2019 IEEE International Conference on Big Data (Big Data)

Detection of Anomaly State Caused by Unexpected Accident using Data of Smart Card for Public Transportation

Sakura YAMAKI
Department of Computer Science and Communications Engineering, Graduate School of Fundamental Science and Engineering,
Waseda University, Tokyo, Japan,
s142613342@asagi.waseda.jp

Shou-de Lin
Department of Computer Science and Information Engineering,
National Taiwan University,
Taipei, Taiwan
sdlin@csie.ntu.edu.tw

Wataru KAMEYAMA
Faculty of Science and Engineering,
Waseda University, Tokyo, Japan
wataru@waseda.jp

Abstract—The railway is an indispensable means of transportation for people living in urban areas in Japan. However, unexpected accidents or disasters disturb the train operation. People usually check the operation status of trains on the official websites or Twitter of each railway company. However, it is still unclear whether such information is provided in realtime, when it is updated and which station is severely affected. Therefore, we tackle a real-world application of transportation big data using 8 months' data collected by smart cards for public transportation in Keikyu Line operating in Tokyo and Kanagawa Prefectures. We propose a method to detect the anomaly state by using the number of train users every 10 minutes in major 9 stations in Keikyu Line. In the method, outlier detections by interquartile range, interval estimation and Hotelling's theory are utilized to detect anomaly points. As the results, our proposal detects anomaly state better than the official announcement by Twitter on some points in terms of realtimeness, update frequency and geographic detail.

Keywords—anomaly detection, unexpected train accident, smart card for public transportation

I. INTRODUCTION

According to [1], the daily number of train passengers in the Greater Tokyo Area is approximately 20 million. Thus, railway is an indispensable means of transportation in Japan. On the other hand, the number of train delay is not less. According to Ministry of Land, Infrastructure, Transport and Tourism [2], train delay occurs in half of weekdays. [2] also shows that the most of big delays, i.e. over 30 minutes, are caused by sudden accidents, which makes difficult for passengers to predict large delays. Therefore, people daily check the operation state of trains at the official websites or Twitter of each railway company. From the viewpoint of the railway users, there are three requirements for train operation information: realtimeness, update frequency and geographic detail. There are various ever-proposed detection methods of anomaly state caused by unexpected accident. However, none of the existing methods satisfy the above requirements. Therefore, in order to cope with them, we propose an automatic detection method of anomaly state caused by unexpected accident using the smart card data for public transportation. The purpose of the method is to identify the stations that users should avoid using by automatically detecting congestion and suspension at each station caused by the accident regularly and in realtime. In order to confirm that our proposed method satisfies the three requirements, we use the smart card data for public transportation provided by Keikyu Line operating in Tokyo and Kanagawa Prefectures. And we evaluate the detection results by comparing them with the information announced by the official Keikyu Line Twitter [3], taking into account the following criteria:

1) Detecting more quickly than Twitter
2) Updating more often than Twitter
3) Detecting more detailed location than Twitter

To sum up, our contributions of this paper are:

- Our proposed method using smart card data can be an effective and alternative way for detecting anomaly state caused by unexpected accident.
- Our proposed method is better in accident announcement time than the official Twitter on some points in terms of realtimeness, update frequency and geographic detail.

The remaining of this paper is organized as follows. Section II describes a survey of anomaly state detection methods in public transportation. Section III describes an overview of the data we use and the data preprocessing. Section IV details the proposed method. Section V describes experimental conditions, results of prediction and anomaly detection models, and also evaluates the results by comparing our proposed method with the accident announcement time by the official Keikyu Line Twitter. Finally, we conclude this paper and describe future work in Section VI.

