



# Mobility Irregularity Detection with Smart Transit Card Data

Xuesong Wang<sup>1</sup>(✉), Lina Yao<sup>1</sup>, Wei Liu<sup>1,2</sup>, Can Li<sup>1</sup>, Lei Bai<sup>1</sup>,  
and S. Travis Waller<sup>2</sup>

- <sup>1</sup> Computer Science and Engineering, University of New South Wales,  
Sydney, Australia  
{xuesong.wang1,lina.yao,Wei.Liu,can.li4}@unsw.edu.au,  
baisanshi@gmail.com
- <sup>2</sup> Civil and Environmental Engineering, University of New South Wales,  
Sydney, Australia  
s.waller@unsw.edu.au

**Abstract.** Identifying patterns and detecting irregularities regarding individual mobility in public transport system is crucial for transport planning and law enforcement applications (e.g., fraudulent behavior). In this context, most of recent approaches exploit similarity learning through comparing spatial-temporal patterns between normal and irregular records. However, they are limited in utilizing passenger-level information. First, all passenger transits are fused in a certain region at a timestamp whereas each passenger has own repetitive stops and time slots. Second, these differences in passenger profile result in high intra-class variance of normal records and blur the decision boundaries. To tackle these problems, we propose a modelling framework to extract passenger-level spatial-temporal profile and present a personalised similarity learning for irregular behavior detection. Specifically, a route-to-stop embedding is proposed to extract spatial correlations between transit stops and routes. Then attentive fusion is adopted to uncover spatial repetitive and time invariant patterns. Finally, a personalised similarity function is learned to evaluate the historical and recent mobility patterns. Experimental results on a large-scale dataset demonstrate that our model outperforms the state-of-the-art methods on recall, F1 score and accuracy. Raw features and the extracted patterns are visualized and illustrate the learned deviation between the normal and the irregular records.

**Keywords:** Irregular pattern detection · Spatial-temporal profiling · Similarity learning

## 1 Introduction

Smart public transit card is now widely used in cities all over the world. It brings massive convenience for both passengers and transit operators in daily lives.

Online recharging and fare calculation can be automatically achieved with minimum labour cost involved. In this context, large volume of transit smart card data is continuously generated, which provides an unprecedented opportunity to profile passenger mobility patterns [8]. For example, the Opal card in the Greater Sydney area involves millions of users and records travel patterns of them on a daily basis. Uncovering recurrent patterns, and systematic variations, and irregular behavior patterns from these large datasets can aid in future transit route and stop planning or refinement, bus or train scheduling. Moreover, detecting irregular behavior patterns may also help alleviate or limit potential fraudulent behavior with smart transit card and subsequent loss.

The repetitiveness or regularity of travel patterns indeed allow planners and operators to optimally design public transport systems. However, in recent years, with a growing availability of individual mobility data, detecting irregularities regarding individual mobility in public transport system and quantifying the impacts of these become increasingly important, which can help the system operator and planner to more proactively accommodate both systematic behavior variations and stochasticity related to travel in public transit systems. At the same time, discovering and preventing potential fraudulent behavior with smart transit cards is also important in many cities. For example, the public transit card in cities such as Sydney, Melbourne, and Hong Kong is often linked to users' credit cards. Moreover, public transit cards in some cities (e.g., Octopus card in Hong Kong) can also be used to purchase daily goods from a variety of shops. Detecting irregular and potential fraudulent behavior with smart transit cards at early stages may help avoid considerable loss of travelers who lose their smart transit cards.

In order to address the problem of detecting irregular or fraudulent behavior in smart transit data, recent efforts have been made in spatial-temporal profile extraction and pattern comparison. First, convolutional-based or graph-based methods are adopted for describing spatial connections and layout for different stops [1, 5, 14]. Then a sequential layer is applied to model temporal correlations among historical transits. In this way, the model is able to predict a traffic-related value for a location at a timestamp. Finally, similarity-based algorithms are used to compare different passenger patterns. The similarity function can either be measured by a statistical distance metric between the normal and fraud data, or defined by a reconstruction error derived from an encoder-decoder structure [6, 13, 16]. Therefore, a reference passenger pattern can be discriminated from a potential irregular/fraudulent pattern.

