.

Table 5: Relationship between LoRA rank and annotation accuracy.

Rank	Accuracy
4	0.792
8	0.797
16	0.794

A.1.2 Performance on different LLMs

To investigate the generalization capability of our method across different LLMs, we tested the effectiveness of the intent annotation workflow using various LLMs. We evaluated the performance of three larger models, GPT-4o-mini, Qwen2.5-72B-Instruct, and Llama3.1-70B-Instruct, without fine-tuning. Additionally, we tested two smaller models, LLaMA-3-8B-Instruct and Qwen2.5-7B-Instruct, after fine-tuning with data generated by GPT-4o-mini. The data division for the experiments is the same as described in Section 3.4. The experimental results are shown in Table 6. The three larger models exhibit some performance differences but all achieve relatively high performance, with accuracies of 0.767, 0.709, and 0.816, respectively. The two smaller models demonstrate relatively stable accuracy, both ranging between 0.79 and 0.80. Overall, the experimental results indicate that our method is transferable across different LLMs.

Table 6: The intent annotation accuracy of different *teacher models* and the intent annotation accuracy of various finetuned *student models* when using GPT-4o-mini as the *teacher model*.

Groups	Model	Accuracy
teacher model (Annotation with agentic workflow)	GPT-4o-mini	0.767
	Qwen2.5-72B-Instruct	0.709
	Llama3.1-70B-Instruct	0.816
student model (Annotation after finetuning)	Llama3-8B-Instruct	0.793
	Qwen2.5-7B-Instruct	0.797

A.2 Baselines

- **RNN (Graves and Graves, 2012)**: A classical model for processing sequential data, capturing temporal dependencies through recurrent connections.
- **DeepMove (Feng et al., 2018)**: A model that combines recurrent networks and attention layer to capture multi-scale temporal periodicity of human mobility.
- **STAN (Luo et al., 2021)**: A model that utilizes a dual-attention structure to enhance next-location recommendation by aggregating spatio-temporal correlations and incorporating personalized item frequency (PIF).
- **LSTPM (Sun et al., 2020)**: It integrates a non-local network for capturing long-term preferences and a geo-dilated recurrent neural network for short-term preferences modelling.
- **GETNext (Yang et al., 2022)**: A state-of-the-art model that introduces a user-agnostic global trajectory flow map and a novel graph enhanced transformer to improve next POI recommendation.
- **LLM-Mob (Wang et al., 2023a)**: It introduces a method using LLMs to predict human mobility by capturing dependencies in movement data and enhancing accuracy through context-inclusive prompts.
- **LLM-Zero-Shot-NL (Beneduce et al., 2024)**: It provides a method for mobility prediction using LLMs with a zero-shot approach

A.3 Detailed Training Settings

A.3.1 Hardware Platform for Main Results and Baselines

- **CPU:** Intel Xeon Platinum 8358
- **GPU:** NVIDIA GeForce RTX 4090

A.3.2 Baselines

- **RNN and DeepMove:**

  - Embedding dimensions for location and user: 128
  - Optimizer: Adam
  - Learning rate: 1e-3
  - Maximum training epochs: 20
- **LSTPM:**
  - Embedding dimensions for location and user: 50
  - Optimizer: Adam
  - Learning rate: 1e-4
  - Maximum training epochs: 20
- **STAN:**
  - Embedding dimensions for location and user: 50
  - Optimizer: Adam
  - Learning rate: 3e-3
  - Maximum training epochs: 20
- **GETNext:**
  - Embedding dimensions for location and user: 128
  - Optimizer: Adam
  - Learning rate: 1e-3
  - Maximum training epochs: 200
- Trajectory length standardized to 12.

