Open Anomalous Trajectory Recognition via Probabilistic Metric Learning

Qiang Gao[1] , **Xiaohan Wang**[2] , **Chaoran Liu**[1] , **Goce Trajcevski**[3] , **Li Huang**[1] and **Fan Zhou**[2*]
[1]Southwestern University of Finance and Economics, Chengdu, China, 611130
[2]University of Electronic Science and Technology of China, Chengdu, China, 610054
[3]Iowa State University, Iowa, USA
qianggao@swufe.edu.cn, xiaohanwang@std.uestc.edu.cn, liuchaoran@smail.swufe.edu.cn
gocet25@iastate.edu, lihuang@swufe.edu.cn, fan.zhou@uestc.edu.cn

Abstract

Typically, trajectories considered anomalous are the ones deviating from usual (e.g., traffic-dictated) driving patterns. However, this closed-set context fails to recognize the *unknown anomalous trajectories*, resulting in an insufficient self-motivated learning paradigm. In this study, we investigate the novel **A**nomalous **T**rajectory **R**ecognition problem in an **O**pen-world scenario (**ATRO**) and introduce a novel probabilistic **M**etric learning model, namely **ATROM**, to address it. Specifically, ATROM can detect the presence of unknown anomalous behavior in addition to identifying known behavior. It has a *Mutual Interaction Distillation* that uses contrastive metric learning to explore the interactive semantics regarding the diverse behavioral intents and a *Probabilistic Trajectory Embedding* that forces the trajectories with distinct behaviors to follow different Gaussian priors. More importantly, ATROM offers a probabilistic metric rule to discriminate between known and unknown behavioral patterns by taking advantage of the approximation of multiple priors. Experimental results on two large-scale trajectory datasets demonstrate the superiority of ATROM in addressing both known and unknown anomalous patterns.

1 Introduction

The popularization of GPS-integrated devices and the comprehensive utilization of satellite sensor systems enabled a generation of a large number of location traces, yielding collections of diverse time-varying trajectories. Numerous researchers and practitioners endeavor to discover valuable knowledge from these trajectories and exploit it in specific applications like, for example, using diverse mobility patterns to provide accurate individual services. Broadly, identifying anomalous trajectories naturally becomes the primary concern in many downstream applications as it brings multiple benefits regarding the accuracy of certain services, e.g., promoting trajectory quality, governing valuable moving objects, discovering taxi driving fraud, etc. [Chen *et al.*, 2018; Liu *et al.*, 2020; Han *et al.*, 2022; Kieu *et al.*, 2022].

Most of the endeavors primarily seek to detect the anomaly in the sense that a particular trajectory deviates from the normal trajectories of a specific travel need, such as from a source S and a destination D [Belhadi *et al.*, 2020; Zhang *et al.*, 2020; Wang *et al.*, 2021]. Towards that, most traditional studies identify anomalies by following heuristic-based and learning-based paradigms, where the former mainly relies on density or distance measurements to find outliers, while the latter aims at employing machine learning (including deep learning) methods to obtain promising discriminative models [Meng *et al.*, 2019; Oh and Iyengar, 2019]. Heuristic-based approaches, such as iBAT [Zhang *et al.*, 2011] and TRAOD [Lee *et al.*, 2008], using multiple hand-crafted features, usually fail to discover the diverse patterns from normal trajectories – which in turn results in the degradation of specifying the anomalies. Due to the tremendous achievements in deep learning, recent studies turn to investigate the complex spatial-temporal dependent patterns underlying enormous trajectories' volumes for anomalous trajectory detection/recognition. For instance, using recurrent neural network (RNN) to tackle the diverse sequential information of trajectories [Song *et al.*, 2018] or devising an end-to-end framework with prior assumptions [Kieu *et al.*, 2022].

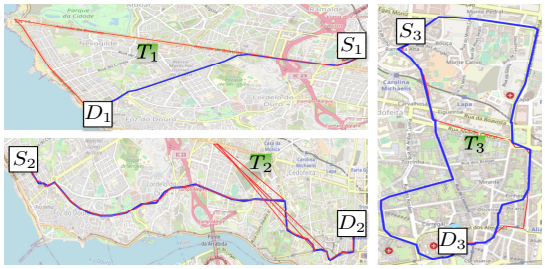


Figure 1: Examples of anomalous trajectories.

Despite significant breakthroughs achieved by the recent learning-based paradigms, existing approaches actually focus on tackling anomalous trajectory recognition in a closed-set scenario. They have achieved a promising performance in identifying the known anomalies, however, they cannot recognize any unknown anomaly that has never appeared in the learning systems. Fig. 1 illustrates the concerns. T_1 is a trajectory from the source S_1 to the destination D_1. We consider

*Corresponding author (Fan Zhou)

that it is an anomaly since it is significantly longer than the majority of daily routes. Thus, we call this case the *detour anomaly*. The second case is *navigation anomaly*. The trajectory (e.g., T_2) sometimes deviates from the normal routes due to the influence of satellite signal interference and other factors. These two cases are the protagonists of the anomalies encountered in the system. However, there could be an unseen anomalous pattern that has not yet been reported in the recognition database. For example, due to traffic accidents or road construction, there may be options for drivers to switch roads while driving (e.g., T_3), and this case may not be recorded in the existing database. Therefore, the learning system is likely to determine that this is a normal route. The reason is that the existing learning system is conditioned on the closed set (or incomplete knowledge of the world) and it cannot recognize unknown anomalies that occur in the future.

This is what motivates us to study a new interesting **A**nomalous **T**rajectory **R**ecognition problem in an **O**pen-world scenario (ATRO), rather than traditional anomalous trajectory detection (ATD). However, ATRO is a challenging task due to the facts that: (1) as a trajectory is a collection containing massive GPS points, multiple behavioral intents behind trajectories are unexplored, such as the correlations between an (S, D) pair and its associated points and the interactions between these points; (2) prior arts using probabilistic generative models, e.g., variational autoencoder (VAE) [Liu *et al.*, 2020; Kieu *et al.*, 2022], enable relieving the uncertainty underlying the limited recorded trajectories and boost the generation ability regarding the trajectory representation learning. However, they either approximate the trajectory distribution following a simple prior or fail to explore the diverse behavioral patterns, especially for unknown ones. In this sense, ATRO differs from traditional closed-set learning tasks as it needs to not only recognize patterns of known anomalies but also detect the unknown anomalous behavior.