Despite promising success in excavating spatial-temporal profile based on morphological layout or traffic flow, passenger level information is often fused within a certain region, which is inadequate when distinguishing fine-grained passenger profile since everyone has their own mobility patterns. Besides, most of the existing fraud or irregularity detection methods only consider all passenger data in an aggregate manner and learn a decision boundary for common outlier patterns. However, these methods tend to fail when normal data has high intra-class variance, meaning that normal data is not compacted and the boundary between the normal and fraud data is blurry, or reconstructed errors

are high even for normal data. This is exactly the case of mobility pattern where one passenger’s normal record can be another one’s abnormal data and totally confuses the aforementioned methods, thus calling for a personalised fraudulent behavior detection method.

In this paper, we build a deep learning framework to detect irregular/fraudulent behavior with respect to smart transit card (note that we may simply use “fraud” or “fraudulent” to refer to “irregular” behaviors later on). We managed to extract passenger level spatial-temporal profile and present a personalised similarity learning for detection. More specifically, a route-to-stop embedding is first proposed to extract spatial correlations between different transit stops and routes for each passenger. Then attentive fusion is adopted to find spatial repetitive and time invariant pattern. Finally, a personalised similarity function is learned to evaluate the historical and recent mobility patterns. To summarize, this paper makes the following major contributions.

- We propose a novel route-to-stop embedding to explore spatial correlations between routes and transit stops. It does not need to compute tens of thousands of global nodes in a graph. Instead, it only focuses on personalised nodes for each passenger, meanwhile maintain the ability to abstract node and edge correlations.
- We propose a learnable similarity function to measure the distance between repetitive invariant mobility pattern and recency pattern. Instead of integrating all the passenger data with high intra-class variance, the function directly applies to each passenger and makes personalised decision.
- We conduct experiments on a large-scale real-world dataset. Results demonstrated that using 20% of the total fraudulent data can achieve state-of-the-art performance. With overall data, our model gains significant improvements on F1 and accuracy.

## 2 Related Works

Recently a host of studies have investigated to extract spatial-temporal profile for a passenger. One typical approach is to use convolutional layers over a spatial map and to use recurrent layers over a time sequence. Lan et al. [7] and Chen et al. [3] adopted convolutional layers on the morphological layout of a city and LSTM to estimate travel time and predict urban air quality index. Recently convolutional graph networks have drawn more attention in helping discover non-linearity correlation between nodes and edges in a graph [5, 17]. Each node is represented by a vector while connections of each node are represented by an adjacency matrix or by fixed edge features. Nevertheless, most of existing approaches [1, 14] fuse passenger features in a node, while features on the same node can be dynamic for different passengers, since two passengers on the same stop may have different time slot preference. Also, usually nodes were built by segmenting city layout into hundreds of grids so as to compute adjacency matrix. However, in our scenario nodes are tens of thousands of transit stops, demanding highly expensive computation to build a traditional graph.

In order to distinguish unique passenger profile from potential irregular or fraudulent behavior, distance-based and reconstruction-based detection methods have been widely used. The distance-based approaches assume that normal data are compact in distribution and far from sparse fraud. Conventional methods include isolation Forest [9], and One-Class Support Vector Machine. Present deep learning-based models try to learn a distinguished density space between the normal and fraud data [10, 11, 15]. DevNet [10] defines a deviation loss based on z-score of a prior Gaussian distribution and squeezes the outliers to the tail of the distribution. Perera et al. [11] present a compactness and a descriptiveness loss for one-class learning. Yoon et al. [15] propose a learnable scoring matrix to detecting incongruity between news headline and body text. The reconstruction based methods adopt encoder-decoder structures and claim that fraud can not be well-reconstructed through the structure. Cao et al. [2] and Xu et al. [13] propose variation-auto-encoder based models with extensive experiments. Zhang et al. [16] detect fraud on reconstructed residual correlation matrix. LSTM-NDT [6] and OmniAnomaly [12] adopt dynamic thresholding on reconstructed feature errors. Despite great effectiveness, these approaches tend to underperform in our task due to high intra-class variance of normal data, where one passenger's normal records can be irregular/fraudulent for others.

