Towards Robust Travel Time Estimation: An Out-of-Distribution Generalization Approach

Anonymous Author(s)

Abstract

Travel is becoming increasingly convenient with the development of the Internet. Travel Time Estimation (TTE) serves as a fundamental task for online traffic services. However, it faces a pressing challenge due to the volatile nature of traffic: the out-of-distribution (OOD) problem. In this paper, we investigate the OOD generalization problem, with a specific focus on the TTE task. We analyze the underlying generative process of traffic data by constructing a relational Structural Causal Model (SCM). We reveal that the complex causal relationships can be simplified through a technique we term selective blocking. Based on this simplification, we propose a two-step deconfounding procedure to eliminate the spurious effects of the environment on the invariant representation. Specifically, we design an invariant learning model, **OOD** model for **T**ravel **T**ime **E**stimation (OOD4TTE). First, our model infers potential environments for data generation through an environment inference module. Then, we implement the two-step deconfounding procedure using a contrastive learning module and an invariant gating network. Finally, we generate robust travel time estimates based on the invariant representations. We demonstrate the superior generalization capability of OOD4TTE through comprehensive experiments on two real-world datasets from different cities.

CCS Concepts

• **Information systems** → **Information systems applications**.

Keywords

Travel Time Estimation, OOD Generalization, Traffic Data Management

ACM Reference Format:

Anonymous Author(s). 2018. Towards Robust Travel Time Estimation: An Out-of-Distribution Generalization Approach. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 11 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Travel Time Estimation (TTE), as a core service of location-based services (LBS), refers to the task of predicting how long it will take to travel from a given origin to a destination along a specific route, using available traffic context such as departure time, weather, etc. For instance, in Fig. 1, traveling on the red route on a rainy Saturday at 4:13 p.m. is estimated to cost 33 minutes. The TTE task plays a vital role in various applications, including trip planning [12], transportation scheduling [31], and delivery services [25].

Although several studies have addressed this topic [13, 14, 21, 32, 35], the TTE task remains vulnerable to out-of-distribution (OOD) issues. These problems stem from distributional shifts in covariates due to different data generation environment between training and testing phases, whereas shortcut features undermine generalization by failing to transfer to unseen testing environments. As shown in Fig. 2, improving OOD generalization in the TTE task is particularly challenging because of the varying covariates and changing environments.

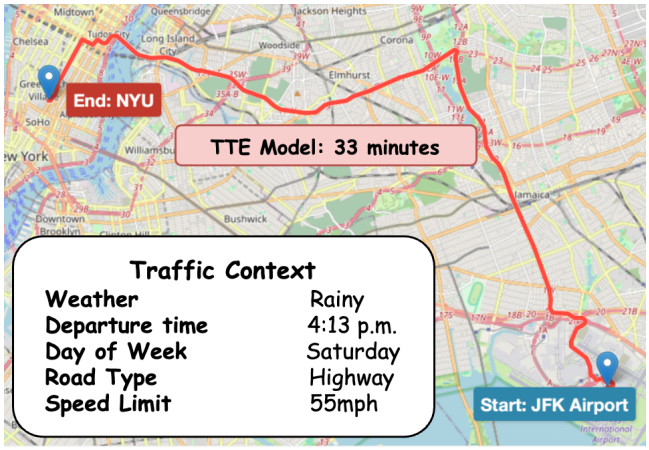

Figure 1: An example of the TTE: The travel time from JFK Airport to New York University is estimated to be 33 minutes.

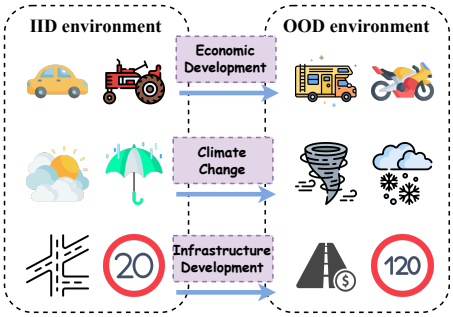

Figure 2: Illustration of the OOD problem in the TTE task: Training on IID, Testing on OOD

Invariant learning [2, 4, 15, 36] is a widely adopted approach for OOD generalization. It assumes underlying environments for

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

data generation and aims to extract invariant representations from input features to support downstream tasks. The invariant features should remain stable with respect to the task targets across different environments. Therefore, regardless of the distribution shift, the invariant model can still generalize well to the test data.

In this paper, we adopt the invariant learning framework for OOD generalization in the TTE task. We propose an **OOD** model for **T**ravel **T**ime **E**stimation (OOD4TTE). However, there are two challenges: i) the environment labels for data generation are unobservable; ii) dependencies of causal variables are complex, which increases the difficulty of invariant learning. To solve the challenges, we aim to construct a structural causal model (SCM) specific to the TTE task. With a careful analysis of the causality among variables, we propose a two-step deconfounding procedure. We propose a selective blocking technique to make the invariant learning easier and enhance the generalization performance. Moreover, the potential environments are inferred using our environmental inference module. Experimental results demonstrate that OOD4TTE achieves strong OOD generalization performance.

Our literature review reveals that no existing studies have addressed the OOD problem in the TTE task, despite its urgency and practical importance. In a nutshell, the main contributions of this paper are as follows:

- **New Problem.** To our best knowledge, we are the first to discuss the OOD problem in the TTE task. We use the causal theory and invariant learning to address the challenges.
- **Novel Method.** We propose an end-to-end model, OOD4TTE, to address the OOD generalization problem in the TTE task. To tackle the two key challenges, we introduce an *environment inference module* and a *two-step deconfounding procedure* for learning invariant representations. By leveraging *selective blocking*, we simplify the underlying causal structure, thereby reducing the difficulty of invariant learning and improving generalization performance. Based on this simplified structure, the *invariant gating network* effectively captures stable features across different environments.
- **Extensive Experiments.** We conduct extensive experiments to evaluate the performance of our method. OOD4TTE is evaluated on two real-world datasets. Our experimental results demonstrate that OOD4TTE achieves superior generalization performance under OOD scenarios.

2 Preliminary

2.1 Definition & Problem Formulation

Generally, the queries for travel time estimation are composed of two types of covariates, categorized by their underlying semantics.

Definition 1 (Route). *The route T is of sequential characteristics with location information, which usually displays a multi-step time series $\{s_1, s_2, \ldots, s_m\}$, such as road names, longitudes and latitudes.*

Definition 2 (Traffic context). *The traffic context C consists of traffic attributes, such as the current time, weather, infrastructure information, etc.*

Our core problem, Travel Time Estimation (TTE), is formally defined as follows:

Problem 1 (Travel Time Estimation). *Given a query $\langle T, C\rangle$, where C represents the traffic context and T denotes the planned route, the objective is to accurately estimate the travel time Y.*

We define the OOD generation problem in TTE as follows:

Problem 2 (OOD Generalization in TTE). *Given the environment support $\mathcal{E}$ representing data generation conditions, we aim to find a function $\hat{f}(\cdot)$ that satisfies:*

$$\hat{f} = \arg\min_{f} \left\{ \sup_{e \in \mathcal{E}} \mathbb{E}_{P(T,C,Y|E=e)} [\mathcal{L}(f(T,C), Y)] \right\}, \tag{1}$$

where $\mathcal{L}$ denotes the loss function. The objective is to optimize performance in the *worst-case* environment, thereby enhancing generalization to previously unseen environments.