To bridge the above gaps, we address the **ATRO** problem with probabilistic **M**etric learning (**ATROM**). In ATROM, we design two main procedures: – *Mutual Interaction Distillation* with intent distillation to contrastively capture the correlations between SD pairs and their associated GPS points at the grid level in addition to the mutual interaction between these GPS points; and *Probabilistic Trajectory Embedding* following Variational Bayes (VB) to correlate the behavioral patterns with massive trajectories. Unlike prior probabilistic models, we design a probabilistic metric rule to approximate the behaviors of trajectories with varying Gaussian assumptions, where we use KL divergence to measure the distance between the latent representation of a given trajectory and its possible patterns. In addition, we provide the natural rejection rule using probabilistic distance to detect whether there exists an unknown pattern from unseen trajectories. We summarize our contributions as follows:

- To the best of our knowledge, this is the first work that addresses anomalous trajectory recognition in an open-world scenario, i.e., ATRO. The main difference from prior arts is that ATRO aims to not only identify known patterns of anomalies but also discover possible unknown anomalies.
- We introduce a probabilistic metric learning approach, ATROM, to solve the ATRO problem. ATROM primarily takes the advantages of variational Bayes to explore the behavioral patterns of trajectories guided by the probabilistic metric rule. Also, it is the first attempt that considers the multiple interactive information underlying massive trajectories to improve the knowledge distillation.
- We conducted experiments on two large-scale taxi datasets, demonstrating the superiority of ATROM over several state-of-the-art baselines.

2 Preliminaries

Definition 1 (Raw Trajectory)**.** *A raw trajectory $T = \{p_1 \rightarrow p_2 \cdots \rightarrow p_n\}$ is a chronological sequence of GPS points. Each point p_i contains a geographical coordinate (e.g., longitude $p_i.lo$ and latitude $p_i.la$) and a timestamp $p_i.t$.*

Following recent representative studies [Li *et al.*, 2018; Liu *et al.*, 2020; Han *et al.*, 2022], we partition the geographical space into a grid of equal regions and map each GPS point into a region to relieve the modeling overhead problem. Given a city, we divide it into multiple non-overlapping regions by setting fixed physical dimensions. Assume that there are $m_c \times n_c$ equal-sized regions.

Definition 2 (Mapped Trajectory)**.** *For a given raw trajectory T, its corresponding mapped trajectory tr is obtained by mapping each p_i into the corresponding regions in which it is located. Assuming an enumeration (i.e., tokens) for the cells of the grid, tr can be represented as a sequence of region tokens, i.e., $tr = \{r_1 \rightarrow r_2 \cdots \rightarrow r_\tau\}$, $(\tau \leq n)$.*

Herein, we treat the $source$ and the $destination$ of a mapped trajectory tr as $S_{tr} = r_1$ and $D_{tr} = r_\tau$. In an open set scenario, we formalize the problem as follows:

Open Anomalous Trajectory Recognition. Given a source S and destination D pair (S, D) and the set of trajectories $\mathcal{D}$ between them, we assume that the trajectory patterns reported in the recognition system only have K known categories, i.e., $\{c_1, c_2, \cdots, c_K\}$ where c_1 refers to *normal*. The goal of open anomalous trajectory recognition is to *(i)* identify patterns of anomalous trajectories from $\mathcal{D}$ containing massive *normal* trajectories, and *(ii)* also detect whether there is an unknown/unseen anomalous pattern of trajectories not reported in the existing system, i.e., c_{K+1}.

3 Methodology

As shown in Fig. 2, ATROM mainly has two procedures: **(I) Mutual Interaction Distillation** is to distill two interactive semantics for the regions through which a large number of trajectories pass, resulting in a dense but semantic representation for each region. **(II) Probabilistic Trajectory Embedding** follows the rule of Variational Bayes and contains three key components: *Inference Net*, *Contrastive Net*, and *Generative Net*. Firstly, *Inference Net* takes the role of encoding each trajectory into a low-dimensional space, along with considering the prior dependencies between regions and the temporal information. Next, *Contrastive Net* attempts to force the trajectories with distinct patterns to follow different Gaussian priors. Finally, *Generative Net* enables the reconstruction of an input trajectory.

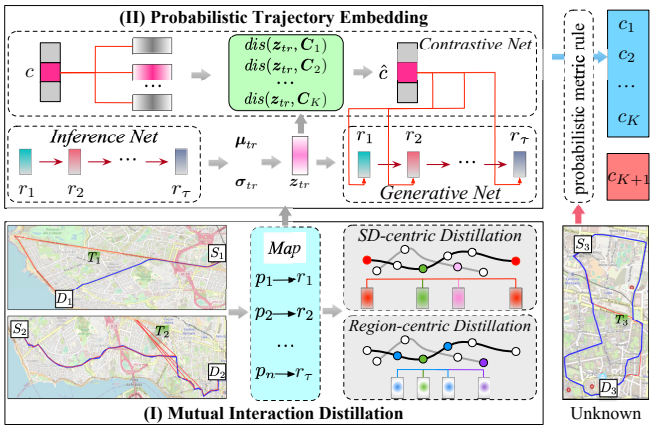


Figure 2: The framework of ATROM.

3.1 Mutual Interaction Distillation (MID)

Exploring the mutual interactions between different regions traveled by massive trajectories is a crucial prerequisite for downstream sequential information learning. Recent studies, e.g., [Liu *et al.*, 2020], usually concentrate on the capture of global dependencies of a specific trajectory based on random initialization of regions while ignoring context-aware learning at the regional level, which prevents the correlations between the various regions from being captured. We conjecture that there is a strong correlation between each region (or GPS point) and its corresponding (S, D) pair in a given trajectory, as each visited region in the trajectory may assume an important role to indicate if it is *normal* or *anomalous*. Furthermore, the fact that each trajectory traversed multiple regions indicates that these regions have possibly significant interactions reflecting behavioral patterns. Thus, we need to distill these two significant aspects underlying trajectories.

Trajectory Compression. Given an original trajectory set $\mathcal{D}$, we first transform it to a mapped trajectory set $\mathcal{D}_{tr}$ (cf. Definition 2). Consider an SD pair (S, D) from $\mathcal{D}_{tr}$. We search for a trajectory subset $\mathcal{D}_{tr}^{(S,D)} \subseteq \mathcal{D}_{tr}$, where each $tr \in \mathcal{D}_{tr}^{(S,D)}$ has the same SD pair. To efficiently enhance the learning of interactive semantics, we remove the consecutive repeated traveled regions of each trajectory in $\mathcal{D}_{tr}^{(S,D)}$, and formulate a compressed trajectory subset $\mathcal{PD}_{tr}^{(S,D)}$.