II. RELATED WORKS

Various approaches have been taken for anomaly detection for railway. The widely used one is a traffic monitoring method relying on various track sensors, such as loop detectors [4] and surveillance cameras [5]. However, the main problem is that the geographic detail of sensors is usually limited due to the high cost of deploying and maintaining them [6]. Another way is Twitter-based spatio-temporal anomaly detection for railways [7]. In this case, the analysis can be conducted without physical sensors and implemented in low cost. Some proposal achieves high accuracy around 90% for detecting unexpected accident in realtime such as [7, 8]. However, Twitter-based approach cannot detect the status at each station stably because tweets related to accident or delay are not always provided at each station. There are many studies using smart card data related to user behavior pattern analysis. However, there are few studies on anomaly detection using smart card data except for [9], which tackles anomaly detection caused by incident by applying NMF (Non-negative Matrix Factorization) to smart card log. However, as this method detects only anomaly day

978-1-7281-0858-2/19/$31.00 © 2019 IEEE

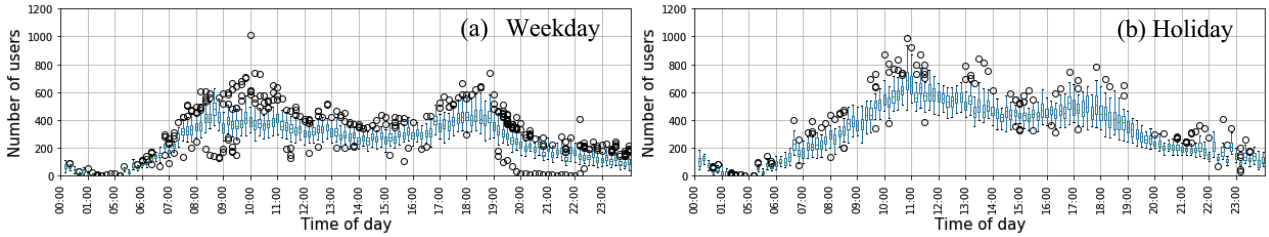

Fig. 2: Box Plot of Number of Users with Outliers from Apr. to Jul. in 2017. (a) Weekdays and (b) Holidays at Yokohama Station

but not anomaly time, it cannot satisfy realtimeness. Therefore, none of the existing methods for anomaly detection caused by unexpected accident satisfy the aforementioned three requirements.

III. Data Discription and Preprocessing

A. Smart Card Data

We use the data of Keikyu Line users collected by PASMO, one of the major smart cards for public transportation in Japan, from April 1st to July 31st in 2016 and from April 1st to July 31st in 2017. The average number of unique users per a month in the data is about 1.7 million.

B. Twitter Data

We can check the train operation information on the official Keikyu Line Twitter [3] such as shown in Fig. 1. Typically, location information in tweets is announced just by area, but not by station. TABLE I shows summary of the train accidents from April to July in 2017.

Keikyu Line operation information [official] @ keikyu_official June 2, 2017 9:22 [Operation information] Due to the personal injury that occurred at Keikyu Kamata Station, train operation is suspended in the inbound and outbound lines of the train between Shinagawa Station and Keikyu Kawasaki Station. Transfer transportation is being carried out on the route as following. JR Line, Tokyu Line, ...

Fig. 1: An Example of Keikyu Line Twitter Announcement (Translated)

TABLE I: Summary of Accidents from April to July in 2017

Month In 2017	Total Number	Cause of Accidents				
		Inspection	*Congestion*	*Personal Injury*	*Customer Treatment*	*Out of Order*
Apr.	24	6	8	5	4	1
May	20	7	3	3	6	1
Jun.	22	7	2	2	7	3
Jul.	18	8	1	4	4	0

C. Selecting 9 Stations

We have chosen the major 9 stations in Keikyu Line as targets for detecting anomaly state, that are Sengakuzi, Shinagawa, Haneda Airport Domestic Terminal, Keikyu Kamata, Keikyu Kawasaki, Yokohama, Kamiooka, Kanazawa Bunko, and Yokosuka Chuo.

D. Smart Card Data Preprocessing

We create the number-of-users log data every 10 minutes at the 9 target stations. TABLE II shows preprocessed data samples. And Fig. 2 shows the box plot with outliers on the number of users at Yokohama station for weekdays and holidays, which shows periodicity. Note that we exclude the data points from 2 o'clock to 4 o'clock where the railway is not operating, i.e. there are no data, and the days in vacation seasons and national holidays are treated as holiday.