3 A Causal Perspective for OOD

We aim to address the two problems using causal theory. Nevertheless, the provable identification of invariant features requires the imposition of certain general constraints.

Assumption 1 (Identifiability of Environment). *The environment can be inferred from the traffic context C and routes T.*

It is reasonable to make this assumption in traffic scenarios. The traffic contexts and the routes contain rich information that reflects the data-generating distribution. For example, bad weather typically leads to longer travel times, while route patterns often reflect dynamic changes in travel demand or trip necessity.

Assumption 2 (Sufficient Condition). *The sufficient invariants for accurate and generalizable travel time estimation can be identified from traffic contexts C and routes T.*

This assumption aligns with those made in many OOD generalization studies [15, 39]. Considering the success of incorporating traffic context and route as covariates in practical online travel time estimation [8, 16, 34], we believe that they contain sufficient invariant information. Then, the invariance principle for invariant identification is defined as:

Corollary 1 (TTE Invariance Principle). *For the random variables: route T, traffic context C, and travel time Y, two distinct representation functions $\Phi(\cdot)$ and $\Psi(\cdot)$ capture invariants from C and T respectively, which satisify:*

$$P^{e_i}(Y \mid C = \Phi(\mathbf{c}), T = \Psi(\mathbf{t})) = P^{e_j}(Y \mid C = \Phi(\mathbf{c}), T = \Psi(\mathbf{t})), \tag{2}$$

We aim to learn invariant representations for traffic contexts and routes simultaneously. This allows the model to avoid learning shortcut features and to generalize robustly from the training data when it properly captures the invariants. However, it remains a challenge to design an appropriate representation method. Next, we leverage causal theory to address this problem.

3.1 A Causal Perspective of TTE

To rigorously analysis the problem in a causal perspective, we design a Structural Causal Model (SCM) [23] for analyzing complex interactions in the travel time estimation. Specifically, in Fig. 3, each node represents a causal variable, and each edge denotes a causal relation between two variables. The SCM illustrates the

causal relations among five key variables: environment E, route T, traffic context C, invariant representation R, and travel time Y. The interactions between these variables are described as follows:

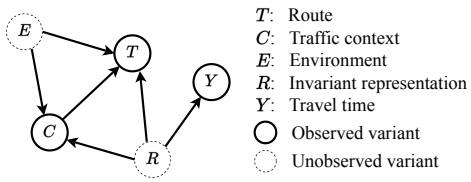

Figure 3: SCM for TTE.

- $C \leftarrow E \rightarrow T$: The environment E is characterized by factors such as socio-economic conditions and infrastructure quality, which affect the underlying generative distribution of traffic data but are difficult to fully observe. The observed C and T contain information from the unobserved confounder E, which creates shortcuts in the estimation.
- $T \leftarrow R \rightarrow C$: In addition to the confounder, invariant information is also contained in C and T. The key variable R is what we aim to infer. Thus we can make an accurate estimation with the observed C and T. For example, the speed limit could be an invariant of the traffic context, and the expressway segments could be an invariant of the routes.
- $R \rightarrow Y$: Since the invariant representation has a stable relationship with travel time, R has sufficient information for estimation.
- $C \rightarrow T$: The route is usually influenced by the traffic context. For example, people may choose to take a detour to avoid congestion during peak hours.

From this SCM, we observe that the paths $E \rightarrow C \leftarrow R$ and $E \rightarrow T \leftarrow R$ form collider structures, which induce spurious associations between E and R. For example, E = *rainy season* causes C = *rainy* and T = *expressway* (because small roads are blocked), which leads to a decrease in expected travel time. A spurious association may be learned: C = *rainy* $\rightarrow$ $Y \downarrow$. Thus, we need to adjust for C and T (i.e., by learning ivariant representation $\Phi(C)$ and $\Psi(T)$) to block the spurious association between E and R.

Intuitively, removing the path $C \rightarrow T$ can simplify the causal relationships. Although directly removing the path is feasible, it inevitably leads to the loss of essential information. Hence, we propose **selective blocking**, which partially blocks $C \rightarrow T$ to eliminate confounding influence from E. Selective blocking offers two benefits: (1) Simplifying the causal relationships by reducing the number of paths from E to R from four to two, thereby weakening confounding; (2) Enhancing the learnability of double invariants: misidentifying invariants in C (or T) increases difficulty in identifying invariants in T (C). Therefore, we propose a **two-step deconfounding procedure**: first by selectively blocking $C \rightarrow T$, then blocking $E \rightarrow C$ and $E \rightarrow T$. Based on this analysis, we now present our method.

4 Methodology

In this section, we introduce the architecture of our proposed **OOD** model for **T**ravel **T**ime **E**stimation (OOD4TTE), as shown in Fig. 4.

Given routes with various numbers of steps, we first use a temporal method (e.g., LSTM [7] and Transformer [29]) to represent them into vectors $\mathbf{T} \in \mathbb{R}^{N \times d_{\text{model}}}$. Together with the traffic contexts $\mathbf{C} \in \mathbb{R}^{N \times d_{\text{model}}}$, OOD4TTE infers the environment label for invariant learning. Next, we implement the two-step deconfounding procedure to adjust for $\mathbf{C}$ and $\mathbf{T}$, yielding $\Phi(\mathbf{C})$ and $\Psi(\mathbf{T})$, which block spurious associations between the environment labels and the invariant representations. Finally, the estimation head predicts the travel time vector $\hat{\mathbf{y}} \in \mathbb{R}^N$ from $\Phi(\mathbf{C})$ and $\Psi(\mathbf{T})$.

4.1 Environmental Inference Module

The invariant learning aims to identify features with environment-invariant relations to outcomes. However, in traffic data, environment labels are unobservable. The goal of this module is to infer the environment labels from the observable features for invariant identification. We employ soft environmental labels to infer the underlying data generating environments.

We define $\mathbf{E} \in \mathbb{R}^{N \times V}$ as the soft labels for V environments. The risk of e-th environment is defined as the weighted loss of each instance in the e-th environment, as follows:

$$R^e_{env}(f, \mathbf{E}) = \frac{1}{\sum_{i'} \mathbf{E}_{i',e}} \sum_i \mathbf{E}_{i,e}\, \mathcal{L}(f(\mathbf{C}_i, \mathbf{T}_i), y_i), \tag{3}$$

where f is an invariant model, and $\mathcal{L}$ is a loss function. Given that the invariant model learns stable relationships and achieves consistent performance across environments, the soft labels $\mathbf{E}$ should define a proper environment partition that minimizes the risk gap among environments. Here we use the variance of each environmental risk to encourage equal risks [11, 36]. The invariant model and the environment labels are updated iteratively as follows:

$$\mathbf{E}^* = \arg\min_{\mathbf{E}} \operatorname{Var} \mathcal{R}^e_{env}(f, \mathbf{E}), \tag{4}$$

$$f^* = \arg\min_{f} \mathcal{R}_{inv}(f, \mathbf{E}), \tag{5}$$

where $\mathcal{R}_{inv}$ is the risk of the invariant model. Next, we will discuss f and $\mathcal{R}_{inv}$.

4.2 Two-step Deconfounding Procedure

We perform deconfounding of the influence of environment labels on invariant representations in two steps: (1) selectively blocking the confounding path of traffic contexts on routes, where Φ_1 and Ψ_1 are the mediate representing functions for C and T, respectively; (2) based on the updated causal relations, we block the remaining spurious relations from the environment to invariant representations, denoted as $\Phi(C)$ and $\Psi(T)$.