SD-centric Distillation. To distill interactions between each (S, D) and its associated regions, we adopt contrastive metric learning to obtain the SD-centric embedding for each region. Metric learning with contrastive loss [Sohn, 2016] is capable of maximizing the closeness between a given (S, D) and its relative (positive) region while minimizing the correlation between a given (S, D) and its irrelevant (negative) region. Hence, we can treat all the traveled regions regarding (S, D) as the positive samples and the remaining regions from trajectories of other SD pairs' as negative samples. Given a trajectory set $\mathcal{PD}_{tr}^{(S,D)}$, we first sample a region sequence $tr = \{r_1, r_2, \cdots, r_N\}$ with *source* $S_{tr} = r_1$ and *destination* $D_{tr} = r_N$. First, a trainable matrix $\boldsymbol{V}_1 \in \mathbb{R}^{n_c m_c \times d}$ is set as the initial region embeddings, where each $\boldsymbol{v}_1(r_i) \in \boldsymbol{V}_1$ is a unique representation of region r_i and d refers to the embedding size. Then, we extend recent contrastive loss setting [Gutmann and Hyvärinen, 2010; Oord *et al.*, 2018] and introduce a *sum* operation to tackle the (S, D) pair and correspondingly formulate an 'anchor' $\boldsymbol{a}_{SD}$, i.e., $\boldsymbol{a}_{SD} = \boldsymbol{v}_1(S_{tr}) + \boldsymbol{v}_1(D_{tr})$. Thus, the objective of SD-centric distillation is to maximize:

$$\mathcal{J}(\boldsymbol{a}_{SD}) = \mathbb{E}_{r \in tr \setminus (S,D)}[s\left(\boldsymbol{v}_1(r), \boldsymbol{a}_{SD}\right) - \log \sum_{j=1}^{J-1} \exp\left(s\left(\boldsymbol{v}_1\left(r'_j\right), \boldsymbol{a}_{SD}\right)\right)], \tag{1}$$

where '$\setminus$' denotes the set-difference operation and s refers to the similarity measure. This study uses the *cosine* function. r'_j is a negative sample and J denotes the sample number.

Region-centric Distillation. Another important interactive semantics is the adjacent proximity among traveled regions, as each trajectory is formed by sequentially passing these regions. Thus, we focus on the distillation of region-centric semantics by exploring the mutual interactions among the traveled regions. Given a regions r_i in the trajectory tr associated with SD pair (S, D), we treat $A(r_i) = \{r_{i-1}, r_{i+1}\}$ as the 'anchor', the r_i as the 'positive' sample, and the traveled region that does not appear in $\mathcal{PD}_{tr}^{(S,D)}$ as the 'negative' sample denoted as r'_j. Similar to SD-centric distillation, we define the objective of region-centric distillation as:

$$\mathcal{J}(r|(S, D)) = \mathbb{E}_{r \in tr \setminus (S,D)}[s\left(\boldsymbol{v}_2(r), \mathbb{A}(\boldsymbol{v}_2(r))\right) - \log \sum_{j=1}^{J-1} \exp\left(s\left(\boldsymbol{v}_2\left(r'_j\right), \mathbb{A}(\boldsymbol{v}_2(r))\right)\right)], \tag{2}$$

where $\mathbb{A}(\boldsymbol{v}_2(r_i)) = \boldsymbol{v}_2(r_{i-1}) + \boldsymbol{v}_2(r_{i+1})$ when $r = r_i$.

3.2 Probabilistic Trajectory Embedding (PTE)

Recalling recent studies [Kingma *et al.*, 2014; Guo *et al.*, 2021; Wang and Lan, 2021] on VAEs, we aim to estimate a conditional density $p(tr|c)$ by maximizing the evidence lower bound (ELBO), which can be specified as:

$$\begin{aligned} \log p(tr|c) \geq & - D_{\mathrm{KL}}\left[q(\boldsymbol{z} \mid tr) \| p(\boldsymbol{z} \mid c)\right] \\ & + \mathbb{E}_{q(\boldsymbol{z}|tr)}\left[\log p(tr \mid \boldsymbol{z}, c)\right], \end{aligned} \tag{3}$$

where D_{KL} refers to Kullback–Leibler divergence. In our context, Inference Net is to parameterize a posterior $q(\boldsymbol{z} \mid tr)$, yielding latent variables with low dimensions for a given set of trajectories. Generative Net is a likelihood function $p(tr \mid \boldsymbol{z}, c)$ that takes the role of reconstructing the input tr over the latent variable $\boldsymbol{z}$ conditioned on pattern category c. In sum, the first term in Eq. (3) is the distribution divergence between the posterior distribution $q(\boldsymbol{z} \mid tr)$ and a prior distribution $p(\boldsymbol{z}|c)$. The second term is the reconstruction loss where the generated trajectory is conditioned on $\boldsymbol{z}$ and c. In contrast to the standard divergence metrics [Kingma and Welling, 2013; Kingma *et al.*, 2014], we devise a Contrastive Net to measure the distribution distance between $q(\boldsymbol{z} \mid tr)$ and $p(\boldsymbol{z} \mid c)$, where we set $p(\boldsymbol{z} \mid c)$ to follow multiple Gaussian assumptions over pattern category c.

Inference Net. As each tr is a sequence of traveled regions, the Inference Net aims to encode them in a continuous latent space, where the underlying temporal dynamics should be considered as well. For each traveled region r_i in a given trajectory $tr = \{r_1, r_2, \cdots, r_\tau\}$, we first transform it into a dense representation by using the embedding results from the MID. Specifically, we fuse the SD-centric and region-centric semantics into a unified representation as the embedding of each region. For instance, the embedding of region r_i can be obtained by:

$$\boldsymbol{v}(r_i) = [\boldsymbol{v}_1(r_i); \boldsymbol{v}_2(r_i)]\mathbf{W}_r + \mathbf{b}_r, \tag{4}$$

where $[\cdot;\cdot]$ refers to the concatenation operation. As such, tr can be represented as $\{\boldsymbol{v}(r_1), \boldsymbol{v}(r_2), \cdots, \boldsymbol{v}(r_\tau)\}$. Next, we follow earlier works [Liu *et al.*, 2020] and use a recurrent neural network to receive the variable-length trajectories in order to address the underlying sequential dependencies. The hidden representation of each $\boldsymbol{v}(r_i)$ is obtained by:

$$\boldsymbol{h}_i = f_\phi(\boldsymbol{v}(r_i), \boldsymbol{h}_{i-1}), \tag{5}$$

where $\boldsymbol{v}(r_i)$ is the dense representation of r_i and $\boldsymbol{h}_{i-1}$ is the hidden state of previous token r_{i-1}. Herein, f_ϕ – denoting the recurrent neural cell with parameter set ϕ – can be the widely used LSTM (long short-term memory [Hochreiter and Schmidhuber, 1997]) or GRU (gated recurrent unit [Chung *et al.*, 2014]). In this study, we select the GRU as our recursive function since it is simpler and discards the complicated gate operations. In the end, we can receive a set of hidden states regarding trajectory tr, i.e., $\{\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_\tau\}$. Taking into account the contribution of all hidden states, we obtain the trajectory representation as follows:

$$\bar{\boldsymbol{h}} = \frac{1}{\tau}\sum_{1}^{\tau} \boldsymbol{h}_i. \tag{6}$$