TABLE II: Preprocessed Data Samples on Number of Users at Each Station

Date	Sengakuzi Station	Shinagawa Station	…	Yokosuka Chuo Station
2017-04-01 00:00:00	20	220	…	15
2017-04-01 00:10:00	11	96	…	16

E. Anomaly Point Labeling

From the information of the official Keikyu Line Twitter, we extract the time when the 9 target stations are affected by accident. Label "1" is attached to the preprocessed data corresponding to the anomaly points. Other data are taken as normal and labeled with "0". We have labeled anomaly data only for the smart card data of 2017.

IV. Proposed Methodology

As shown in Fig. 3, our proposed methodology has 4 steps in total. The first step is to obtain prediction errors by 8 prediction models. The second step is to divide the data by time and holidays/weekdays. The third step is to acquire the threshold by interquartile range, interval estimation and Hotelling's theory for each group of samples obtained in the second step. The fourth step is to evaluate the results of the detection models.

A. Step1: Building Prediction Model

Prediction-based anomaly detection can detect anomalies taking into account of context, i.e. seasonality and periodicity that are observed in our data as seen in Fig. 2. Details of the models used to predict the number of users at each station are as follows:

Prediction Models: We use 7 prediction models that are: Liner SVC, Logistic, Random Forest, Light GBM, Gradient Boosting, XGBoost, Multi-layer Perceptron.
Objective Variable: The number of users.
Explanatory Variables: They are summarized in TABLE III. Since the numbers in the items are categorical data, dummy conversion is applied to them.

TABLE III: Summary of Explanatory Variables

Variable Name	Meaning of Index
year	{"0": 2016, "1": 2017}
month	{"4": Apr., "5": Mar, "6": Jun., "7": Jul.}
weekday	{"0": Mon., "1": Tue., "2": Wed., ... , "6": Sun. }
weekday_holiday	{"0": Weekday, "1": Weekend or Holiday}
hour	{"0": 0, "1": 1, ..., "23": 23}
window_index	{"0": 1st 10 min. , , ..., "5": 6th 10 min.}

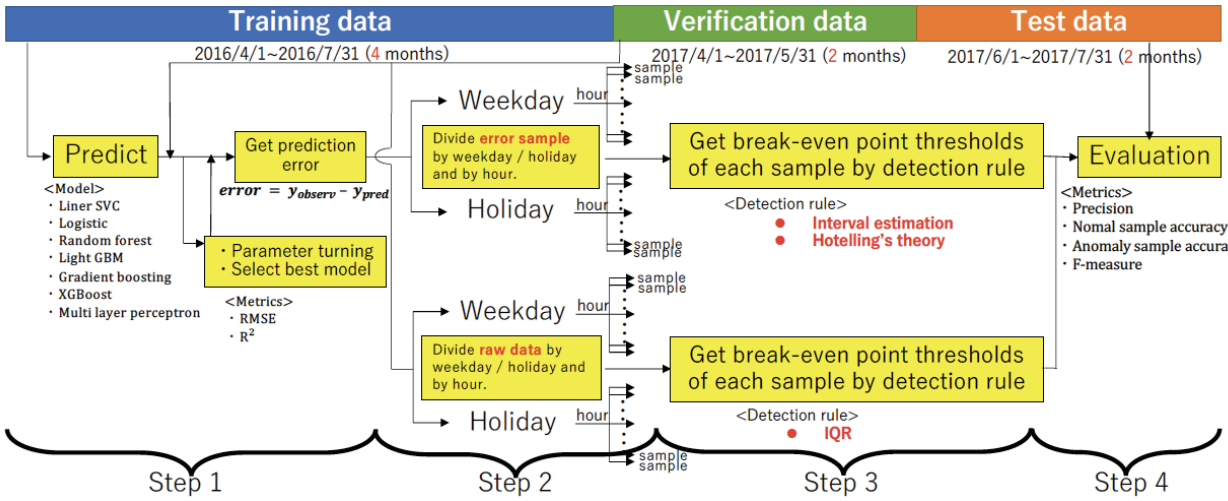

Fig.3: Proposed Methodology of Anomaly Detection Models

Parameter Turning: All hyperparameters are chosen based on each prediction model for each station data. We optimize hyperparameters based on mean squared error loss by Python automatic optimizing-paramter liblary optuna [10].