4.2.1 Selective Blocking. In Fig. 3, path $C \rightarrow T$ increases the complexity of causality due to introducing more paths from E to R, and the model finds it much harder to learn invariants due to misinterpretation arising from the misidentification of T and C. To achieve the goal of selective blocking, it is necessary to ensure that the $\Phi_1(C)$ and $\Psi_1(T)$ are independent conditioned on the environment E (i.e., $\Phi_1(C) \perp\!\!\!\perp \Psi_1(T) \mid E$). To this end, we design a contrastive module as follows.

Initially, we shuffle the traffic contexts in data to generate negative samples. With slight abuse of notation, we define the shuffled

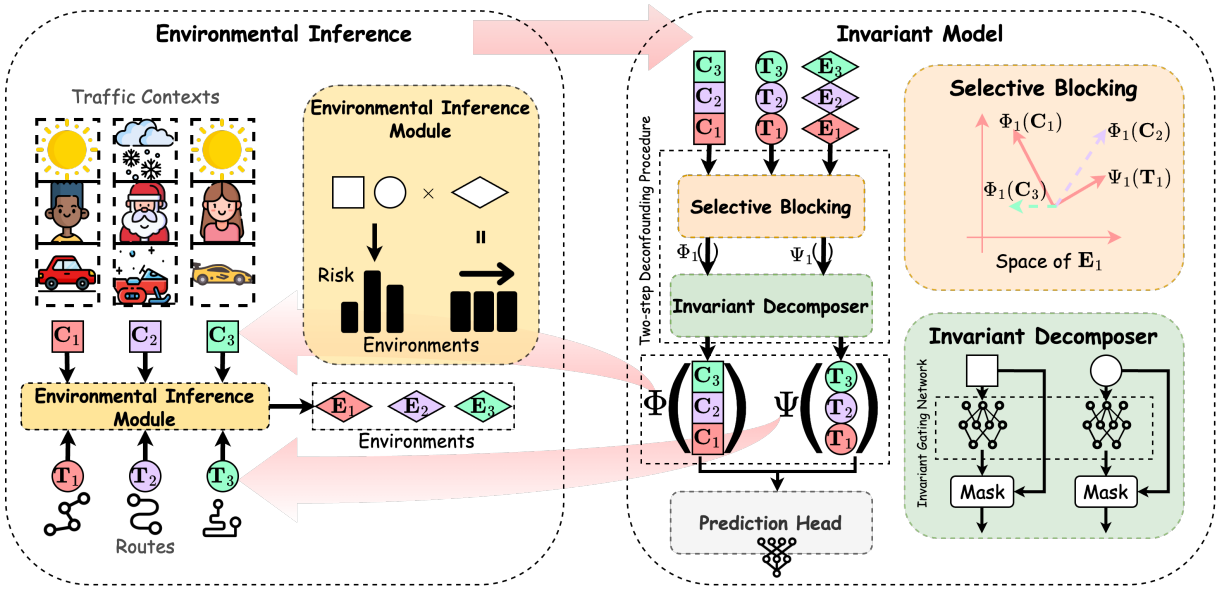

Figure 4: Architecture of OOD4TTE.

$\mathbf{C}'$ to distinguish it from the original $\mathbf{C}$, where $\mathbf{C}$ is regarded as positive samples and $\mathbf{C}'$ as negative ones.

Without loss of generalization, we adopt partial distance measure [28, 42] to quantify the conditional dependency between $\Phi_1(C)$ and $\Psi_1(T)$. The partial distance measure is a scalar quantity that quantifies dependence with the conditional correlation in Gaussian settings. In non-Gaussian situation, a partial distance of zero does not imply conditional independence. However, values closer to zero suggest a weaker association [28].

We first define the *double centered pair-wise distance* of the inferred environments $\mathbf{E}$, denoted as $\pi(\mathbf{E}) \in \mathbb{R}^{N\times N}$, as follows :

$$[\pi(\mathbf{E})]_{i,j} = \|\mathbf{E}_i - \mathbf{E}_j\|_2 - \frac{1}{N}\sum_{k=1}^{N}\|\mathbf{E}_k - \mathbf{E}_j\|_2 - \frac{1}{N}\sum_{l=1}^{N}\|\mathbf{E}_i - \mathbf{E}_l\|_2 + \frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\|\mathbf{E}_k - \mathbf{E}_l\|_2, \tag{6}$$

where $\|\cdot\|_2$ is the l_2 norm. The $\pi(\mathbf{C})$ and $\pi(\mathbf{T})$ can be similarly obtained. The *inner product operator* $\otimes$ for $\pi(\mathbf{C})$ and $\pi(\mathbf{T})$ is defined as:

$$\pi(\mathbf{C}) \otimes \pi(\mathbf{T}) := \frac{\sum_{i\neq j}[\pi(\mathbf{C})]_{i,j}\cdot[\pi(\mathbf{T})]_{i,j}}{N\times(N-3)}. \tag{7}$$

Others are similarly calculated. The *orthogonal projection operator* $\mathcal{P}$ of $\pi(\mathbf{C})$ on $\pi(\mathbf{E})$ is defined as:

$$\mathcal{P}_{\mathbf{E}}(\mathbf{C}) = \pi(\mathbf{C}) - \frac{\pi(\mathbf{C})\otimes\pi(\mathbf{E})}{\pi(\mathbf{E})\otimes\pi(\mathbf{E})}\pi(\mathbf{E}). \tag{8}$$

The projection $\mathcal{P}_{\mathbf{E}}(\mathbf{T})$ can be similarly obtained. The **partial distance measure** of $\mathbf{C}$ and $\mathbf{T}$ to $\mathbf{E}$ is defined as:

$$D_{\mathbf{E}}(\mathbf{C},\mathbf{T}) = \frac{|\,\mathcal{P}_{\mathbf{E}}(\mathbf{C})\otimes\mathcal{P}_{\mathbf{E}}(\mathbf{T})\,|}{\|\mathcal{P}_{\mathbf{E}}(\mathbf{C})\|\cdot\|\mathcal{P}_{\mathbf{E}}(\mathbf{T})\|}, \tag{9}$$

where $|\cdot|$ denotes the absolute value, $\|\mathcal{P}_{\mathbf{E}}(\mathbf{C})\| = (\mathcal{P}_{\mathbf{E}}(\mathbf{C})\otimes\mathcal{P}_{\mathbf{E}}(\mathbf{C}))^{1/2}$, and $\|\mathcal{P}_{\mathbf{E}}(\mathbf{T})\|$ is similarly defined.

To achieve the goal of $\Phi_1(C) \perp\!\!\!\perp \Psi_1(T) \mid E$, we hope the $D_{\mathbf{E}}(\Phi_1(\mathbf{C}), \Psi_1(\mathbf{T})) \ll D_{\mathbf{E}}(\Phi_1(\mathbf{C}'), \Psi_1(\mathbf{T}))$. That means, when projected onto the space of environments, the positive representations $\Phi_1(\mathbf{C})$ are orthogonal to $\Psi_1(\mathbf{T})$, while the negative representations $\Phi_1(\mathbf{C}')$ are not. Accordingly, we define the contrastive loss $\mathcal{L}_{con}$ as follows:

$$\mathcal{L}_{con} = \text{SoftPlus}(D_{\mathbf{E}}(\Phi_1(\mathbf{C}), \Psi_1(\mathbf{T})) - D_{\mathbf{E}}(\Phi_1(\mathbf{C}'), \Psi_1(\mathbf{T}))). \tag{10}$$

Both Φ_1 and Ψ_1 are two-layer MLPs. This contrastive design ensures that the learned representations $\Phi_1(\mathbf{C})$ and $\Psi_1(\mathbf{T})$ become conditionally independent given the environment, thereby selectively blocking the confounding influence through $C \rightarrow T$ from E.