To consider the uncertainties behind diverse trajectories, we use a posterior distribution to model the latent variable $\boldsymbol{z}_{tr}$ over $\bar{\boldsymbol{h}}$, which can be summarized as follows:

$$\boldsymbol{z}_{tr} \sim \mathcal{N}(\boldsymbol{\mu}_{tr}, \boldsymbol{\sigma}_{tr}^2 \odot \boldsymbol{\epsilon}), \tag{7}$$

$$\boldsymbol{\mu}_{tr} = g_1(\bar{\boldsymbol{h}}), \boldsymbol{\sigma}_{tr} = g_2(\bar{\boldsymbol{h}}), \tag{8}$$

where $g_*(\cdot)$ is the one-layer Multilayer Perceptron (MLP) function, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$, and $\odot$ is the element-wise product.

Contrastive Net. As each given trajectory tr implicitly corresponds to a pattern category (e.g., *normal*), we assume each pattern category to be a Gaussian distribution, i.e., $\boldsymbol{C}_k = \mathcal{N}(\tilde{\boldsymbol{\mu}}_k, \tilde{\boldsymbol{\sigma}}_k)$ where $\tilde{\boldsymbol{\mu}}_k$ and $\tilde{\boldsymbol{\sigma}}_k$ are trainable vectors. We employ contrastive loss to estimate the probability of the real pattern category of a given trajectory. Thus, we have:

$$\hat{c}_k = \frac{\exp^{-dis(\boldsymbol{z}_{tr}, \boldsymbol{C}_k)}}{\sum_{j=1}^{K} \exp^{-dis(\boldsymbol{z}_{tr}, \boldsymbol{C}_j)}}, \quad 1 \le k \le K. \tag{9}$$

dis is the distance measure function. Here, we choose the D_{KL} to measure the distance between the latent variable and possible category of a given trajectory and we define the distance between $\boldsymbol{z}_{tr}$ and $\boldsymbol{C}_k$ as:

$$dis\left(\boldsymbol{z}_{tr}, \boldsymbol{C}_k\right) = D_{\mathrm{KL}}\left[\boldsymbol{z}_{tr} \| \boldsymbol{C}_k\right]. \tag{10}$$

Generative Net. Following the rule of VAE [Kingma and Welling, 2013], our Generative Net operating in a recurrent generation process aims to reconstruct an input trajectory conditioned on its latent variable and possible pattern category. Specifically, given a predicted $\tilde{r}_{i-1}$ and its possible pattern category representation $\hat{\boldsymbol{c}}$, we have:

$$\tilde{r}_i = \arg\max(\tilde{\boldsymbol{h}}_i^r \mathbf{W}_r + \mathbf{b}_r), \tag{11}$$

$$\tilde{\boldsymbol{h}}_i = f_\theta([\boldsymbol{v}(\tilde{r}_{i-1}); \hat{\boldsymbol{c}}], \tilde{\boldsymbol{h}}_{i-1}). \tag{12}$$

Notably, we use the latent variable $\boldsymbol{z}_{tr}$ to generate the initial hidden state of Generative Net, defined as follows:

$$\tilde{\boldsymbol{h}}_0 = \mathrm{ReLU}(\boldsymbol{z}_{tr}\mathbf{W}_d + \mathbf{b}_d). \tag{13}$$

In this manner, the Generative Net will finally produce a trajectory $\tilde{tr} = \{\tilde{r}_1, \tilde{r}_2, \cdots, \tilde{r}_\tau\}$.

3.3 Training and Recognition

Training. By following the merit of metric learning with contrastive loss [Sohn, 2016], one of the key objectives is to minimize the distance between the latent variable $\boldsymbol{z}_{tr}$ over tr and its associated category representation, e.g., $\boldsymbol{C}_c$, where c is the real category. Hence, the first objective is to minimize:

$$\mathcal{L}_{KL}(\boldsymbol{z}_{tr}, c) = dis\left(\boldsymbol{z}_{tr}, \boldsymbol{C}_c\right). \tag{14}$$

Meanwhile, we also need to maximize the distance between $\boldsymbol{z}_{tr}$ and other category representations $\boldsymbol{C}_{k\neq c}$. Thus, the second objective is to minimize:

$$\mathcal{L}_{dis}(tr, c) = \frac{1}{K-1}\sum_{k\neq c}^{K}\left[l_k - dis\left(\boldsymbol{z}_{tr}, \boldsymbol{C}_k\right)\right], \tag{15}$$

where l_k is a margin loss [Lin and Xu, 2019] that enforces all $\boldsymbol{C}_{k\neq c}$ away from $\boldsymbol{C}_c$. In addition, we need to minimize the reconstruction loss (e.g., $\mathcal{L}_{rec}(tr, \tilde{tr})$) to ensure the latent variables can capture useful information from massive trajectories. We summarize that our final objective is to minimize:

$$\mathcal{L}(tr, c) = \mathcal{L}_{KL}(\boldsymbol{z}_{tr}, c) + \mathcal{L}_{dis}(tr, c) + \mathcal{L}_{rec}(tr, \tilde{tr}). \tag{16}$$

Recognition. After the model reaches its optimum, ATROM can well undertake the role of identifying the behavioral patterns of a set of given trajectories, where the patterns have been seen in the training process. To enable ATROM to detect the presence of an unknown anomalous pattern underlying the unseen trajectories, we devise a probabilistic metric rule inspired by [Lu *et al.*, 2022] to discriminate between known and unknown patterns by:

$$\hat{c} = \begin{cases} K+1, \text{ if } \max_k\left\{-dis\left(\boldsymbol{z}_{tr}, \boldsymbol{C}_k\right)\right\} < \beta \\ \arg\max_k\left\{-dis\left(\boldsymbol{z}_{tr}, \boldsymbol{C}_k\right)\right\}, \text{ otherwise.} \end{cases} \tag{17}$$

where β is the natural rejection threshold to determine if the pattern behind a given trajectory is an unseen anomaly (never appeared in the training set). Note that directly defining a threshold can introduce serious uncertainty into the recognition process. Instead, we dynamically determine the value of β by ranking the maximum probability of category distribution of each trajectory predicted by ATROM on the training set. We consider that the given trajectory will be detected as an unknown anomaly if its maximum value in its predicted category distribution is quite small. Therefore, we use 9-th decile [Peck *et al.*, 2015; Rana *et al.*, 2012] of the ranked list as the value of threshold β, indicating that most trajectories collected in the future usually fall into known patterns.