Evaluation Metrics: RMSE (Root Mean Square Error) and R^2 (Coefficient of Determination) are adopted as the evaluation metrics of the prediction models.

Prediction Error Calulation: it is calculated by (5) described in Subsection C.

B. Step2: Dividing the Data

In outlier determination, the inferred population depends on the each collected sample. Observing Fig. 2, it is inferred that the characteristic of data changes with time and weekday/holiday. Therefore, the data is divided by hour and by weekday/holiday, and a threshold is set for each sample.

C. Step3: Getting Threshold

Threshold Setting: We use three different detection rules: Inter-quartile Range (IQR) as a major nonparametric method, Interval Estimation and Hotelling's Theory with standard deviation as major parametric methods.

a) IQR: IQR measures the spread of the distribution by a single quantitative variable by the range between the third quartile Q_3 and the first quartile Q_1 as follows:

$$\boldsymbol{IQR = Q_3 - Q_1} \tag{1}$$

IQR is calculated from raw data. Anomaly label l is determined where outliers are labeled as "1" and inliers are labeled as "0" by a very simple set of calculations using new observation $y_{observ}{}'$, upper bound $bound_{upper}$, lower bound $bound_{lower}$ and parameter k as shown in (2):

$$l = \begin{cases} \text{"1"}, & \boldsymbol{y_{observ}{}' \geq bound_{upper}\ or\ y_{observ}{}' \leq bound_{lower}} \\ \text{"0"}, & \boldsymbol{bound_{lower} < y_{observ}{}' < bound_{upper}} \end{cases} \tag{2}$$

where

$$\boldsymbol{bound_{upper} = Q_3 + k \cdot IQR} \tag{3}$$

$$\boldsymbol{bound_{lower} = Q_1 - k \cdot IQR} \tag{4}$$

b) Interval Estimation: If the data follow the normal distribution, outliers can be detected by calculating the confidence intervals using the standard deviation σ. The standard deviation σ is calculated from errors that are the differences between observed value y_{observ} and predicted value y_{pred} as shown in (5).

$$\boldsymbol{error = y_{observ} - y_{pred}} \tag{5}$$

Outlier label l is calculated by (2), where $bound_{upper}$ and $bound_{lower}$ are set by using $y_{pred}{}'$ which is a new predicted value and parameter k as in (6) and (7):

$$bound_{upper} = y_{pred}{}' + k \cdot \sigma \tag{6}$$

$$\boldsymbol{bound_{lower} = y_{pred}{}' - k \cdot \sigma} \tag{7}$$

c) Hotelling's Theory: There are 3 steps in threshold setting using Hotelling's theory.

The first step is distribution estimation. Hotelling's theory assumes the data follow the normal distribution. Assuming that the population mean of the normal distribution is μ and the population standard deviation is σ, it is considered that the observation data x appears according to the density distribution in (8). μ and σ^2 are calculated using prediction errors as same as interval estimation method.

$$\boldsymbol{p(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)} \tag{8}$$

The second step is degree of anomaly calculation. If the data follow the normal distribution, the definition of anomaly degree is adopted as a negative log likelihood of probability density distribution. Therefore, the calculation of the degree of anomaly $a(x')$ is calculated by assigning x' shown in (9), where x' is a new observation not used to derive the mean and the variance as in (8)

$$\begin{aligned} a(x') &= -\ln p(x'|\mu,\sigma^2) \\ &= -\ln \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(x'-\mu)^2}{2\sigma^2}) \\ &= \boldsymbol{\frac{1}{2\sigma^2}(x'-\mu)^2 + \frac{1}{2}\ln(2\pi\sigma^2)} \end{aligned} \tag{9}$$

$$\boldsymbol{a(x') = \frac{1}{\sigma^2}(x'-\mu)^2 = \left(\frac{x'-\mu}{\sigma}\right)^2} \tag{10}$$