4.2.2 Invariant Decomposer. Selective blocking initially mitigates spurious associations from $C \rightarrow T$. However, the environment E still has spurious associations with the invariant representation R. Hence, we further process the issue.

To further blocking spurious relations through C, we design the **invariant gating network** for the invariant feature extraction. Selectively blocked representations $\Phi_1(\mathbf{C})$ are shuffled, denoted as $\Phi_1'(\mathbf{C})$, and then, the gating weight $\mathbf{G}$ is generated as follows:

$$\mathbf{G} = \sigma\left(topk\left(\mathbf{W}_C\Phi_1(\mathbf{C}) \mid \text{Var}\left[\mathbf{W}_C\Phi_1(\mathbf{C}), \mathbf{W}_C\Phi_1'(\mathbf{C})\right]\right)\right), \tag{11}$$

where *topk* selects the smallest $k\%$ of elements according to the given condition, while setting all remaining entries to zero. σ is an activation function. $\mathbf{W}_C$ is a learnable parameter for linear projection. The gating weight is applied to $\Phi_1(\mathbf{C})$, and the invariant representation function Φ is defined as:

$$\Phi(\mathbf{C}) = \mathbf{G}\odot\Phi_1(\mathbf{C}). \tag{12}$$

Similarly, routes $\mathbf{T}$ can be represented by Ψ with a different linear projection weight $\mathbf{W}_T$.

Table 1: Performance in the traffic context OOD scenarios of the Shenzhen dataset. The first row indicates the testing period. The star indicates statistical significance with $p < 0.05$. The bold font means the best results, and the underlined font indicates the second best.

Method	0:00-5:59			6:00-11:59			12:00-17:59			18:00-23:59		
	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓
WDR												
ERM	119.40	16.41	176.56	153.18	17.48	229.53	105.71	14.97	145.57	132.54	18.96	215.84
+ RevIN	110.19	15.73	163.68	138.73	16.42	208.87	101.50	15.00	149.16	114.42	17.19	184.31
+ FOIL	134.47	24.88	174.65	131.61	18.92	189.37	143.88	27.60	178.21	158.95	32.44	201.50
+ HRM	196.49	24.53	272.29	250.75	27.65	342.85	149.35	19.79	212.39	131.97	18.69	213.88
WDDRA												
ERM	113.77	15.95	166.47	149.86	17.23	222.12	104.30	14.99	145.05	115.09	17.70	171.24
+ RevIN	113.02	15.97	166.85	131.60	15.73	198.69	104.94	14.98	144.51	113.71	17.38	177.09
+ FOIL	118.51	19.29	163.87	132.96	15.82	206.07	112.71	16.18	164.17	209.85	27.59	302.34
+ HRM	112.06	16.05	164.38	132.54	15.89	200.15	101.14	**14.82**	147.55	112.13	17.24	171.40
ProbTTE												
ERM	119.80	16.64	193.15	147.73	16.21	240.95	108.01	15.35	145.21	119.24	17.94	192.63
+ RevIN	111.43	15.66	164.42	137.73	16.23	209.54	101.13	15.13	149.19	112.16	16.93	175.17
+ FOIL	118.20	17.48	169.76	148.76	19.17	216.72	167.21	27.20	219.08	117.00	17.52	174.10
+ HRM	118.43	16.29	175.59	133.25	15.93	201.38	103.77	14.96	145.89	115.43	17.33	188.08
Informer												
ERM	161.71	22.25	241.27	226.58	25.23	332.76	153.51	21.77	222.18	182.46	25.33	277.21
+ RevIN	150.26	19.42	242.93	174.04	20.49	267.68	143.06	20.85	208.63	153.03	22.66	240.86
+ FOIL	252.99	45.47	366.16	229.35	31.08	337.97	207.12	36.31	285.12	223.74	44.37	307.97
+ HRM	182.28	23.70	286.28	305.58	32.30	427.61	182.88	24.33	269.59	182.98	27.93	277.71
OOD4TTE	**109.30***	**15.28***	**163.62***	**119.79***	**14.75***	**183.30***	**99.10***	14.95	**144.35**	**108.33***	**16.88**	**166.11***

4.3 Estimation Head

Under the sufficient condition, we use the invariant representations $\Phi(\mathbf{C})$ and $\Psi(\mathbf{T})$ as inputs to the estimation head. The estimation head is implemented as a three-layer MLP, with the sum of the invariant representations as the input. The estimation is as follows:

$$\hat{\mathbf{y}} = \text{MLP}(\Phi(\mathbf{C}) + \Psi(\mathbf{T})). \tag{13}$$

4.4 Model Optimization

The quality of estimated travel time $\hat{\mathbf{y}}$ is evaluated using Mean Absolute Percentage Error (MAPE). Additionally, to encourage the model to learn invariant representations and enhance generalization, we introduce the variance of the loss. Note that the variance term used here serves the purpose of invariant representation learning by optimizing the invariant model f, whereas the one in $\mathcal{R}_{env}$ is used for environment inference by optimizing $\mathbf{E}$. The risk of invariant learning $\mathcal{R}_{inv}$ is defined as:

$$\mathcal{R}_{inv} = \underbrace{\mathbb{E}_E\left[\mathcal{L}_{tte}\right]}_{first\ moment} + \alpha \underbrace{\mathbb{E}_E\left[\text{Var}\mathcal{L}_{tte}\right]}_{second\ central\ moment} + \beta\mathbb{E}\left[\mathcal{L}_{con}\right], \tag{14}$$

where α and β are tunable hyperparameters that tradeoff the weights between different risk terms. Eq. (4) and (5) are optimized iteratively. During training, we adopt Adam optimizer [10] for the invariant model and SGD for the environmental inference module.

In the invariance principle (Eq. 2), we aim to align the loss distributions across different environments, i.e., $\forall e_i, e_j \in \mathcal{E}$, we have $P_{e_i}(\mathcal{L}_{tte}) \approx P_{e_j}(\mathcal{L}_{tte})$. The three terms in our invariant risk can then be interpreted as the first moment, the second central moment, and a regularization term. Minimizing the risk can thus be seen as performing moment matching, which approximates the loss distributions. In this way, our method adheres to the invariance principle and can capture the invariance.

5 Experiments

In this section, we report our experimental results. The implemented code and data are available at our repository [1].

5.1 Experimental Setting

5.1.1 Datasets. The datasets used in the experiments are the **Shenzhen Dataset**[2] and the **Porto Dataset**[3]. The **Shenzhen Dataset** contains 30-day routes from August 1st, 2020 to August 30th, 2020

[1] Link to be provided upon publication
[2] https://sigspatial2021.sigspatial.org/sigspatial-cup/
[3] http://www.geolink.pt/ecmlpkdd2015-challenge/dataset.html

Table 2: Performance in the traffic context OOD scenarios of the Porto dataset. The first row indicates the testing period. The star indicates statistical significance with $p < 0.05$. The bold font means the best results, and the underlined font indicates the second best.