4 Evaluation

4.1 Experimental Setup

Datasets & Preprocessing. We conduct our experiments on two real-world taxi trajectory datasets. The first taxi dataset [Liu *et al.*, 2020] is collected from 442 taxis operating in the city of Porto during Jan 07 2013 to Jun 30 2014. Each taxi is equipped with a GPS device that can report its geospatial location every 15 seconds. We follow previous studies [Li *et al.*, 2018; Liu *et al.*, 2020] and also partition the geographical space into a grid of 100m $\times$100m cells. Then, we remove the SD pairs with less than 25 trajectories to obtain rich trajectories to indicate the normal routes/paths. The second dataset is collected from DiDi Chuxing[1], containing a large number of taxi traces generated from the city of Chengdu in Aug 2014. We partition the geographical space into a grid of 300m $\times$300m cells and remove the SD pairs with less than 40 trajectories. Table 1 summarizes the statistics of the preprocessed datasets, where '#' and 'Avg' denote the total number of (e.g., Trajectories) and the average length, respectively.

Dataset	#Points	#Trajectories	#(S, D)	Avg
Porto	11,635,104	262,574	4,567	44.31
Chengdu	24,266,393	414,414	11,860	58.56

Table 1: Trajectory Datasets after preprocessing.

Ground Truth. Following prior works [Zheng *et al.*, 2017; Zhang *et al.*, 2011; Liu *et al.*, 2020], we manually label the anomalous trajectories and treat the labeled trajectories as the ground truth for the fair evaluation. [Liu *et al.*, 2020] generates two different types of anomalous trajectories, i.e., *detour* anomalies and *route-switching* anomalies. We also generate the trajectories with GPS errors as anomalies, namely, *navigation* anomalies. For the training and testing setups, we use 90% of the trajectories as the training set and the rest as the testing set. In addition, we randomly inject two of the above anomalies into 70% trajectories in the training set and treat the rest of the training set as the normal routes. We inject the last type of anomaly that never appears in training set into the testing set and regard it as the unknown behavioral category.

Baselines. We compare ATROM with several state-of-the-art methods covering recently popular anomalous trajectory detection and trajectory learning methods, including: **SAE** [Malhotra *et al.*, 2016] is a conventional sequence-to-sequence model with RNNs, we use the GRU cell as the basic kernel for fair comparison; **ATD-RNN** [Song *et al.*, 2018] attempts to characterize the trajectory by learning the trajectory embeddings, whereafter it discovers anomalous trajectories in a supervised learning manner; **VSAE** [Liu *et al.*, 2020] is a sequence-to-sequence model based on variational inference, where trajectories are encoded into a latent space with multivariate Gaussian assumptions; **GM-VSAE** [Liu *et al.*, 2020] is a variational sequence AutoEncoder, where a Gaussian mixture model is jointly deployed to model the probability distribution of trajectories; **MainTUL** [Chen *et al.*, 2022] uses RNNs to explore the sequential patterns of given trajectories while a time-aware self-attentive module is able to learn long-term temporal dependencies; **VQRAEs** [Kieu *et al.*, 2022] employs quasi-recurrent neural networks for anomaly detection in time series data. Since these methods are conditional on closed-set scenarios and fail to detect unseen anomalies, we use our recognition method defined in Eq. (17) for unseen anomaly detection and fair comparison.

Evaluation Protocols. We select three common metrics for evaluations: F_1-score, Precision-Recall AUC (PR-AUC), and AUROC (Area Under ROC Curve). Specifically, PR-AUC and F_1-score are frequently used in anomalous trajectory detection [Liu *et al.*, 2020]. AUROC is commonly reported in out of distribution (OOD) tasks [Sun *et al.*, 2020].

Implementation Detail. We implemented ATROM and all the baselines in Python using the PyTorch library, accelerated by the NVIDIA Tesla A100. In the implementation, we set d to 128, J is 10, the size of hidden state is 256, and the learning rate is initialized as 0.001. We use Adam as the optimization algorithm. For reproducibility, the source codes are available at https://github.com/ypeggy/ATROM.

4.2 Overall Performance

Table 2 reports the overall performance of all the methods, where the best is shown in **bold** and the second best is underlined. We respectively select each anomalous pattern as the unknown for alleviating the behavior bias – i.e., *Unseen: Navigation* indicates that such an anomalous pattern does not appear in the training set.

Among the baselines, we observe that ATD-RNN and MianTUL perform well, which demonstrates that considering the sequential dynamics via trajectory embedding can boost the capture of behavioral intents underlying diverse trajectories. Traditional variational-based methods such as VSAE and GM-VSAE do not perform well on ATRO problem and possible reason is that they can facilitate generative capabilities regarding trajectory modeling, while failing to enrich the diversity between large-scale trajectories and their possible behavioral patterns. We find that ATROM consistently outperforms all baselines while it slightly underperforms SAE regarding F_1-score. For instance, ATROM yields 6.02% F_1-score, 12.5% AUROC, and 5.84% PR-AUC improvements compared to the best-performing baseline on Porto, where the pattern of detour anomaly is unknown. We consider that there are two reasons for the superiority of ATROM: (1) ATROM enables the distillation of the multiple task-oriented interactive semantics underlying massive trajectories, which not only explore the correlation between a specific travel need and its possible regions to travel but also consider the adjacent proximity that uncovers diverse interactions between traveled regions. (2) ATROM based on probabilistic metric learning is capable of forcing the trajectories with distinct behaviors to follow different priors, resulting in the advantage of disentangling trajectories into different latent spaces.

4.3 Ablation Study

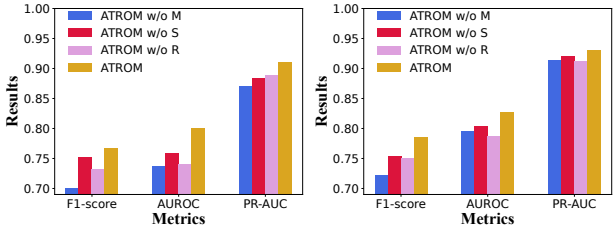
Interactive Semantics. We design three variants of MID in ATROM to investigate the contribution of interactive semantics, including: *ATROM w/o M* removes the MID procedure and uses a token embedding method [Liu *et al.*, 2020]