*Summary.* The proposed method in this paper is fundamentally different from the literature in the following aspects. Firstly, the proposed route-to-stop embedding does not need to compute tens of thousands of node features in a graph. Instead, it only concentrates on personalised nodes for each passenger, meanwhile maintaining the ability of abstracting node and edge correlations. Furthermore, this study directly applies a personalised similarity function to learn the discrepancy between historical and new records for each passenger. In this way, the variance of data can be reduced by only focusing records of the same passenger. Therefore, the high intra-class variance problem of some existing approaches is avoided.

## 3 Proposed Approach

### 3.1 Data Characteristics

**Data Description.** We use public transit card of the Greater Sydney area (Opal card) in the case study. In total 187,000 passenger records were collected from April 1st to 30th, 2017. Each record is characterized with transit stop features and route features. Transit stop features consist of tap on/off time, longitude/latitude and postcode of stops, stop id and whether the tap is a transfer from another vehicle. Transit route features include vehicle type, vehicle id, duration, distance bands, journey costs and run direction. In this way, every record can be described as a starting stop to an ending stop through a route, and a passenger can be represented as a sequence of such records. As demonstrated in Fig. 1, there are four transit stops marked as red and three connecting routes for this passenger. Each stop has one or more records, representing the passenger pattern such as visiting frequency.

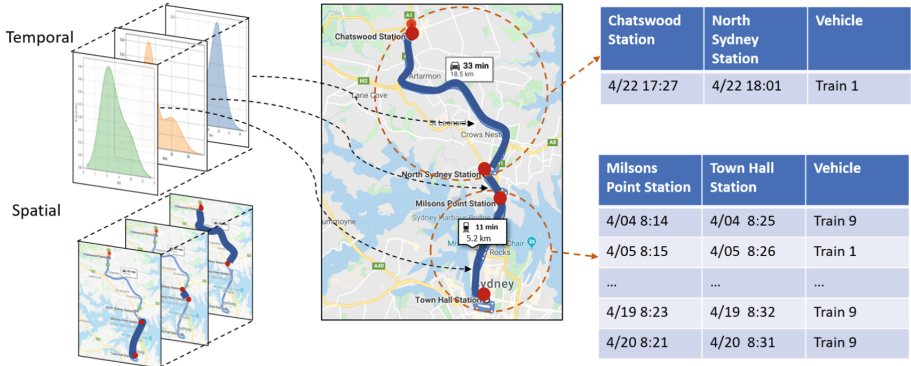


Fig. 1. Passenger spatial-temporal profile.

From Fig. 1, it is able to profile a passenger through spatial-temporal pattern. For instance, a route is made from Milsons Point Station to Town Hall Station nearly every weekday morning, indicating home/office location and potential occupation of the passenger. These repetitive stops together with less visiting stops, each with their own time invariant pattern, profile a unique passenger. We hereby aim to leverage this profile by excavating spatial correlations between transit stops and temporal preferences for each route, named “personalised spatial-temporal profile” for short. We expect to learn discriminative passenger profile, which can not only help transit operators provide more customised service, but can also alert passengers when irregular/fraudulent behavior occurs.

### 3.2 Problem Statement

Given a set of historical records  $X = \{x_1, x_2, \dots, x_N\}$  with  $x_i \in \mathbb{R}^D$  and a recent record  $x_{N+1}$  for a passenger, each record  $x_i = (s_{i1}, s_{i2}, r_i)$  can be viewed as a route  $r_i$  from a stop  $s_{i1}$  to a stop  $s_{i2}$  where  $(.,.)$  denotes concatenation. The goal is to learn a personalised similarity function  $\phi : (x_{N+1}, X) \mapsto \mathbb{R}^2$  in a way that  $x_{N+1}$  is similar to  $X$  if  $\phi(x_{N+1}, X)_1 > \phi(x_{N+1}, X)_2$ . Otherwise, it is a irregular/fraudulent behavior.