The third step is a threshold setting. According to Hotelling's theory Theorem 2.1 [11], the distribution of anomaly follows the χ^2 distribution with 1 degree of freedom and 1 scale factor. Each observation independently follows the same normal distribution, and the new observation x' independently follows the same normal distribution. At

this time, the constant multiple of the anomaly degree $a(x')$ follows the F-distribution of the degree of freedom (1, N-1). In particular, when $N >> 1$, $a(x')$ itself follows the χ^2 distribution with 1 degree of freedom and 1 scale factor:

$$\boldsymbol{a(x') \approx \chi^2(1,1)} \quad (11)$$

The $\boldsymbol{\chi^2}$ distribution can be expressed by (12) and (13) using degree of freedom $\boldsymbol{d}$, scale factor $\boldsymbol{s}$ and gamma function $\boldsymbol{\Gamma}$:

$$\boldsymbol{u = a(x')} \quad (12)$$

$$\chi^2(u|d,s) = \frac{1}{2s\Gamma\left(\frac{2}{d}\right)} \frac{\mu^{\left(\frac{d}{2}-1\right)}}{2s} exp\left(-\frac{1}{2s}\right) \quad (13)$$

Where d has the effect of changing the shape of the distribution in degrees of freedom, and s is the scale factor to adjust the size of the distribution. The important thing here is that the area of the χ^2 distribution becomes a probability α. Therefore, probability α can be calclulated as follows:

$$\alpha = \int_{ath}^{\infty} \chi^2(u|1,1)du = 1 - \int_{0}^{ath} \chi^2(u|1,1)du \quad (14)$$

Thus, the threshold value for anomaly determination is given by the probability value α, and the threshold value α_{th} is obtained from the χ^2 distribution table. Therefore, in outlier detection using Hotelling's theory, outlier label l is expressed by (15) by using $CHIINV$ which is the reverse function of the χ^2 distribution, degree of freedom d and the new error value $error'$:

$$l = \begin{cases} "1" \; if \;\; a(error') < \;\; CHIINV\,(\alpha, d = 1) \\ "0" \; if \;\; a(error') \geq \;\; CHIINV\,(\alpha, d = 1) \end{cases} \quad (15)$$

How to Choose the Best Thresholds: We use the index of break-even point accuracy to determine the point where the normal sample accuracy r_1 formulated by TP/(TP+FN) matches the anomaly sample accuracy r_0 formulated by TN/(FP+TN). The point before the anomaly sample accuracy which is lower than the normal sample accuracy is taken as the break-even point.

D. Step4: Evaluation

We consider precision, normal sample accuracy, anomaly sample accuracy and F-measure for performance metrics.

V. Experiment Results and Evaluation

A. Datasets

We split the whole 8 months data into the training data, the verification data and the test data as shown in TABLE IV. The building of prediction models is performed with the training data. The determination of the prediction model parameters, the determination of the best prediction model in each station data and the threshold setting by three methods as introduced in Section IV C performed with the verification data. Then, the anomaly detection model constructed by using the verification data are evaluated using the test data.

TABLE IV: Summary of Data Split

Data	Number of Users Data and Explanatory Variables Data	Anomaly Label
Training	From April 1st to July 31st in 2016	Not Used (All are taken as normal)
Verification	From April 1st to May 31st in 2017	Used
Test	From June 1st to July 31st in 2017	Used

B. Initial Threshold

Anomaly points may not be included in all samples. Therefore, in this case, it is necessary to set an initial value. The ratio of anomaly points at each station is under 0.7 %. So, we set k = 1.5 as the initial value for IQR because the value for the probability of anomaly samples occurrence is around 0.7 % when assuming a normal distribution in IOR. In the same way, k = 2.7 for the confidence interval. The threshold of Hotelling's theory is the appearance probability of the anomaly sample itself. So, we set α = 0.007 as the initial value. They are summarized in TABLE V with the ranges of the used thresholds.