A.3.3 Learning from Experience Process

- **Iterations:** 5
- **Each iteration:**
  - Number of proposals: 3
  - Annotation attempts per proposal: 5

A.3.4 Fine-Tuning Settings in Main Results

- **Teacher Model:** GPT-4o-mini
- **Student Model:** Llama3-8B-Instruct
- **Fine-tuning Framework:** LLaMA-Factory (Zheng et al., 2024)
- **Fine-tuning Method:**
  - LoRA adapters
  - Rank: 8
- **Training Details:**
  - Epochs: 3
  - Batch size: 2
  - Learning rate: 5e-5

A.3.5 Intent-Aware Mobility Prediction Model in Main Results

- **Embedding Dimensions**
  - POI and user: 128
  - Time, POI category, intent: 32
- **Constructive Learning:**
  - Epochs: 20
  - Batch size: 2048
  - Learning rate: 0.001
- **Mobility Prediction Model:**
  - Trajectory: 12
  - Number of layers: 2
  - Feed-forward network dimension: 1024
  - Attention heads: 2
  - Dropout: Probability of 0.3
  - Optimizer Type: Adam
  - Learning rate: 1e-3
  - Weight decay rate: 5e-4
  - Maximum training epochs: 200

A.4 Prompt example

Prompt and answer example of In-context Schema Learning:

```
Prompt: Your task is to identify the
    user's home and work place based on
    the trajectory data.\n        The
    home place is usually have a very
    high frequency of visits, And it
    may have peak visit times in the
    evening, at night, or on early
    mornings.\n        The work place
    is usually have a relatively high
    frequency of visits, And it may
    have peak visit times in the
    daytime.\n        The trajectory
    data under analysis is as follows:
    [{'Name': 'POI1', 'Percent':
    '48.4%', 'Time Distribution':
    [('0:00', '46.7%'), ('9:00',
    '6.7%'), ('18:00', '20.0%'),
    ('19:00', '20.0%'), ('22:00',
    '6.7%')]}, {'Name': 'POI2',
    'Percent': '29.0%', 'Time
    Distribution': [('7:00', '22.2%'),
    ('8:00', '33.3%'), ('10:00',
    '22.2%'), ('15:00', '11.1%'),
    ('16:00', '11.1%')]}, {'Name':
    'POI3', 'Percent': '12.9%', 'Time
    Distribution': [('9:00', '50.0%'),
    ('14:00', '25.0%'), ('15:00',
    '25.0%')]}, {'Name': 'POI4',
    'Percent': '3.2%', 'Time
    Distribution': [('18:00',
    '100.0%')]}, {'Name': 'POI5',
    'Percent': '3.2%', 'Time
```

```
    Distribution': [('17:00',
    '100.0%')]}].\n        Each record
    represents a POI that the user has
    visited.\n        The meanings of
    each feature are as follows:\n
           - Name: POI name\n        -
    Percent: The percentage of times
    the behavior pattern occurred\n
           - Time Distribution: The
    time distribution of visits to the
    POI, in the format of (hour,
    percentage).\n        Respond using
    the following JSON format:\n
    {\"home\": \"home place\",\"work\":
    \"work place\",\"reason\": \"reason
    for prediction\"}\n

Answer: ```json\n{\n  \"home\":
    \"POI1\",\n  \"work\": \"POI2\",\n
    \"reason\": \"POI1 has the highest
    visit frequency at 48.4%, with
    significant evening and night
    visits, indicating it is likely the
    home. POI2 has a visit pattern
    concentrated during daytime hours,
    suggesting it is the
    workplace.\"\n}\n```
```