### 3.3 The Proposed Framework

Figure 2 presents the overall framework of the personalised spatial-temporal similarity learning network. Historical pattern  $u_N$  and the latest pattern  $u_{N+1}$  are extracted and compared from the corresponding data  $\{x_1, x_2, \dots, x_N\}$  and  $x_{N+1}$  to detect a irregular/fraudulent behavior. There are four major components in the framework: a route-to-stop embedding to describe the spatial mapping function from routes to stops, an attentive fusion to capture repetitive and time invariant pattern, a fully connected layer to extract recent pattern, and a similarity function to learn the discrepancy between the historical and recent patterns.

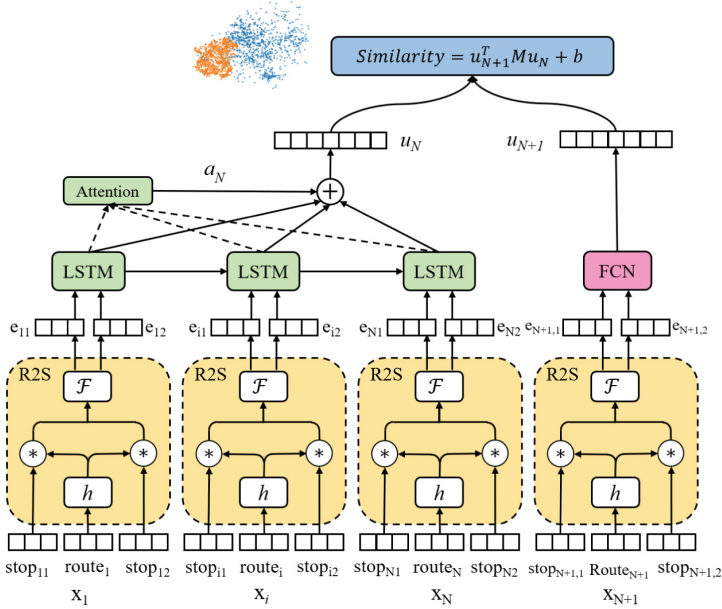


Fig. 2. The framework of personalised spatial-temporal similarity learning.

**Route-to-Stop Embedding (R2S).** As mentioned before, every record is featured by two transit stops and a route. Instead of directly input raw features  $x_i = (s_{i1}, s_{i2}, r_i)$  into the model, we want to map two stop features  $s_{i1}, s_{i2}$  into the same space since starting and ending stops share the same raw input space. Also, stop and route features  $(s_i, r_i)$  have strong dependency correlations, for example, in Fig. 1, given route features like vehicle id, time duration and cost and starting stop features like tap on time and location, it is possible to speculate corresponding ending stop features. Hence, route features should also be projected to the same space as transit stops. Inspired by [4], a route-to-stop(R2S) embedding that is generalizable for each passenger is proposed to represent spatial correlation:  $\{\text{route } r_i; \text{ starting stop } s_{i1} \mapsto \text{ ending stop } s_{i2}\}$ :

$$e_i = \mathcal{E}(s_{i1}, s_{i2}, r_i) = (\mathcal{F}[h(r_i) \odot s_{i1}], \mathcal{F}[h(r_i) \odot s_{i2}]) \tag{1}$$

where  $e_i$  is the embedding of the  $i$ -th record.  $\mathcal{F}$  and  $h$  denote neural networks.  $h(r_i)$  has the same dimension as  $s_{i1}$  and  $s_{i2}$ , and  $\odot$  denotes element-wise multiplication. What R2S has achieved is to use  $h$  to project route features into stop space and to share a projection  $\mathcal{F}$  between two stops.

The advantage of R2S layers lies in the ability to draw a customized spatial profile for each passenger. Different weights  $h(r_i)$  are assigned to stops. The starting and ending stops are treated equally with  $\mathcal{F}$ . In this way, two passengers who share the same starting stop will have distinguished weights on that stop during to other factors like duration or costs.