Method	0:00-5:59			6:00-11:59			12:00-17:59			18:00-23:59		
	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓
WDR												
ERM	196.39	23.03	383.93	137.44	24.26	239.71	166.18	21.21	406.88	195.50	21.84	479.32
+ RevIN	211.45	25.19	378.59	137.74	24.79	239.97	177.99	23.46	411.91	203.79	23.46	481.20
+ FOIL	208.49	23.94	394.18	143.13	25.19	268.02	181.12	23.74	480.04	231.16	24.12	470.36
+ HRM	207.55	23.98	375.34	136.28	24.08	242.03	172.63	21.33	441.26	211.63	22.95	468.27
WDDRA												
ERM	195.53	22.92	360.63	140.27	24.99	241.22	168.35	21.36	385.63	194.56	21.74	436.11
+ RevIN	206.90	24.27	407.14	138.10	24.32	251.18	178.65	23.43	399.21	206.85	23.85	441.75
+ FOIL	329.72	77.52	429.75	438.26	107.95	510.20	321.76	78.32	475.21	291.18	58.66	560.69
+ HRM	201.97	23.71	382.57	145.48	24.14	236.79	168.00	**21.14***	382.66	189.51	21.64	452.39
ProbTTE												
ERM	207.72	24.19	383.92	166.69	26.87	273.45	186.65	23.92	382.09	195.79	22.05	461.09
+ RevIN	203.17	24.96	362.15	145.19	24.08	259.82	212.31	26.26	456.62	220.16	26.96	435.11
+ FOIL	266.29	50.49	424.93	304.21	83.70	390.98	277.73	50.16	485.49	254.50	43.98	475.57
+ HRM	197.78	23.43	364.30	144.84	24.44	237.85	173.34	22.16	391.96	191.78	21.89	438.36
Informer												
ERM	258.81	29.65	434.79	258.81	29.65	434.79	229.57	29.11	426.41	240.16	27.83	476.37
+ RevIN	196.85	23.24	364.91	140.27	24.99	241.22	170.49	22.47	402.97	200.27	22.54	463.98
+ FOIL	325.93	70.76	453.37	410.11	118.90	476.03	304.21	83.70	390.98	316.03	34.38	509.83
+ HRM	200.25	23.80	365.59	139.24	24.43	246.83	177.11	23.17	442.97	199.19	22.51	455.91
OOD4TTE	**189.71***	**22.58***	**358.91***	**132.24***	**23.43***	**235.84***	**164.29**	21.84	**377.34***	**187.48***	**21.44**	**432.49***

in Shenzhen, China, collected from the Didi Platform. We selected the first 5 days of data, which includes 636,373 routes and their corresponding traffic contexts. The traffic contexts include features such as weather, departure time, and driver ID, etc. The **Porto Dataset** contains 1,710,671 routes and corresponding traffic contexts from 442 taxis in Porto, Portugal, spanning from July 1st, 2013 to June 30th, 2014. Since the routes consist of GPS positions, we mapped them onto road segments using the Valhalla[4] engine. The traffic contexts here include features such as road width, speed limit, departure time, and driver ID, etc. Both datasets are split into three folds, with ratios of 75%, 5%, and 20% for the training, validation, and testing sets, respectively.

5.1.2 Evaluation Metrics. Three metrics are used in this paper: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE). Their definitions are as follows:

$$MAE(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{N}\sum_{i=1}^{N} | \hat{\mathbf{y}}_i - \mathbf{y}_i |, \tag{15}$$

$$MAPE(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{N}\sum_{i=1}^{N} \frac{| \hat{\mathbf{y}}_i - \mathbf{y}_i |}{\mathbf{y}_i}, \tag{16}$$

$$RMSE(\hat{\mathbf{y}}, \mathbf{y}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (\hat{\mathbf{y}}_i - \mathbf{y}_i)^2}. \tag{17}$$

All these metrics are better when their values are lower. They evaluate different aspects of performance: MAE and MAPE are sensitive to small values (i.e., short routes in TTE), while RMSE is sensitive to large values (i.e., long routes in TTE).

5.1.3 Baselines. We conduct experiments on two real-world datasets from different cities, Shenzhen and Porto. The baselines include several TTE methods, namely **WDR** [35], **WDDRA** [6], **ProbTTE** [16], and **Informer** [40]. As there are no OOD methods specifically designed for the TTE task, we further include several general OOD approaches that do not require environment labels, including **RevIN** [9], **FOIL** [17], and **HRM** [18]. These OOD methods are built upon the aforementioned TTE models as backbones. **ERM** (Empirical Risk Minimization) denotes the standard training setting where no OOD method is applied.

[4]https://github.com/valhalla/valhalla

Table 3: Ablation study in the traffic context OOD scenario of the Shenzhen dataset. The first row indicates the testing period.

Method	0:00-5:59			6:00-11:59			12:00-17:59			18:00-23:59		
	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓
OOD4TTE	**109.30**	**15.28**	**163.62**	**119.79**	**14.75**	**183.30**	**99.10**	**14.95**	**144.35**	**108.33**	**16.88**	**166.11**
$\alpha = 0$	113.11	15.62	169.81	143.00	16.64	217.45	100.18	15.18	145.53	109.84	17.72	168.92
$\beta = 0$	110.77	15.33	167.54	124.59	15.98	189.78	103.28	15.27	149.96	109.75	17.21	168.17

Table 4: Ablation study in the traffic context OOD scenario of the Porto dataset. The first row indicates the testing period.

Method	0:00-5:59			6:00-11:59			12:00-17:59			18:00-23:59		
	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓
OOD4TTE	**189.71**	**22.58**	**358.91**	**132.24**	**23.43**	**235.84**	**164.29**	**21.84**	**377.34**	**187.48**	**21.44**	**432.49**
$\alpha = 0$	194.66	23.43	360.96	136.53	24.18	239.13	177.41	22.87	431.73	188.02	21.78	435.66
$\beta = 0$	194.31	23.30	362.01	135.19	23.99	237.77	172.07	22.49	412.84	189.70	21.82	436.10

5.1.4 Experimental Environment. OOD4TTE is implemented with PyTorch. The model is trained on a server equipped with three pieces of Nvidia(R) A40 GPU and one piece of Intel(R) Xeon(R) Platinum 8268 CPU.

5.2 Expermental Results

To better reveal the effectiveness of our design, we conclude with three questions and try to answer them with experimental results.

5.2.1 ***Q1: What is the performance on Travel Time Estimation in OOD scenarios?*** The OOD performance is evaluated under two scenarios: variable traffic contexts and variable routes. For variable traffic contexts, we divide the departure times into four folds, select three as training data, and use the remaining one as testing data. For variable routes, we divide the routes into 2 folds based on their lengths to simulate out-of-distribution conditions.

Table 1 displays the results of the Shenzhen dataset in the traffic context OOD scenario, while Table 2 presents those of the Porto dataset. Specifically, OOD4TTE demonstrates a significant improvement of 8.97% in worst-case outcomes on the Shenzhen dataset during the 6:00–11:59 time window. Consequently, OOD4TTE effectively delivers robust results. Results of the route OOD scenario are in the Appendix.

5.2.2 ***Q2: Are the components for optimization effective?*** We conduct an ablation study to verify the effectiveness of our method. We separately remove the corresponding loss in OOD4TTE: (1) $w/o\ \mathbb{E}_E\left[\mathrm{Var}\mathcal{L}_{tte}\right]$ ($\alpha = 0$): checking the effectiveness of the invariant model f; and (2) $w/o\ \mathcal{L}_{con}$ ($\beta = 0$): checking the effectiveness of selective blocking.