Prompt and answer example of Intent Annotation Attempt and intent annotation:

```
\n        Your task is to give intent
    prediction using trajectory data.
    Let's think step by step.\n
    The user's home and work place are
    as follows: home place is POI1 and
    work place is POI2.\n\n        The
    trajectory data under analysis is
    as follows: (POI1,Residential
    community,2019-10-13
    00:00:00),(POI1,Residential
    community,2019-10-13
    09:00:00),(POI1,Residential
    community,2019-10-13 19:16:47),.\n
           Each stay in trajectory data
    is represented as (poi, category,
    start time).\n        Here's what
    each element means:\n        - poi:
    the POI the user visited.\n
    - category: the category of the
    POI.\n        - start time: the
    time the user visit the POI.\n
           \n        Here are some
    rules you should follow:['When the
    user is at Home, the intent is At
    Home.', 'When the user is at
    Workplace, the intent is Working.',
    'When the user is at a restaurant
    during meal hours, the intent is
    Eating Out.', 'When the user is at
    a location during shopping hours,
    the intent is Shopping.', 'When the
    user is at a location during
    leisure hours, the intent is
    Leisure and entertainment.', 'When
    the user is at a location that is
    not Home or Workplace and does not
    fit any other criteria, the intent
    is Running errands.']\n        The
    rules are nessary and sufficient
    conditions, which means if a rule
    is satisfied, the intent in the
    rule should be annotated and when
    the rule is not satisfied, the
    intent should not be annotated.\n
           The order of the rules
    represents the priority of the
    proposals. The top rules have
    higher priority.\n        Intent
    you can choose:['At Home',
    'Working', 'Running errands',
    'Eating Out', 'Leisure and
    entertainment', 'Shopping']. You
    should not annotate the intent that
    is not in this list.\n        \n
           There are 3 stays in the
    trajectory data. So, the output
    should have 3 predicted intents.\n
           Consider step by step,
    finally respond using the following
    JSON format (Make sure to have one
    predicted intent for each stay in
    the trajectory data, And you have
    to assign one of the intents to
    each stay in the trajectory
    data):{\"predicted_intent\":
    [\"predicted intents\"]}

Answer: To predict the intent based on
    the given trajectory data and the
    associated rules, let's analyze
    each stay one by one.\n\n1. **First
    Stay: (POI1, Residential community,
    2019-10-13 00:00:00)**\n   - POI:
    POI1 (Home)\n   - Category:
    Residential community (not
    specified in the rules, but it is
    the home)\n   - Time: 00:00:00 \n
    - According to the rule: \"When the
    user is at Home, the intent is At
    Home.\"\n   - **Predicted Intent:
    At Home**\n\n2. **Second Stay:
    (POI1, Residential community,
    2019-10-13 09:00:00)**\n   - POI:
    POI1 (Home)\n   - Category:
    Residential community\n   - Time:
    09:00:00 \n   - Again at home.
    According to the same rule as
    above: \"When the user is at Home,
    the intent is At Home.\"\n   -
    **Predicted Intent: At Home**\n\n3.
    **Third Stay: (POI1, Residential
    community, 2019-10-13 19:16:47)**\n
      - POI: POI1 (Home)\n   -
    Category: Residential community\n
    - Time: 19:16:47 \n   - Still at
    home. According to the same rule:
    \"When the user is at Home, the
    intent is At Home.\"\n   -
    **Predicted Intent: At
    Home**\n\nFinally, compiling the
    predicted intents for each stay in
    the specified JSON
    format:\n\n```json\n{\"predicted_intent\":
    [\"At Home\", \"At Home\", \"At
    Home\"]}\n```
```