[1] http://outreach.didichuxing.com/research/opendata/

Dataset	Method	Unseen: Navigation			Unseen: Detour			Unseen: Route-switching		
		F_1-score	AUROC	PR-AUC	F_1-score	AUROC	PR-AUC	F_1-score	AUROC	PR-AUC
Porto	SAE	0.721	0.702	0.877	0.663	0.666	0.858	0.706	0.726	0.890
	ATD-RNN	0.717	0.704	0.884	0.681	0.678	0.856	0.741	0.755	0.904
	VSAE	0.243	0.503	0.750	0.309	0.499	0.750	0.242	0.504	0.753
	GM-VSAE	0.202	0.493	0.747	0.129	0.497	0.752	0.406	0.525	0.763
	MainTUL	0.709	0.721	0.888	0.667	0.651	0.849	0.709	0.736	0.896
	VQRAEs	0.367	0.468	0.744	0.317	0.466	0.750	0.421	0.577	0.797
	ATROM	**0.768**	**0.801**	**0.911**	**0.722**	**0.763**	**0.906**	**0.757**	**0.779**	**0.912**
Chengdu	SAE	0.638	0.634	0.840	**0.639**	0.616	0.848	0.127	0.490	0.870
	ATD-RNN	0.618	0.653	0.865	0.597	0.623	0.845	0.672	0.666	0.873
	VSAE	0.243	0.503	0.750	0.309	0.499	0.750	0.242	0.504	0.753
	GM-VSAE	0.216	0.503	0.750	0.434	0.640	0.863	0.139	0.517	0.758
	MainTUL	0.598	0.561	0.797	0.596	0.603	0.839	0.566	0.555	0.826
	VQRAEs	0.394	0.468	0.741	0.312	0.381	0.718	0.465	0.612	0.833
	ATROM	**0.785**	**0.828**	**0.931**	0.604	**0.673**	**0.879**	**0.753**	**0.749**	**0.892**

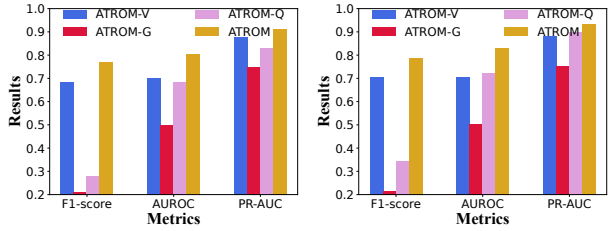
Table 2: Performance comparison on two large-scale taxi datasets.

to represent each unique region traveled by the trajectories; *ATROM w/o S* removes the SD-centric semantics in MID; and *ATROM w/o R* removes the region-centric semantics in MID. We choose *Navigation* anomaly as an unknown pattern. As shown in Fig 3, we find that ATROM w/o M performs poorly, even the worst on Porto, demonstrating that considering the multiple interactive semantics does help in promoting the performance of ATROM. Besides, removing each component results in performance degradation, indicating that both semantics contribute positively to performance improvement.



(a) Porto. (b) Chengdu.

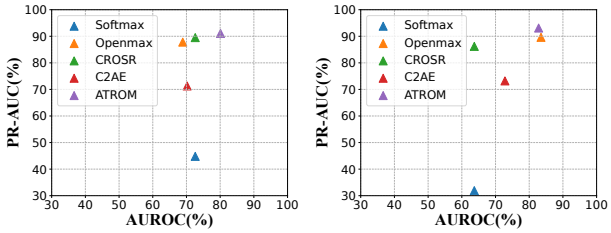
Figure 3: The contribution analysis of interactive semantics.



(a) Porto. (b) Chengdu.

Figure 4: The contribution analysis of trajectory learning.

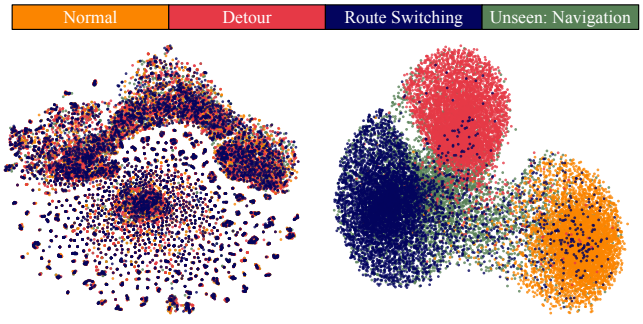
Trajectory Learning. To explore the impact of trajectory embedding in ATROM, we replace our PTE with the embedding in SAE, GM-VASE, and VQRAEs, respectively. And we formulate three variants, namely, *ATROM-V*, *ATROM-G*, and *ATROM-Q*. As Fig. 4 shows, we find that ATROM still performs best compared to these variants, suggesting that our embedding method derived from variational Bayes can effectively bridge the gap between behavioral patterns and massive trajectories with probabilistic metric learning.



(a) Porto. (b) Chengdu.

Figure 5: Comparison among distinct unknown recognition ways.

Unknown Recognition. To show the benefit of the unknown recognition rule based on divergence measure in ATROM, we choose four benchmark OOD detection methods for comparison: **Softmax** [Yoshihashi *et al.*, 2019] is a standard confidence-based method that provides the score of a predicted result. **Openmax** [Bendale and Boult, 2016] is an extension of the Softmax function. **CROSR** [Yoshihashi *et al.*, 2019] uses latent representations for input reconstruction and enables robust unknown detection. **C2AE** [Oza and Patel, 2019] uses the Extreme Value Theory (EVT) [Coles *et al.*, 2001] to discover the threshold for identifying known/unknown class instances. As shown in Fig. 5, we also set the *Navigation* anomaly as an unknown pattern. We can clearly observe that Openmax, CROSR (a variant of Openmax) and ours show better performance in terms of unknown anomaly detection. Importantly, ATROM performs more robustly than other methods and achieves the best PR-AUC.

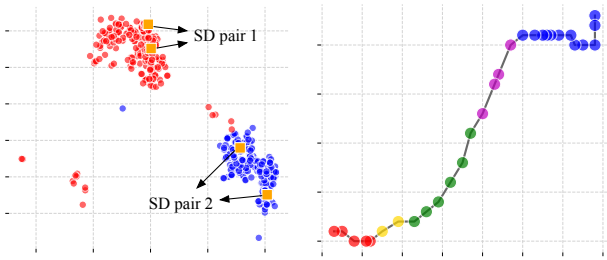


ADT-RNN ATROM

Figure 6: Visualization of trajectory embeddings in Porto.

Interpretability Analysis. We analyze whether ATROM can learn intuitive observations from an interpretable perspective. We use the t-SNE toolkit to visualize the distribution of trajectory embeddings from the testing set. As Fig. 6 shows, we compare the embedding results obtained by ATROM's PTE with ADT-RNN, which performs well on most metrics among the baselines. We can observe that ATROM enables the disentanglement of different behavioral patterns underlying trajectories. More importantly, ATROM enables the trajectories with an unknown pattern (marked in green) away from the trajectories with the *normal* pattern while closer to the anomalous patterns, which is what we expect.