The results in the traffic context scenario have been included in Tables. 3 and 4. The results of the route OOD scenario are in the Appendix. The studied components enhance performance on both datasets. Consequently, we regard the two components as effective.

5.2.3 ***Q3: How sensitive is the model to the hyperparameters?*** We study the sensitivity of four hyperparameters: the coefficients α and β, the number of environments V and the invariant ratio k. As shown in Fig. 5, we observe that:

(1) **Cofficienet α.** The hyperparameter varies over {0.1, 0.5, 1, 5}. The results indicate that it is sensitive to the OOD, as we find it is much more sensitive between 6:00 and 11:59 on the Shenzhen dataset, which we regard as the worst-case scenario in Q1. This is because the OOD obviously decreases the generalization performance, and the variance across environments is large. Therefore, the heavier the OOD scenario is, the more sensitive the hyperparameter is.
(2) **Cofficienet β.** The hyperparameter varies from {1e-1, 5e-1, 1, 5}. The results also indicate that the hyperparameter is sensitive to the OOD. However, a larger value would not significantly degrade the performance in heavy OOD scenarios.
(3) **Number of environment V.** The V is adjusted in {4, 8, 16, 32}. A few environments reduce the representative ability, whereas a greater number of basic environments fail to extract common factors of environments.
(4) **Invariant Ratio k.** The invariant ratio k controls the ratio of shortcuts to invariant features in embeddings. The values of k are tuned in {0.2, 0.4, 0.6, 0.8}. A lower value could lose invariant information, whereas a larger value could introduce shortcuts. The results suggest a value around 0.5.

6 Related work

6.1 Travel Time Estimation

Some previous work split entire routes into sub-links (e.g., road segments) and make predictions for each sub-link [1, 32, 34]. The summing of sub-links is estimated as travel time. Obviously, prediction errors usually accumulate with these methods. WDR [35] is one of the leading solutions for such TTE tasks. It takes routes as a whole, avoiding accumulated errors. Besides these, multi-task learning is also popular for estimating travel time. DeepTTE [31] and DuETA [8] use segment-level travel time estimation as an auxiliary learning objective. MURAT [13] represents the topological structure of road networks, locations, and timestamps and considers trip distance and the number of road segments as auxiliary learning objectives. WDDRA [6] uses traffic conditions as auxiliary learning

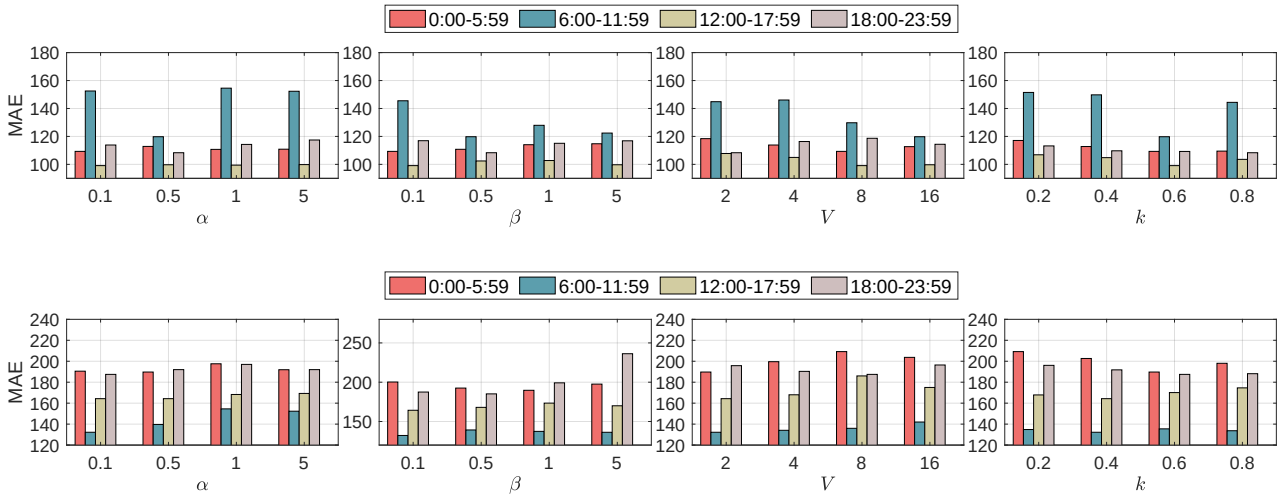

Figure 5: Sensitivity analysis of four hyperparameters in the temporal OOD scenarios. The first row is on the Shenzhen dataset. The second row is on the Porto dataset.

objectives. The probabilistic framework ProbTTE [16] studies the travel time uncertainty induced by various dynamic contextual factors. Thanks to route-wise distributions collected on a large scale by Didi, it claims a good performance on TTE tasks. DOT [14] learns the correlations between the origination and destination pair as well as the given timestamp, constructing the pixelated route (PiT) via a diffusion-based inference process. Kindly note that many methods [24, 27, 33] are spatiotemporal and thus are not directly comparable to our work due to their reliance on additional spatial information. Moreover, origin-destination TTE tasks [13, 14] are not included in our experiments, since route data are typically unavailable for such tasks.

6.2 OOD Generalization

The accessibility of environments mainly distinguishes OOD generalization methods. IRM [2] aims to learn representations that are invariant across multiple training environments. GroupDRO [26] employs a regularizer to minimize the worst-group risk, thereby improving performance across diverse environments. REx [11] promotes risk equalization across training domains to enhance generalization to unseen distributions. However, these methods typically require environment labels during training, which can be challenging to obtain in traffic scenarios. From a debiasing perspective, LfF [22] labels biased data as either bias-aligned or bias-conflicting, operating under the assumption that spurious correlations are usually much easier to learn than the expected relations. EIIL [4] proposes to infer environments based on the 'Environment Invariance Constraint principle'. ZIN [15] uncovers that it is impossible to infer environments without prior knowledge and provides guidance on selecting suitable prior knowledge for environmental inference. DDG [38] is devoted to disentangling environment information from label information. CauSTG [41] conducts deep research on spatio-temporal data in OOD scenarios. CauSTG segments time into intervals, treating them as environments based on spatial correlation situations, and subsequently discovers invariant relations across different environments. AdaRNN [5] dynamically adjusts representations across time steps to better capture evolving dependencies, while RevIN [9] employs a simple yet effective normalization strategy. FOIL [17] introduces a novel label-free environmental inference method. Heterogeneous Risk Minimization [18] proposes a novel optimization framework designed to simultaneously learn the latent heterogeneity and invariant relationships in the data, thereby achieving stable predictive performance even when the data distribution shifts. DIVERSIFY [19, 20] and ITSR [19] focus on segment-level classification tasks, and environment labels are required. Some approaches [3, 30, 37, 41] focus on spatio-temporal data. However, the data of our task is not spatio-temporal. Therefore, we do not compare with these methods.

7 Conclusion

In this paper, we first study the OOD generalization problem in the travel time estimation (TTE) task. We use the Structural Causal Model to reveal the complex causal relationships in the TTE task. Our model, OOD4TTE uses an environmental inference module to infer the missed environment labels in the TTE task and a two-step deconfounding procedure for better invariant learning. Experimental results show that OOD4TTE can effectively solve the OOD generalization problem.