Prompt and answer example of Correction Sugges-

tion Generation:

```
Prompt: Your task is to give your
    proposals for the rules for intent
    prediction using trajectory data.
    Let's think step by step.\n
    There are 6 intents to choose from:
    ['At Home', 'Working', 'Running
    errands', 'Eating Out', 'Leisure
    and entertainment', 'Shopping'].\n
       The rules used for intent
    prediction now are as follows:
    ['When the user is at Home, the
    intent is At Home.', 'When the user
    is at Workplace, the intent is
    Working.', 'When the user is at a
    location during meal hours, the
    intent is Eating Out.', 'When the
    user is at a location during
    shopping hours, the intent is
    Shopping.', 'When the user is at a
    location during leisure hours, the
    intent is Leisure and
    entertainment.', 'When the user is
    at a location that is not Home or
    Workplace, the intent is Running
    errands.'].\n    There are several
    results of intent prediction for
    trajectory data under the rules.
    Here are what each element means:\n
       - Home: Place where the user
    lives, which is related to the
    intent 'At Home'.\n    - Workplace:
    Place where the user works, which
    is related to the intent
    'Working'.\n    - Trajectory: The
    user's trajectory data under
    analysis. Each stay in trajectory
    data is represented as (poi,
    category, start time).\n    -
    Predicted intent list: Based on the
    home, workplace and trajectory
    data, the intent list annotated by
    the rules for each stay in the
    trajectory data. Each intent is
    corresponding to a stay in the
    trajectory data.\n    - True intent
    list: The true intent list for each
    stay in the trajectory data.\n
    \n    user0: Home is POI1,
    Workplace is POI2, trajectory is
    (POI1, Educational Facilities,
    2019-11-04 00:00:00)(POI2,
    Residential Community, 2019-11-04
    08:00:00)(POI1, Educational
    Facilities, 2019-11-04 19:00:00),
    the predicted intent list under the
    rules is ['At Home', 'Working', 'At
    Home']. The true intent list is
    ['At Home', 'Working', 'At
    Home'].\n    \n    user1: Home is
    POI3, Workplace is None, trajectory
    is (POI3, Residential Community,
    2019-12-16 00:00:00)(POI4,
    Companies and Enterprises,
    2019-12-16 08:00:00)(POI3,
    Residential Community, 2019-12-16
    16:15:00), the predicted intent
    list under the rules is ['At Home',
    'Running errands', 'At Home']. The
    true intent list is ['At Home',
    'Running errands', 'At Home'].\n
    \n    user2: Home is POI5,
    Workplace is POI6, trajectory is
    (POI6, Companies and Enterprises,
    2019-11-30 00:00:00)(POI7, Other
    Banking and Finance, 2019-11-30
    14:45:00)(POI8, Residential
    Community, 2019-11-30
    17:45:00)(POI6, Companies and
    Enterprises, 2019-11-30 21:45:00),
    the predicted intent list under the
    rules is ['Working', 'Running
    errands', 'Running errands',
    'Working']. The true intent list is
    ['Working', 'Running errands',
    'Running errands', 'Working'].\n
    \n    user3: Home is POI9,
    Workplace is None, trajectory is
    (POI9, Other Food, 2019-12-05
    00:00:00), the predicted intent
    list under the rules is ['At
    Home']. The true intent list is
    ['At Home'].\n    \n    user4: Home
    is POI10, Workplace is POI11,
    trajectory is (POI10, Real Estate
    Community Facilities, 2019-12-24
    00:00:00)(POI11, Residential
    Community, 2019-12-24
    09:00:00)(POI10, Real Estate
    Community Facilities, 2019-12-24
    14:00:00)(POI11, Residential
    Community, 2019-12-24
    14:45:00)(POI10, Real Estate
    Community Facilities, 2019-12-24
    18:15:00), the predicted intent
    list under the rules is ['At Home',
    'Working', 'At Home', 'Working',