**Repetitive and Time Invariant Pattern.** A unique passenger profile lies in repetitive visiting stops and invariant transit time slot preference observed from historical records. To capture such repetitive and time invariant pattern  $u_N$ , an LSTM layer followed by an attentive fusion is adopted:

$$\begin{aligned} u_i &= LSTM(e_i, u_{i-1}), \quad \text{s.t.} \quad 2 \leq i \leq N, u_1 = \mathbf{0} \\ a &= softmax(\mathbf{u} * W^u) \in \mathbb{R}^{N \times 1}, \quad u_N = \sum_{i=0}^N a_i u_i \end{aligned} \quad (2)$$

where  $u_i$  is the hidden layer of the  $i$ -th record defined by the R2S embedding  $e_i$  and the hidden layer of the last timestamp  $u_{i-1}$ . Then, an attentive probability  $a_i$  is calculated for each  $u_i$  through attention matrix  $W^u$ . The attention matrix is learned to focus on repetitive and time invariant stops so that the extracted pattern  $u_N$  is distinguished from irregular/fraudulent behavior pattern.

**Recency Mobility Pattern.** For the latest record, the spatial profile  $e_{N+1}$  is directly fed into a fully-connected layer to get recency mobility pattern  $u_{N+1}$ :

$$u_{N+1} = FCN(e_{N+1}) \quad (3)$$

The structure of hidden layers were chosen as same as the lstm hidden layers in order to learn a homogeneous projection as the historical pattern.

**Personalised Similarity Learning.** Given historical and recent record patterns  $u_N$  and  $u_{N+1}$ , A similarity function is modified from [15] aiming to provide personalised discrepancy decision:

$$\begin{aligned} P_{fraud} &= (u_{N+1})^T M_1 u_N + b_1 \\ P_{normal} &= (u_{N+1})^T M_2 u_N + b_2 \\ P(u_{N+1}, u_N) &= softmax([p_{fraud}, p_{normal}]) \end{aligned} \quad (4)$$

where  $M_1, b_1, M_2, b_2$  are learnable parameters. If  $p_{fraud} > p_{normal}$ , then it is irregular/fraudulent behavior. Using  $u^T M u$  instead of directly adopting inner product  $u^T u$  introduces a learnable matrix that can diminish differences between normal and historical data meanwhile enlarge the distributional gap between historical and fraud data. Hence, during training process, a pair of three records, i.e., (normal, irregular/fraudulent, historical) is built intentionally. The loss function is defined as:

$$\mathcal{L}(u_{N+1}, u_N) = -\mathbb{E}_{u_{N+1} \sim u_N} \log[P(u_{N+1}, u_N)] - \mathbb{E}_{u_{N+1} \not\sim u_N} \log[1 - P(u_{N+1}, u_N)] \quad (5)$$

$u_{N+1} \sim u_N$  refers to normal data and  $u_{N+1} \not\sim u_N$  refers to fraud data. The reason why the problem is designed in a supervised manner is that a limited number of labeled anomalies can always provide critical prior knowledge for an

unsupervised model [10]. However, in many real-world applications it is unable to collect massive labels. In order to address this issue, we generate labels based on the intuition that  $u_{N+1}$  is a normal record if it comes from the same passenger as the historical records and it is an irregular record if it comes from other passengers. In this way, hand-crafted labels are not necessary.

## 4 Experiments

### 4.1 Experimental Settings

For the data set, records of 175,000 and 12,000 passengers were used as training and testing data. For each passenger, the number of historical records  $N$  is set to be 20. Embedding was adopted for categorical features. The overall dimension after the embedding is 322, with the dimension of stop1, stop2 and route features equalling 105, 105 and 112 respectively. For the R2S embedding  $\mathcal{E}$ , hidden units of the neural network  $h$  is the same as the stop dimension 105, and the number of hidden units of  $\mathcal{F}$  is 50. The hidden states and output dimensions of LSTM are set to be 50 and 30. The hidden units of the FCN are 50 and 30.

### 4.2 Experimental Results

**Baseline Methods.** We choose three fraud detection methods on general data: OCSVM, iForest and DevNet, and two fraud detection methods on time series data: LSTM-NDT, MSCRED as baseline models. Also, two variants of our method are compared to test efficiency.