A key limitation of OOD4TTE is that it is designed to learn invariant representations specific to a single city, which prevents direct transfer to a new city with different traffic dynamics and road network structures. This restricts its generalizability in multi-city settings. To address this, future work will explore transfer learning and cross-city learning strategies, such as domain adaptation or meta-learning, to enable the model to leverage knowledge from one city while maintaining robust performance in others.

References

[1] Pouria Amirian, Anahid Basiri, and Jeremy Morley. 2016. Predictive analytics for enhancing travel time estimation in navigation apps of Apple, Google, and Microsoft. In *Proceedings of the 9th ACM SIGSPATIAL International Workshop on Computational Transportation Science*. 31–36.

[2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893* (2019).

[3] Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, MA Kaili, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. 2022. Learning causally invariant representations for out-of-distribution generalization on graphs. *Advances in Neural Information Processing Systems* 35 (2022), 22131–22148.

[4] Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. 2021. Environment inference for invariant learning. In *International Conference on Machine Learning*. PMLR, 2189–2200.

[5] Yuntao Du, Jindong Wang, Wenjie Feng, Sinno Pan, Tao Qin, Renjun Xu, and Chongjun Wang. 2021. Adarnn: Adaptive learning and forecasting of time series. In *Proceedings of the 30th ACM international conference on information & knowledge management*. 402–411.

[6] Yunchong Gan, Haoyu Zhang, and Mingjie Wang. 2021. Travel Time Estimation Based on Neural Network with Auxiliary Loss. In *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*. 642–645.

[7] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[8] Jizhou Huang, Zhengjie Huang, Xiaomin Fang, Shikun Feng, Xuyi Chen, Jiaxiang Liu, Haitao Yuan, and Haifeng Wang. 2022. Dueta: Traffic congestion propagation pattern modeling via efficient graph learning for eta prediction at baidu maps. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3172–3181.

[9] Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. 2021. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International conference on learning representations*.

[10] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[11] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. 2021. Out-of-distribution generalization via risk extrapolation (rex). In *International conference on machine learning*. PMLR, 5815–5826.

[12] Yaguang Li, Dingxiong Deng, Ugur Demiryurek, Cyrus Shahabi, and Siva Ravada. 2015. Towards fast and accurate solutions to vehicle routing in a large-scale and dynamic environment. In *Advances in Spatial and Temporal Databases: 14th International Symposium, SSTD 2015, Hong Kong, China, August 26-28, 2015. Proceedings 14*. Springer, 119–136.

[13] Yaguang Li, Kun Fu, Zheng Wang, Cyrus Shahabi, Jieping Ye, and Yan Liu. 2018. Multi-task representation learning for travel time estimation. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1695–1704.

[14] Yan Lin, Huaiyu Wan, Jilin Hu, Shengnan Guo, Bin Yang, Youfang Lin, and Christian S Jensen. 2024. Origin-destination travel time oracle for map-based services. *Proceedings of the ACM on Management of Data* 1, 3 (2024), 1–27.

[15] Yong Lin, Shengyu Zhu, Lu Tan, and Peng Cui. 2022. ZIN: When and how to learn invariance without environment partition? *Advances in Neural Information Processing Systems* 35 (2022), 24529–24542.

[16] Hao Liu, Wenzhao Jiang, Shui Liu, and Xi Chen. 2023. Uncertainty-aware probabilistic travel time prediction for on-demand ride-hailing at didi. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4516–4526.

[17] Haoxin Liu, Harshavardhan Kamarthi, Lingkai Kong, Zhiyuan Zhao, Chao Zhang, and B Aditya Prakash. 2024. Time-Series Forecasting for Out-of-Distribution Generalization Using Invariant Learning. In *International Conference on Machine Learning*. 31312–31325.

[18] Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. 2021. Heterogeneous risk minimization. In *International Conference on Machine Learning*. PMLR, 6804–6814.

[19] Wang Lu, Jindong Wang, Xinwei Sun, Yiqiang Chen, Xiangyang Ji, Qiang Yang, and Xing Xie. 2024. Diversify: A general framework for time series out-of-distribution detection and generalization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, 6 (2024), 4534–4550.

[20] Wang Lu, Jindong Wang, Xinwei Sun, Yiqiang Chen, and Xing Xie. 2022. Out-of-distribution representation learning for time series classification. *arXiv preprint arXiv:2209.07027* (2022).

[21] Xiaowei Mao, Tianyue Cai, Wenchuang Peng, and Huaiyu Wan. 2021. Estimated time of arrival prediction via modeling the spatial-temporal interactions between links and crosses. In *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*. 658–661.

[22] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. 2020. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems* 33 (2020), 20673–20684.

[23] Judea Pearl. 2009. *Causality: Models, Reasoning and Inference* (2nd ed.). Cambridge University Press, USA.

[24] Yuecheng Rong, Juntao Yao, Jun Liu, Yifan Fang, Wei Luo, Hao Liu, Jie Ma, Zepeng Dan, Jinzhu Lin, Zhi Wu, et al. 2023. GBTTE: Graph Attention Network Based Bus Travel Time Estimation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 4794–4800.

[25] Sijie Ruan, Zi Xiong, Cheng Long, Yiheng Chen, Jie Bao, Tianfu He, Ruiyuan Li, Shengnan Wu, Zhongyuan Jiang, and Yu Zheng. 2020. Doing in one go: delivery time inference based on couriers' trajectories. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2813–2821.

[26] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. 2019. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731* (2019).

[27] Yibin Shen, Cheqing Jin, Jiaxun Hua, and Dingjiang Huang. 2020. TTPNet: A neural network for travel time prediction based on tensor decomposition and graph embedding. *IEEE Transactions on Knowledge and Data Engineering* 34, 9 (2020), 4514–4526.

[28] Gábor J Székely and Maria L Rizzo. 2014. Partial distance correlation with methods for dissimilarities. *The Annals of Statistics* 42, 6 (2014), 2382.

[29] A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).

[30] Binwu Wang, Jiaming Ma, Pengkun Wang, Xu Wang, Yudong Zhang, Zhengyang Zhou, and Yang Wang. 2024. Stone: A spatio-temporal ood learning framework kills both spatial and temporal shifts. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2948–2959.

[31] Dong Wang, Junbo Zhang, Wei Cao, Jian Li, and Yu Zheng. 2018. When will you arrive? Estimating travel time based on deep neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[32] Hongjian Wang, Xianfeng Tang, Yu-Hsuan Kuo, Daniel Kifer, and Zhenhui Li. 2019. A simple baseline for travel time estimation using large-scale trip data. *ACM Transactions on Intelligent Systems and Technology (TIST)* 10, 2 (2019), 1–22.

[33] Qiang Wang, Chen Xu, Wenqi Zhang, and Jingjing Li. 2021. GraphTTE: Travel time estimation based on attention-spatiotemporal graphs. *IEEE Signal Processing Letters* 28 (2021), 239–243.

[34] Yilun Wang, Yu Zheng, and Yexiang Xue. 2014. Travel time estimation of a path using sparse trajectories. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 25–34.

[35] Zheng Wang, Kun Fu, and Jieping Ye. 2018. Learning to estimate the travel time. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 858–866.

[36] Qitian Wu, Hengrui Zhang, Junchi Yan, and David Wipf. 2022. Handling distribution shifts on graphs: An invariance perspective. *arXiv preprint arXiv:2202.02466* (2022).