    'At Home']. The true intent list is
    ['At Home', 'Working', 'At Home',
    'Working', 'At Home'].\n    \n
    Please give your proposals based on
    the difference between predicted
    intent list and true intent list.\n
       You should consider the
    relationship between the user's
    home, workplace, trajectory and
    intent.\n    The rules should be as
    simple and loose as possible. Don't
    add unnecessary restrictions.\n
    Each intent should have only one
    rule using simple statement
    sentences, such as \"When the user
    is at Workplace, the intent is
    Working.\"\n    The rules should be
    nessary and sufficient conditions,
    which means if a rule is satisfied,
    the intent in the rule should be
    annotated and when the rule is not
    satisfied, the intent should not be
    annotated.\n    There should not
    contain specific user POI name or
    category name in the proposals.\n
       The order of the rules
    represents the priority of the
    proposals. The top rules have
    higher priority.\n    Only modify
    the rules when it is necessary. If
    not necessary, you can give the
    proposals to keep the original
```

```
    rules.\n    Let's think step by
    step. Finally answer in the json
    format, example: {{\"proposals\":
    [\"Add a rule that when the user is
    at mall, the intent is
    Shopping.\",\"Modify the rule when
    the user is at restaurant, the
    intent is Eating Out to when the
    user is at restaurant and the time
    is 12:00-13:00, the intent is
    Eating Out.\"]}}\n

Answer: After analyzing the provided
    user data and the predicted versus
    true intent lists, I propose the
    following rules for intent
    prediction using trajectory data.
    The current rules are generally
    effective, but some adjustments can
    help improve accuracy while keeping
    the rules simple and
    loose.\n\n**Proposed Rules:**\n1.
    When the user is at Home, the
    intent is At Home.\n2. When the
    user is at Workplace, the intent is
    Working.\n3. When the user is at a
    location during meal hours, the
    intent is Eating Out.\n4. When the
    user is at a location during
    shopping hours, the intent is
    Shopping.\n5. When the user is at a
    location during leisure hours, the
    intent is Leisure and
    entertainment.\n6. When the user is
    at a location that is not Home or
    Workplace, the intent is Running
    errands.\n\n**Suggestions for
    modification:**\n- **Modify Rule
    3**: To ensure clarity on meal
    hours, I propose to specifically
    define meal hours. For instance,
    the rule can state: \"When the user
    is at a restaurant during
    established meal hours (e.g.,
    7:00-9:00, 12:00-14:00,
    18:00-20:00), the intent is Eating
    Out.\" This clarity can help in
    correctly capturing the intent
    related to meal times.\n  \n-
    **Modify Rule 4**: Similar to the
    Eating Out rule, we may specify
    shopping hours. For example, \"When
    the user is at a shopping location
    during established shopping hours
    (e.g., 10:00-21:00), the intent is
    Shopping.\" This will help clarify
    the intention behind being in a
    shopping-related
    location.\n\nOverall, I suggest
    maintaining most of the original
    rules while adding specificity to
    the time-related rules.
    \n\nHeres the response in JSON
    format:\n\n```json\n{\n
    \"proposals\": [\n    \"Modify the
    rule when the user is at a
    restaurant to specify that the
    intent is Eating Out during
    established meal hours (e.g.,
    7:00-9:00, 12:00-14:00,
    18:00-20:00).\",\n    \"Modify the
    rule when the user is at a shopping
    location to specify that the intent
    is Shopping during established
    shopping hours (e.g.,
    10:00-21:00).\"\n  ]\n}\n```
```