- OCSVM. The One-Class Support-Vector-Machine profiles normal data distribution boundary and claims  $x_{N+1}$  a fraud if it is outside the frontier.
- iForest [9]. The isolation Forest finds anomalies far from distributed-dense data. Each passenger data  $(x_{N+1}, X)$  is fit with an iForest model and a score of  $x_{N+1}$  is given.
- DevNet [10]. A new deviation loss based on z-score is proposed for anomaly detection. The objective is to squeeze normal data into a small range and deviate outliers from this range using a network. Since it does not have sequential layers, all historical as well as the latest features were stacked to prevent input information loss.
- LSTM-NDT [6]. The LSTM with the Nonparametric Dynamic Thresholding uses LSTM for multivariate time series prediction and defines a reconstructed error threshold based on historical error mean and variance.
- MSCRED [16]. The Multi-Scale Convolutional Recurrent Encoder-Decoder encodes the correlation matrix for multivariate time series and uses the residual reconstructed matrix to detect a fraud.
- Conv-Sim. A variant of our approach that replaces R2S embedding with 1d convolution layers. A  $1 \times 1$  kernel whose channel equals to input dimension slides over the 20 historical data. The motivation is that all historical data should share the same kernel.



- R2S-Net. A variant of our approach whose similarity function is a fully connected neural network. The input of the function is the concatenation of historical and recent data representation, and the output is irregular/fraudulent score.

Each experiment is given 10 runs and the mean and deviation values of the metrics are displayed in Table 1. The general fraud detection methods (OCSVM, DevNet) outperforms the sequential methods (LSTM-DNT and MSCRED). This is probably due to high intra-variance of the normal data, whereas a binary decision for a general method is easier to make than generating reconstructed data for a sequential model.

Overall, R2S-Sim as well as two variants significantly improves the state-of-the-art algorithms and is robust on different runs. It achieves high performance on all of the metrics other than a skewed precision. It gains 0.11 and 15.6% improvements on F1 and accuracy over DevNet. Conv-Sim is the closest to our method since attentive fusion compensates for the lack of temporal layers in 1d-convolution. The accuracy of R2S-Sim is around 2% higher than R2S-Net, verifying the effectiveness of the learned similarity function.

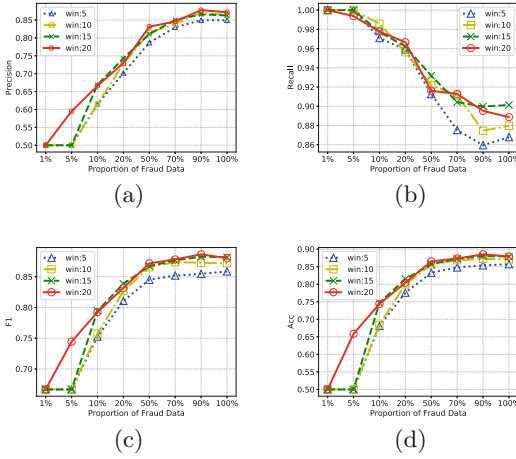
**Table 1.** Performance comparison

Method	Precision (Std.Dev.)	Recall (Std.Dev.)	F1 (Std.Dev.)	Acc (Std.Dev.)
OCSVM	<b>0.984 (0)</b>	0.286 (0)	0.443 (0)	0.641 (0)
iForest [9]	0.562 (0.002)	0.873 (0.006)	0.684 (0.003)	0.596 (0.004)
DevNet [10]	0.668 (0.028)	<b>0.900 (0.042)</b>	0.765 (0.006)	0.724 (0.018)
LSTM-NDT [6]	0.700 (0.003)	0.485 (0.006)	0.573 (0.005)	0.638(0.003)
MSCRED [16]	0.484 (0.010)	0.888 (0.078)	0.626 (0.028)	0.473 (0.017)
Conv-Sim	0.866 (0.007)	0.879 (0.007)	0.873 (0.002)	0.872 (0.002)
R2S-Net	0.847 (0.005)	0.880 (0.008)	0.863 (0.003)	0.861 (0.003)
R2S-Sim(ours)	0.868 (0.005)	<b>0.895 (0.005)</b>	<b>0.881 (0.003)</b>	<b>0.880 (0.003)</b>