[37] Yutong Xia, Yuxuan Liang, Haomin Wen, Xu Liu, Kun Wang, Zhengyang Zhou, and Roger Zimmermann. 2023. Deciphering spatio-temporal graph forecasting: A causal lens and treatment. *Advances in Neural Information Processing Systems* 36 (2023), 37068–37088.

[38] Hanlin Zhang, Yi-Fan Zhang, Weiyang Liu, Adrian Weller, Bernhard Schölkopf, and Eric P Xing. 2022. Towards principled disentanglement for domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8024–8034.

[39] Zeyang Zhang, Xin Wang, Ziwei Zhang, Zhou Qin, Weigao Wen, Hui Xue, Haoyang Li, and Wenwu Zhu. 2024. Spectral invariant learning for dynamic graphs under distribution shifts. *Advances in Neural Information Processing Systems* 36 (2024).

[40] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 11106–11115.

[41] Zhengyang Zhou, Qihe Huang, Kuo Yang, Kun Wang, Xu Wang, Yudong Zhang, Yuxuan Liang, and Yang Wang. 2023. Maintaining the status quo: Capturing invariant relations for ood spatiotemporal learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3603–3614.

[42] Minqin Zhu, Anpeng Wu, Haoxuan Li, Ruoxuan Xiong, Bo Li, Xiaoqing Yang, Xuan Qin, Peng Zhen, Jiecheng Guo, Fei Wu, et al. 2024. Contrastive balancing representation learning for heterogeneous dose-response curves estimation. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 38. 17175–17183.

Table 5: Ablation study in the route OOD scenario of the Shenzhen dataset. The first row indicates the range of testing route length.

Method	0% ~ 50%			50% ~ 100%		
	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓
OOD4TTE	**110.23**	**24.93**	**150.53**	**287.50**	**17.84**	**491.40**
$\alpha = 0$	110.36	25.33	149.37	329.08	20.12	547.02
$\beta = 0$	123.63	30.42	160.50	337.73	20.54	563.90

Table 6: Ablation study in the traffic context OOD scenario of the Porto dataset. The first row indicates the range of testing route length.

Method	0% ~ 50%			50% ~ 100%		
	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓
OOD4TTE	**112.09**	**43.48**	**196.70**	**285.98**	**18.83**	**746.57**
$\alpha = 0$	112.89	46.70	209.66	289.98	19.46	755.80
$\beta = 0$	149.93	63.66	226.40	292.45	19.98	764.00

A Experimental Reproducibility

A.1 Data Preparation

Since the routes in the Porto Dataset consist of GPS positions, we first used the Valhalla engine to map the GPS positions onto the road network, obtaining additional road information such as road width and speed limit. This was done to align it with the Shenzhen dataset, which is composed of anonymous segment IDs. Valhalla ensures consistent results due to its reliance on OpenStreetMap data.

A.2 Hyperparameter Settings

The OOD4TTE model includes 4 hyperparameters: two coefficients (α and β) in the risk function, the invariant ratio k, and the number of environments V. The coefficients α and β are searched over {1e-1, 5e-1, 1, 5}. k is searched over {0.2, 0.4, 0.6, 0.8}, and V is searched over {4, 8, 16, 32}. Since the combinations of the four are very large (128 combinations), we adopt a group grid search strategy to find the suboptimal hyperparameters. Specifically, parameters in one group are grid searched first, and then the search is conducted sequentially across groups. In our model, α and β are grouped together, and k and V form the other group. All the reported results are the average of five runs.

The learning rate is set to 0.0005 with the batch size of 256 and the other parameters are displayed in Table. 7.

B Additional experiments

Tables 5 and 6 present the results of the ablation study on the Porto Dataset. The introduced components improve the model's generalization performance.

The OOD generalization performance in the route OOD scenario is shown in Table 8. We observe that the MAPE value on the shorter-half route subset is significantly higher than that of Informer + RevIN, despite our method achieving lower MAE. This is because MAPE is highly sensitive to small ground truth values, which are common in shorter routes. As a result, although our model yields smaller absolute errors, the relative errors measured by MAPE appear much larger.

Table 7: Parameter settings in our experiments.

Parameter	Shenzhen						Porto					
	0:00-5:59	6:00-11:59	12:00-17:59	18:00-23:59	0%∼50%	50%∼100%	0:00-5:59	6:00-11:59	12:00-17:59	18:00-23:59	0%∼50%	50%∼100%
α	0.1	0.5	0.1	0.5	0.1	0.5	0.5	0.1	0.1	0.1	0.1	0.1
β	0.1	0.5	0.1	0.5	5	5	1	0.1	0.1	0.1	0.1	0.1
V	8	16	2	2	8	8	2	2	2	8	8	16
k	0.6	0.6	0.6	0.8	0.8	0.8	0.6	0.4	0.4	0.6	0.4	0.8

Table 8: Performance in the route OOD scenarios on the Shenzhen and Porto datasets. The first row indicates the range of testing route length.

Method	Shenzhen 0%∼50%			Shenzhen 50%∼100%			Porto 0%∼50%			Porto 50%∼100%		
	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓	MAE ↓	MAPE ↓	RMSE ↓
WDR												
ERM	116.81	25.04	154.46	345.04	22.06	516.74	128.68	48.93	208.95	353.45	24.11	814.48
+ RevIN	114.66	25.71	156.38	299.81	18.97	499.73	124.45	31.55	209.66	333.93	21.96	807.83
+ FOIL	126.14	26.95	159.72	357.65	22.31	529.86	255.78	85.77	316.86	356.64	36.89	781.60
+ HRM	152.03	29.91	202.70	353.48	22.17	551.82	120.86	41.22	208.97	362.63	23.99	837.49
WDDRA												
ERM	118.43	25.73	159.40	346.77	22.22	518.59	121.60	52.91	206.68	414.78	29.22	871.08
+ RevIN	116.68	25.74	157.92	345.42	22.19	510.53	126.25	29.42	213.82	379.89	25.74	790.76
+ FOIL	121.86	25.32	153.19	330.99	21.80	503.17	348.83	133.17	441.85	311.03	25.28	780.26
+ HRM	114.15	25.01	152.88	345.86	22.03	509.05	117.91	38.99	204.77	341.01	21.96	778.47
ProbTTE												
ERM	118.09	26.39	174.27	295.62	18.94	496.98	118.29	39.62	201.27	513.22	31.44	1034.85
+ RevIN	158.30	32.79	218.08	294.74	18.27	493.70	130.83	26.09	225.06	504.38	31.00	2059.83
+ FOIL	174.22	33.79	239.61	292.32	18.69	**489.17**	211.14	74.42	311.00	331.00	28.13	842.55
+ HRM	116.90	26.99	168.70	293.94	18.45	494.15	122.92	27.90	209.00	320.98	26.61	830.26
Informer												
ERM	234.16	48.82	313.76	610.55	44.36	751.03	225.02	72.55	311.10	635.29	53.67	929.61
+ RevIN	261.68	43.19	347.64	380.98	24.68	583.55	131.66	**25.47**	221.66	661.14	56.78	973.15
+ FOIL	263.38	62.24	337.06	500.26	32.91	692.86	291.83	80.67	356.53	525.96	40.39	825.24
+ HRM	332.14	68.49	392.55	628.58	43.97	797.14	224.63	61.84	320.56	618.90	51.08	896.00
OOD4TTE	**110.23***	**24.93***	**150.53***	**287.50***	**17.84***	491.40	**112.09***	43.48	**196.70**	**285.98***	**18.83***	**746.57***