In addition, we investigate the interpretability of two interactive semantics obtained by ATROM. To this end, we randomly select two different SD pairs from Chengdu dataset and also use the t-SNE toolkit to visualize the distribution of these two pairs' associated regions. As Fig. 7(a) shows, we can find that the associated regions of different SD pairs are well separated while well clustered within the same SD pair, which suggests that the MID designed in ATROM is able to refine the interactions between SD pairs and their associated regions. Second, we use k-means algorithm to cluster the region-centric embeddings and mark different colors for distinct clusters to investigate whether ATROM can successfully incorporate the adjacent proximity among traveled regions. As shown in Fig. 7(b), we can find that the consecutive regions in a randomly sampled trajectory are marked with the same color, which indicates that ATROM is capable of distilling the region-centric semantics.



(a) SD-centric. (b) Region-centric.

Figure 7: Visualization of interactive semantics in Chengdu.

5 Related Work

Traditional Anomalous Trajectory Detection. The majority of recent studies [Meng *et al.*, 2019; Wang *et al.*, 2021], fall into two main categories of anomaly detection techniques: (1) The heuristic (statistical)-based and learning-based mainly aim at discovering the popular or common routes from normal trajectories, referring to them as the *representative trajectories*. The distance or density metrics are then used to produce a deviation score between them and a target trajectory. A variety of distance metrics can be used, e.g., Euclidean Distance, Hausdorff Distance, Dynamic Time Warping, etc. [Laxhammar and Falkman, 2013; Meng *et al.*, 2019]. [Liu *et al.*, 2013] proposed a density-based trajectory outlier detection method, which takes account of the distribution of neighborhood objects. As anomalous trajectories are few and different, an isolation-based method, namely iBAT, was devised in [Zhang *et al.*, 2011] to discover the outlier trajectories. Most heuristic-based solutions heavily rely on statistical distance measurement on trajectories while ignoring the potential transitional dependencies among the locations. (2) The learning-based scheme provides a new perspective employing machine learning to the feature context from intricate trajectories [Schmidl *et al.*, 2022]. A k-nearest neighbor (kNN) method for identifying the group outlier trajectories was presented in [Djenouri *et al.*, 2021]. A novel time-series detection method based on one-class support vector machines was introduced in [Ma and Perkins, 2003]. However, traditional learning-based methods are mostly based on simple feature extraction and cannot handle well complex semantic dynamics or interactions.

Deep Trajectory Anomaly Detection. Recent deep learning works have explored the rich context of underlying trajectories via representation learning and have achieved remarkable gains for trajectory-based applications, including anomalous trajectory detection [Liu *et al.*, 2020; Li *et al.*, 2018]. A supervised learning method based on recurrent neural networks to explore the temporal dynamics behind the varying trajectories was introduced in [Song *et al.*, 2018]. An LSTM-based autoencoder to learn the normal traffic pattern as well as the latent features was devised in [Said Elsayed *et al.*, 2020], where the OC-SVM algorithm is combined to enhance the performance of anomaly detection. Due to the inherent uncertainty and diversity of trajectories, recent studies use deep generative models to capture the latent variability from complex and varying trajectories [Li *et al.*, 2018; Liu *et al.*, 2020; Han *et al.*, 2022; Huang *et al.*, 2022]. Most of them employ the seq2seq-based method with variational Bayes [Kingma and Welling, 2013; Wang *et al.*, 2019] to encode the trajectories into a low-dimensional space – e.g., GM-VSAE jointly uses a Gaussian Mixture model to learn the diversity underlying massive trajectories. DeepTEA, which considers complex traffic conditions, leverages the variational autoencoder to handle the time-dependent anomalies from a huge volume of trajectories [Han *et al.*, 2022]. However, existing methods with deep generative models are constrained by the closed-set context and fail to recognize unknown anomalies that may occur in the future. In contrast, ATROM uses probabilistic metric learning, which enables the handling of incomplete knowledge of the world and has the ability to detect whether there exists an unknown pattern from unseen trajectories.

6 Conclusion

We presented ATRO, a novel anomalous trajectory recognition problem in an open-world scenario, which has not been formally addressed in prior works. Correspondingly, we introduced a novel probabilistic metric learning model called ATROM which, unlike traditional anomalous trajectory detection methods, considers the multiple interactive semantics underlying massive trajectories datasets and correlates the (known and unknown) behavior with possible anomalous trajectories with the probabilistic metric rule. As part of our future work, we will investigate how to detect more unseen anomalous patterns with incomplete knowledge.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No.62102326), the Natural Science Foundation of Sichuan Province (Grant No.2023NSFSC1411), the Key Research and Development Project of Sichuan Province (Grant No.2022YFG0314), the NSF SWIFT 2030249, and Guanghua Talent Project.

References

[Belhadi *et al.*, 2020] Asma Belhadi, Youcef Djenouri, Jerry Chun-Wei Lin, and Alberto Cano. Trajectory outlier detection: Algorithms, taxonomies, evaluation, and open challenges. *ACM Transactions on Management Information Systems (TMIS)*, 11(3):1–29, 2020.

[Bendale and Boult, 2016] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1563–1572, 2016.

[Chen *et al.*, 2018] Pudi Chen, Shenghua Liu, Chuan Shi, Bryan Hooi, Bai Wang, and Xueqi Cheng. Neucast: seasonal neural forecast of power grid time series. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 3315–3321, 2018.

[Chen *et al.*, 2022] Wei Chen, Shuzhe Li, Chao Huang, Yanwei Yu, Yongguo Jiang, and Junyu Dong. Mutual distillation learning network for trajectory-user linking. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1973–1979, 2022.

[Chung *et al.*, 2014] Junyoung Chung, Caglar Gulcehre, Kyung Hyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[Coles *et al.*, 2001] Stuart Coles, Joanna Bawa, Lesley Trenner, and Pat Dorazio. *An introduction to statistical modeling of extreme values*, volume 208. Springer, 2001.

[Djenouri *et al.*, 2021] Youcef Djenouri, Djamel Djenouri, and Jerry Chun-Wei Lin. Trajectory outlier detection: New problems and solutions for smart cities. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(2):1–28, 2021.

[Guo *et al.*, 2021] Yunrui Guo, Guglielmo Camporese, Wenjing Yang, Alessandro Sperduti, and Lamberto Ballan. Conditional variational capsule network for open set recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 103–111, 2021.

[Gutmann and Hyvärinen, 2010] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304, 2010.