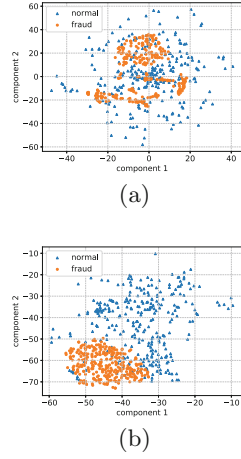
### 4.3 Ablation Study

*Injection of fraud data.* The original ratio of normal to fraudulent data is 1:1. We try to decrease the proportion of fraud training data. If 0% fraud data is injected, then it becomes a completely unsupervised problem. Hence, we use different proportion of fraud training data: {1%, 5%, 10%, 20%, 50%, 70%, 90%, 100%} to validate the performance. The ratio of test data is still 1:1 to keep consistency. Therefore if all the test data is treated as normal, the accuracy is 0.5 and the recall 1.0.

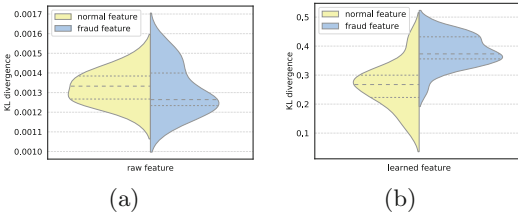
*Different Historical Window Size.* Default window size is 20 in the setting. We assume that a passenger tap a card twice a day, then 20 is approximately a collection of weekly data. Different window sizes including {5, 10, 15} were tested to see if passenger profile can be captured within a shorter time span.



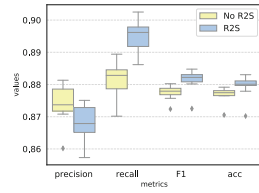
**Fig. 3.** Performance w.r.t. fraud data ratio and window size. (a) Precision. (b) Recall. (c) F1. (d) Accuracy.



**Fig. 4.** t-sne visualization on (a) raw features. (b) learned features.



**Fig. 5.** KL divergence between normal and fraud on (a) raw features. (b) learned features.



**Fig. 6.** Performance with/without R2S embedding.

Figure 3 demonstrates the metrics with respect to data proportion and window size. All test data was regarded as normal using 1% fraud data. However, with 5% data, the performance of the model using a 20-time-step sequence begins to improve. A pattern is observed that the longer the sequence, the less training data is needed for better performance. With 20% of the overall fraud data, all models can achieve the state-of-the-art accuracy of 0.7.

*Remove R2S Embedding.* Route-to-Stop embedding is removed and raw input features are input to attentive fusion. From Fig. 6, we can observe that except for precision, all the other metrics are improved with the embedding. Also, due to the reduced dimension of the input for the later LSTM layer, the overall training time is shorter than using raw features even with the added embedding.

#### 4.4 Visualization on the Passenger Pattern

In order to test if distinguishable patterns were learnt, a subset of the test data is visualized on raw features and learned features  $u_N, u_{N+1}$  (see Fig. 4). A t-sne model is trained with the pair (historical, normal, fraudulent). The relative vectors  $\mathbf{V}_{historical} - \mathbf{V}_{fraudulent}, \mathbf{V}_{historical} - \mathbf{V}_{normal}$  are displayed rather than  $\mathbf{V}_{historical}, \mathbf{V}_{fraudulent}, \mathbf{V}_{normal}$ . In this way, each data point in the plot is the difference between the historical and the latest representation of every passenger. Besides, test data was split into 40 groups and KL divergence was compared. Specifically,  $KL(normal||historical)$  and  $KL(fraudulent||historical)$  were compared and the violin plot is given in Fig. 5.

It is observed from Fig. 4(a) that the center of both normal and fraud cluster is around (0,0), meaning that the distribution of historical, normal and irregular/fraudulent raw data are homogeneous. This can also be concluded from Fig. 5(a) where difference between  $KL(normal||historical)$  and  $KL(fraud||historical)$  is negligible. After passenger pattern extraction, both differences are enlarged in Fig. 4(b) and Fig. 5(b), indicating that the model is pushing recent data away from historical one yet push irregular/fraudulent data harder so as to be distinctive. Also, there is a clear boundary between irregular/fraudulent and normal data.

