Technical Report

1 HYPERPARAMETER STUDY

Since the trends on MTR-2 Weeks, MTR-3 Weeks, and MTR-4 Weeks datasets are similar to MTR-1 Week, we show results on MTR-1 Week dataset. Specifically, we explore two hyperparameters:

Minimal Group Size g_m. We evaluate with different minimal group sizes, including 3, 4, 5, and 6. Figure 1 (a) shows that the performance improves with an increase in g_m up to a certain point (i.e., 5), after which it achieves the best results. The reason is that a larger group size results in patterns that have more passengers moving together, which is easier to be detected.

Dimensionality d of Hidden State. We investigate the effect of varying the dimensionality of the hidden state, choosing values from 256, 512, 1024, and 2048. Figure 1 (b) shows that beyond a certain point (i.e., 512), increasing d leads to degradation due to the overfitting problem.

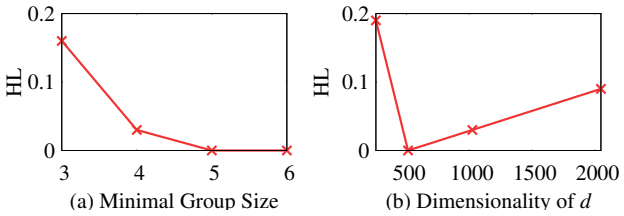

Fig. 1. Hyperparameter study of our model.

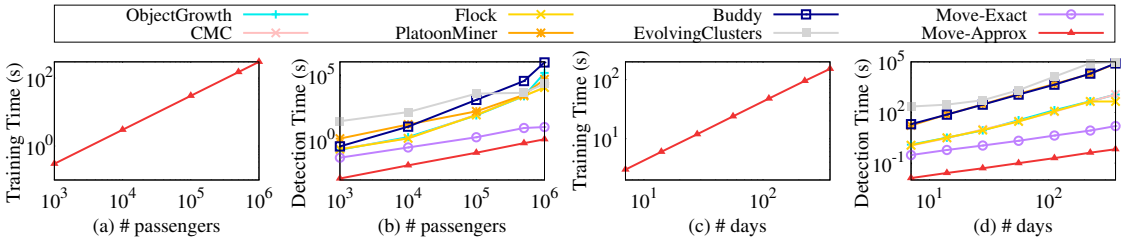

Fig. 2. Scalability on (a-b) the number of passengers; (c-d) the number of days.

Author's address:

Manuscript submitted to ACM

2 SCALABILITY EVALUATION

We use GeForce GTX 1080 Ti 11 GB GPU for evaluation. Note that we report the average running times over five runs for all methods.

2.1 Scalability w.r.t. the number of objects

Figure 2 (a, b) depicts the scalability of our model and other competitors concerning the number of passengers, demonstrating a linear relationship between running time and passenger size.

2.2 Scalability w.r.t. the number of time intervals

Figure 2 (c, d) illustrates the scalability of our model and other competitors concerning the span of the time dimension when enlarging it from 7 days to 365 days by replicating the data multiple times. The trend shows linear scalability, indicating the model's efficiency in handling varying time intervals. Note that we fix the size of passengers as 10,000 here and only change the span of the time dimension.

3 PROOFS

3.1 Lemma 1

Proof. We first prove that all pairs in the ground truth will be included in the result of Move-Exact. Each pair of riders will be put together by Move-Exact, because the pair will be grouped in the same bin in the coarse bin-based mapping stage and finally put in the same group during the finer intra-bin refinement.

Besides, any pair of riders not included in the ground truth will not be put in the same group, as it will be eliminated by different stations during the mapping stage or by different time stamps during the refinement stage. □

3.2 Theorem 1

Proof. We analyze the two main stages of `Move-Exact`: the coarse bin-based mapping and the finer intra-bin refinement.

In the coarse bin-based mapping stage, each object o_i with k_1 trajectory records is scanned once to obtain its complete trajectory tr_i. This requires $O(k_1 \cdot |O|)$ time over all objects. Then, for each object, all possible combinations of n trips are selected, resulting in $k_2 = \binom{|tr_i|}{n}$ subsets per object. Generating and labeling these subsets (i.e., computing $l_b = f_1(tr_n)$) requires $O(k_2 \cdot |O|)$ time in total, since each subset is processed once.

In the finer intra-bin refinement stage, each object within a bin is examined once to compute temporal proximity via $f_2(m_{l_b})$. This step involves simple time-interval comparisons, which are linear in the number of objects, i.e., $O(|O|)$.

Combining both stages, the overall time complexity is $O((k_1 + k_2) \cdot |O|)$. Given that $|tr_i|$ and n are typically small in travel data, k_2 remains much smaller than $|O|$, making the algorithm efficient for large-scale datasets. □

3.3 Theorem 2

Proof. For each object o_i, the self-attention encoder computes pairwise interactions among all timestamps in its trajectory, requiring $O(|p_i|^2)$ time. Since $|p_i| \leq |p|_m$ for all objects, the total computation over the dataset is

$$\sum_{i=1}^{|O|} O(|p_i|^2) \leq O(|O| \cdot |p|_m^2).$$

The multi-head mechanism introduces a constant multiplicative factor proportional to the number of heads H, and subsequent MLP and sigmoid operations are linear in $|p_i|$, which do not affect the asymptotic complexity. □

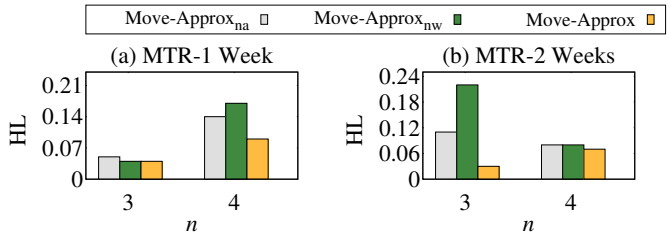

Fig. 3. Ablation study evaluating the contribution of key components in `Move-Approx`.

4 ABLATION STUDY

Since the trends of other settings are similar to MTR-1 and MTR-2 Weeks when $n = 3, 4$, we only show the latter settings in Figure 3. Move-Approx$_{\mathbf{na}}$ (**n**o **a**ttention) excludes the attention. We exclude weights of the positive samples in the group class from the loss function as Move-Approx$_{\mathbf{nw}}$ (**n**o **w**eight). We observe that Move-Approx outperforms Move-Approx$_{\mathbf{na}}$ on all datasets, with a more significant gap on the MTR-2 Weeks dataset. This underscores the importance of considering spatial and temporal dimensions differently to enhance performance. Moreover, Move-Approx outperforms Move-Approx$_{\mathbf{nw}}$, especially on the MTR-2 Weeks dataset, emphasizing the importance of including weights of positive samples in the loss function.