Prompt and answer example of Insights Correction:
:

```
Prompt:  Your task is to update the
    rules for intent prediction using
    trajectory data.\n    In the
    prediction task, the user's home,
    work place and trajectory data( in
    form of (poi, category, start
    time)) are given. The task is to
    annotate the user's intent of each
    stay in the trajectory data.\n
    There are 6 intents to choose from:
    ['At Home', 'Working', 'Running
    errands', 'Eating Out', 'Leisure
    and entertainment', 'Shopping'].\n
       The proposals are generated
    based on the difference between
    predicted intent list and true
    intent list. \n    The rules used
    for intent prediction now are as
    follows: ['When the user is at
    Home, the intent is At Home.',
    'When the user is at Workplace, the
    intent is Working.', 'When the user
    is at a location during meal hours,
    the intent is Eating Out.', 'When
    the user is at a location during
    shopping hours, the intent is
    Shopping.', 'When the user is at a
    location during leisure hours, the
    intent is Leisure and
    entertainment.', 'When the user is
    at a location that is not Home or
    Workplace, the intent is Running
    errands.'].\n    There are several
    groups of proposals for the rules:
    {\"propolsal group 0\": [\"Modify
    the rule when the user is at a
    location during meal hours, the
    intent is Eating Out to include a
    specific time range for Eating Out,
    e.g., when the time is 11:00-14:00
    or 17:00-20:00.\", \"Add a fallback
    rule that when the user is at a
    location that is not Home or
    Workplace and does not fit any
    other criteria, the intent is
    Running errands.\"], \"propolsal
    group 1\": [\"Modify the rule when
    the user is at a restaurant to
    specify that the intent is Eating
    Out during established meal hours
    (e.g., 7:00-9:00, 12:00-14:00,
    18:00-20:00).\", \"Modify the rule
    when the user is at a shopping
    location to specify that the intent
    is Shopping during established
    shopping hours (e.g.,
    10:00-21:00).\"], \"propolsal group
    2\": [\"Keep the existing rule:
    When the user is at Home, the
```

```
    intent is At Home.\", \"Keep the
    existing rule: When the user is at
    Workplace, the intent is
    Working.\", \"Modify the rule for
    Eating Out to: When the user is at
    a location during meal hours, the
    intent is Eating Out, or at a
    restaurant if the time is within
    typical meal times.\", \"Modify the
    rule for Leisure and entertainment
    to: When the user is at a location
    during leisure hours, the intent is
    Leisure and entertainment, or at
    locations that typically host
    leisure activities.\", \"Add a
    rule: When the user is at a
    location that is neither Home nor
    Workplace, the intent is Running
    errands.\"]}.\n    Please update
    the rules based on the proposals.
    Please summarize the common points
    of the proposal groups to update
    the rules.\n    There should not
    contain specific user POI name or
    category name in the rules.\n
    All the rules should be directly
    about the relationship between
    home, workplace, trajectory and
    intent.\n    The rules should be as
    simple and loose as possible. Don't
    add unnecessary restrictions.\n
    Each intent should have only one
    rule using simple statement
    sentences, such as \"When the user
    is at Workplace, the intent is
    Working.\"\n    The rules should be
    nessary and sufficient conditions,
    which means if a rule is satisfied,
    the intent in the rule should be
    annotated and when the rule is not
    satisfied, the intent should not be
    annotated.\n    The order of the
    rules represents the priority of
    the proposals. The top rules have
    higher priority.\n    Let's think
    step by step. Finally answer in the
    json format,
    example:{\"updated_rules\": [\"When
    the user is at Workplace, the
    intent is Working.\",\"When the
    user is at restaurant, the intent
    is Eating Out.\"]}\

Answer: To update the rules for intent
    prediction using trajectory data,
    we need to consider the common
    elements from the proposal groups
    while ensuring that our rules
    remain simple, clear, and maintain
    the necessary and sufficient
    conditions. \n\nHeres how we can
    summarize the proposals:\n\n1.
    **Meal Hours**: There is a
    consensus on specifying that the
    intent 'Eating Out' should apply
    during specific meal hours. This
    applies to general meal times and
    also extends to when the user is at
    a restaurant.\n\n2. **Shopping
    Hours**: The proposals suggest
    defining the intent 'Shopping' to
    apply specifically during
    established shopping hours.\n\n3.
    **Running Errands**: Theres
    agreement on using 'Running
    errands' when the user is at a
    location that is neither Home nor
    Workplace and doesnt meet other
    criteria.\n\n4. **Leisure and
    Entertainment**: The proposals
    suggest that leisure activities can
    encompass both time frames and
    specific types of
    locations.\n\nConsidering these
    points, we can reformulate the
    rules as follows:\n\n1. **'At
    Home'** - When the user is at Home,
    the intent is At Home.\n2.
    **'Working'** - When the user is at
    Workplace, the intent is
    Working.\n3. **'Eating Out'** -
    When the user is at a location
    during meal hours, or at a
    restaurant if within typical meal
    times, the intent is Eating
    Out.\n4. **'Shopping'** - When the
    user is at a location during
    shopping hours, the intent is
    Shopping.\n5. **'Leisure and
    Entertainment'** - When the user is
    at a location during leisure hours,
    or at locations that typically host
    leisure activities, the intent is
    Leisure and entertainment.\n6.
    **'Running Errands'** - When the
    user is at a location that is not
    Home or Workplace and does not fit
    any other criteria, the intent is
    Running errands.\n\nNow,
    consolidating these into JSON
    format, we have:\n\n```json\n{\n
    \"updated_rules\": [\n    \"When
    the user is at Home, the intent is
    At Home.\",\n    \"When the user is
    at Workplace, the intent is
    Working.\",\n    \"When the user is
    at a location during meal hours, or
    at a restaurant if within typical
    meal times, the intent is Eating
    Out.\",\n    \"When the user is at
    a location during shopping hours,
    the intent is Shopping.\",\n
    \"When the user is at a location
    during leisure hours, or at
    locations that typically host
    leisure activities, the intent is
    Leisure and entertainment.\",\n
    \"When the user is at a location
    that is not Home or Workplace and
    does not fit any other criteria,
    the intent is Running errands.\"\n
    ]\n}\n```
```