[Han *et al.*, 2022] Xiaolin Han, Reynold Cheng, Chenhao Ma, and Tobias Grubenmann. Deeptea: effective and efficient online time-dependent trajectory outlier detection. *Proceedings of the VLDB Endowment*, 15(7):1493–1505, 2022.

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

[Huang *et al.*, 2022] Tao Huang, Pengfei Chen, and Ruipeng Li. A semi-supervised vae based active anomaly detection framework in multivariate time series for online systems. In *Proceedings of the ACM Web Conference 2022*, pages 1797–1806, 2022.

[Kieu *et al.*, 2022] Tung Kieu, Bin Yang, Chenjuan Guo, Razvan-Gabriel Cirstea, Yan Zhao, Yale Song, and Christian S Jensen. Anomaly detection in time series with robust variational quasi-recurrent autoencoders. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 1342–1354. IEEE, 2022.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv:1312.6114*, 2013.

[Kingma *et al.*, 2014] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27, 2014.

[Laxhammar and Falkman, 2013] Rikard Laxhammar and Göran Falkman. Online learning and sequential anomaly detection in trajectories. *IEEE transactions on pattern analysis and machine intelligence*, 36(6):1158–1173, 2013.

[Lee *et al.*, 2008] Jae-Gil Lee, Jiawei Han, and Xiaolei Li. Trajectory outlier detection: A partition-and-detect framework. In *2008 IEEE 24th International Conference on Data Engineering*, pages 140–149. IEEE, 2008.

[Li *et al.*, 2018] Xiucheng Li, Kaiqi Zhao, Gao Cong, Christian S Jensen, and Wei Wei. Deep representation learning for trajectory similarity computation. In *2018 IEEE 34th international conference on data engineering (ICDE)*, pages 617–628. IEEE, 2018.

[Lin and Xu, 2019] Ting-En Lin and Hua Xu. Deep unknown intent detection with margin loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5491–5496, 2019.

[Liu *et al.*, 2013] Zhipeng Liu, Dechang Pi, and Jinfeng Jiang. Density-based trajectory outlier detection algorithm. *Journal of Systems Engineering and Electronics*, 24(2):335–340, 2013.

[Liu *et al.*, 2020] Yiding Liu, Kaiqi Zhao, Gao Cong, and Zhifeng Bao. Online anomalous trajectory detection with deep generative sequence modeling. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 949–960. IEEE, 2020.

[Lu *et al.*, 2022] Jing Lu, Yunlu Xu, Hao Li, Zhanzhan Cheng, and Yi Niu. Pmal: Open set recognition via robust prototype mining. In *Proceedings of the AAAI Conference*

on Artificial Intelligence, volume 36, pages 1872–1880, 2022.

[Ma and Perkins, 2003] Junshui Ma and Simon Perkins. Time-series novelty detection using one-class support vector machines. In *Proceedings of the International Joint Conference on Neural Networks, 2003.*, volume 3, pages 1741–1745. IEEE, 2003.

[Malhotra *et al.*, 2016] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. Lstm-based encoder-decoder for multi-sensor anomaly detection. *arXiv preprint arXiv:1607.00148*, 2016.

[Meng *et al.*, 2019] Fanrong Meng, Guan Yuan, Shaoqian Lv, Zhixiao Wang, and Shixiong Xia. An overview on trajectory outlier detection. *Artificial Intelligence Review*, 52(4):2437–2456, 2019.

[Oh and Iyengar, 2019] Min-hwan Oh and Garud Iyengar. Sequential anomaly detection using inverse reinforcement learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & data mining*, pages 1480–1490, 2019.

[Oord *et al.*, 2018] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.

[Oza and Patel, 2019] Poojan Oza and Vishal M Patel. C2ae: Class conditioned auto-encoder for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2307–2316, 2019.

[Peck *et al.*, 2015] Roxy Peck, Chris Olsen, and Jay L Devore. *Introduction to statistics and data analysis*. Cengage Learning, 2015.

[Rana *et al.*, 2012] Sohel Rana, Md Siraj-Ud-Doulah, Habshah Midi, and AHMR Imon. Decile mean: A new robust measure of central tendency. *Chiang Mai journal of science*, 39(3):478–485, 2012.

[Said Elsayed *et al.*, 2020] Mahmoud Said Elsayed, Nhien-An Le-Khac, Soumyabrata Dev, and Anca Delia Jurcut. Network anomaly detection using lstm based autoencoder. In *Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, pages 37–45, 2020.

[Schmidl *et al.*, 2022] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: a comprehensive evaluation. *Proceedings of the VLDB Endowment*, 15(9):1779–1797, 2022.

[Sohn, 2016] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29, 2016.

[Song *et al.*, 2018] Li Song, Ruijia Wang, Ding Xiao, Xiaotian Han, Yanan Cai, and Chuan Shi. Anomalous trajectory detection using recurrent neural network. In *International Conference on Advanced Data Mining and Applications*, pages 263–277. Springer, 2018.

[Sun *et al.*, 2020] Xin Sun, Zhenning Yang, Chi Zhang, Keck-Voon Ling, and Guohao Peng. Conditional gaussian distribution learning for open set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13480–13489, 2020.

[Wang and Lan, 2021] Zhen Wang and Chao Lan. Towards a hierarchical bayesian model of multi-view anomaly detection. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 2420–2426, 2021.

[Wang *et al.*, 2019] Yixin Wang, Andrew C Miller, and David M Blei. Comment: Variational autoencoders as empirical bayes. *Statistical Science*, 34(2):229–233, 2019.

[Wang *et al.*, 2021] Sheng Wang, Zhifeng Bao, J Shane Culpepper, and Gao Cong. A survey on trajectory data management, analytics, and learning. *ACM Computing Surveys (CSUR)*, 54(2):1–36, 2021.

[Yoshihashi *et al.*, 2019] Ryota Yoshihashi, Wen Shao, Rei Kawakami, Shaodi You, Makoto Iida, and Takeshi Naemura. Classification-reconstruction learning for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4016–4025, 2019.

[Zhang *et al.*, 2011] Daqing Zhang, Nan Li, Zhi-Hua Zhou, Chao Chen, Lin Sun, and Shijian Li. ibat: detecting anomalous taxi trajectories from gps traces. In *Proceedings of the 13th international conference on Ubiquitous computing*, pages 99–108, 2011.

[Zhang *et al.*, 2020] Mingyang Zhang, Tong Li, Yue Yu, Yong Li, Pan Hui, and Yu Zheng. Urban anomaly analytics: Description, detection, and prediction. *IEEE Transactions on Big Data*, 8(3):809–826, 2020.

[Zheng *et al.*, 2017] Guanjie Zheng, Susan L Brantley, Thomas Lauvaux, and Zhenhui Li. Contextual spatial outlier detection with metric learning. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 2161–2170, 2017.