## 5 Conclusions

In this study, we build a deep learning framework to detect irregular or fraudulent behavior with respect to smart transit card usage. We managed to extract passenger-level spatial-temporal profile and present a personalised similarity learning for detection. More specifically, a route-to-stop embedding is first proposed to exploit spatial correlations between different transit stops and routes. Then attentive fusion is adopted to find spatial repetitive and time invariant pattern. Finally, a personalised similarity function is learned to discriminate the historical and recent mobility patterns. Experimental results on a large-scale dataset demonstrate that the proposed model outperforms baseline methods in terms of recall, F1 score, and accuracy. Moreover, with 20% of the total fraud/irregular data, we can achieve state-of-the-art performance. Visualization on learned patterns reveals that the method can learn distinctive features among the irregularity/fraud and the normal records.

## References

1. Bai, L., Yao, L., Kanhere, S., Wang, X., Sheng, Q., et al.: Stg2Seq: spatial-temporal graph to sequence model for multi-step passenger demand forecasting. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI), pp. 1981–1987 (2019)
2. Cao, V.L., Nicolau, M., McDermott, J.: Learning neural representations for network anomaly detection. IEEE Trans. Cybern. **49**(8), 3074–3087 (2019)

3. Chen, L., Ding, Y., Lyu, D., Liu, X., Long, H.: Deep multi-task learning based urban air quality index modelling. *Proc. ACM Interact. Mob. Wearable Ubiquit. Technol.* **3**(1), 2 (2019)
4. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1263–1272. *JMLR. org* (2017)
5. Guo, S., Lin, Y., Feng, N., Song, C., Wan, H.: Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, 33, pp. 922–929 (2019)
6. Hundman, K., Constantinou, V., Laporte, C., Colwell, I., Soderstrom, T.: Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 387–395. *ACM* (2018)
7. Lan, W., Xu, Y., Zhao, B.: Travel time estimation without road networks: an urban morphological layout representation approach. In: *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1772–1778 (2019)
8. Li, C., Bai, L., Liu, W., Yao, L., Waller, S.T.: Passenger demographic attributes prediction for human-centered public transport. In: Gedeon, T., Wong, K.W., Lee, M. (eds.) *ICONIP 2019. CCIS*, vol. 1142, pp. 486–494. *Springer, Cham* (2019). [https://doi.org/10.1007/978-3-030-36808-1\\_53](https://doi.org/10.1007/978-3-030-36808-1_53)
9. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422. *IEEE* (2008)
10. Pang, G., Shen, C., van den Hengel, A.: Deep anomaly detection with deviation networks. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 353–362. *ACM* (2019)
11. Perera, P., Patel, V.M.: Learning deep features for one-class classification. *IEEE Trans. Image Process.* **28**(11), 5450–5463 (2019)
12. Su, Y., Zhao, Y., Niu, C., Liu, R., Sun, W., Pei, D.: Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2828–2837. *ACM* (2019)
13. Xu, H., et al.: Unsupervised anomaly detection via variational auto-encoder for seasonal KPIs in web applications. In: *Proceedings of the 2018 World Wide Web Conference*, pp. 187–196. *International World Wide Web Conferences Steering Committee* (2018)
14. Yao, H., Tang, X., Wei, H., Zheng, G., Li, Z.: Revisiting spatial-temporal similarity: a deep learning framework for traffic prediction. In: *AAAI Conference on Artificial Intelligence* (2019)
15. Yoon, S., et al.: Detecting incongruity between news headline and body text via a deep hierarchical encoder. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 791–800 (2019)
16. Zhang, C., et al.: A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 1409–1416 (2019)
17. Zheng, L., Li, Z., Li, J., Li, Z., Gao, J.: AddGraph: anomaly detection in dynamic graph using attention-based temporal GCN. In: *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pp. 4419–4425 (2019)