Prompt and answer example of data selection:

```
Prompt: Your task is to determine
    whether the results of the
    trajectory intent annotation match
    the rules.\n        The trajectory
```

```
    is: (POI1, Residential Community,
    2019-10-04 15:15:00)\n        Each
    trajectory has several stays, and
    each stay is represented by a tuple
    of (POI name, POI category, visit
    time).\n        The user's home
    place is defined as POI2.\n
    The user's work place is defined as
    POI1.\n        The predicted intent
    is: ['Running errands']\n
    Each intent is corresponding to a
    stay in the trajectory.\n
    The rules are: ['When the user is
    at Home, the intent is At Home.',
    'When the user is at Workplace, the
    intent is Working.', 'When the user
    is at a restaurant during meal
    hours, the intent is Eating Out.',
    'When the user is at a location
    during shopping hours, the intent
    is Shopping.', 'When the user is at
    a location during leisure hours,
    the intent is Leisure and
    entertainment.', 'When the user is
    at a location that is not Home or
    Workplace and does not fit any
    other criteria, the intent is
    Running errands.']\n        The
    rules are nessary and sufficient
    conditions, which means if a rule
    is satisfied, the intent in the
    rule should be annotated and when
    the rule is not satisfied, the
    intent should not be annotated.\n
           The order of the rules
    represents the priority of the
    proposals. The top rules have
    higher priority.\n        Analyze
    it step by step in the order of the
    rules, Finally answer whether the
    intent is annotated correctly in
    the format of <yes> or <no>.\n

Answer: "To analyze whether the
    predicted intent of 'Running
    errands' is annotated correctly, we
    need to evaluate the provided
    trajectory against the specified
    rules step-by-step.\n\n1. **Rule 1:
    When the user is at Home, the
    intent is At Home.**\n   - The
    user's home place is defined
    asPOI1. The trajectory indicates
    the user is at POI2, which is not
    their home. Therefore, this rule
    does not apply.\n\n2. **Rule 2:
    When the user is at Workplace, the
    intent is Working.**\n   - The
    user's workplace is defined as
    POI2. The trajectory indicates the
    user is at this location.
    Therefore, according to this rule,
    the intent should be annotated as
    'Working', but the predicted intent
    is 'Running errands'. This
    indicates a mismatch.\n\n3. **Rule
    3: When the user is at a restaurant
    during meal hours, the intent is
    Eating Out.**\n   - The location
    (POI2) does not indicate that it is
    a restaurant, so this rule does not
    apply.\n\n4. **Rule 4: When the
    user is at a location during
    shopping hours, the intent is
    Shopping.**\n   - The location
    (POI2) also does not indicate a
    shopping scenario, so this rule
    does not apply.\n\n5. **Rule 5:
    When the user is at a location
    during leisure hours, the intent is
    Leisure and entertainment.**\n   -
    The location does not specifically
    indicate a leisure context; hence,
    this rule does not apply.\n\n6.
    **Rule 6: When the user is at a
    location that is not Home or
    Workplace and does not fit any
    other criteria, the intent is
    Running errands.**\n   - Since the
    user is at their workplace (POI2),
    this rule does not apply.\n\nAfter
    analyzing all the rules, we find
    that the user is at their
    workplace, and according to Rule 2,
    the intent should be 'Working'.
    However, the predicted intent is
    'Running errands', which does not
    align with the prioritized
    rules.\n\nTherefore, the intent is
    **not annotated correctly**.
    \n\nThe final answer is: <no>"
```