TABLE V: Threshold Values (Interval: Interval Estimation)

Detection Rule	Variable	Start	End	Increment	Initial
IQR	k	0.1	5.0	0.1	1.5
Interval	k	1.1	5.0	0.1	2.7
Hotelling	α	0.05	0	-0.001	0.003

C. Results of Prediction and Anomaly Detection Model

The results of each prediction model at the 9 stations are shown in TABLE VI. We choose the best prediction model with the highest value of R^2 at each station, then it is used to predict test data for anomaly detection model.

The results of each anomaly detection model at the 9 stations are shown in TABLE VII. From the results, the anomaly detection models using the interval estimation is most accurate in all train stations in terms of F-measure.

D. Twitter vs Anomaly Detection

We compare the proposed method of interval estimation with Twitter information in terms of the three requirement as mentioned in Section I. In order to make the comparison clear, we focus on big delays over 30 minutes. There are 7 big accidents from June 1st to July 31st in 2017 as shown in TABLE VIII.

As shown in TABLE VIII, the tweet update frequency for accident takes over 20 minutes except for the accident at Nakakido Station occurred on July 28th, 2017. So, the proposed method can update more often than the official Twitter because it can detect an anomaly point every 10 minutes.

TABLE VIII also shows the time difference between the time when anomaly state firstly detected by our proposed method at each station and the accident announcement time by the official Keikyu Line Twitter. Note that the anomaly detection is done every 10 minutes in our proposal, so the value from +1 to +10 means that our proposal detects it almost at the same time of the announcement. Thus, it indicates that the proposed method can detect the anomaly state as quickly as the accident announcement time by the official Twitter.

Fig. 4 shows states of 9 stations before 1 hour and after 2 hours of the accident on June 2, 2017. From Fig. 4, it is possible to detect the time and the stations when and where the anomaly state occurs. Compared with the tweet announcing the operation status as shown in Fig. 1, the proposed method can provide more detailed location.

With all the aforementioned evaluations, our proposed method satisfies the three requirements.

TABLE VI: Evaluation Results of Prediction Models (The best for each metric is indicated by bold face.)

Station	Metrics	Linear SVC	Logistic	Random Forest	LGBM	Gradient Boost	XGBoost	MLP
Sengakuzi	***RMSE***	6.6364	6.0308	6.0828	4.9189	**4.9062**	4.9176	5.0818
	R^2	0.6397	0.7025	0.6973	0.8021	**0.8031**	0.8022	0.7887
Shinagawa	***RMSE***	203.9779	76.2959	79.6200	50.7632	**48.8061**	49.5589	258.4244
	R^2	-0.0118	0.8584	0.8458	0.9373	**0.9421**	0.9403	-0.6240
Haneda Airport Domestic Terminal	***RMSE***	52.1592	46.3675	45.5167	24.5307	24.2057	**24.1666**	37.0573
	R^2	0.6367	0.7129	0.7233	0.9196	0.9218	**0.9220**	0.8166
Keikyu Kamata	***RMSE***	20.8398	18.4773	18.2687	12.4985	**12.4722**	12.6044	14.1522
	R^2	0.6862	0.7533	0.7588	0.8871	**0.8876**	0.8852	0.8553
Keikyu Kawasaki	***RMSE***	50.8895	27.8793	39.1573	18.6571	**8.6096**	18.8138	1.7446
	R^2	0.5548	0.8664	0.7364	0.9402	**0.9405**	0.9392	0.7005
Yokohama	***RMSE***	179.0093	65.3876	87.6874	45.8653	45.2527	**44.9279**	190.8831
	R^2	0.0141	0.8685	0.7634	0.9353	0.9370	**0.9379**	-0.1211
Kamiooka	***RMSE***	144.3018	29.0868	40.6043	21.8203	21.7897	**21.6681**	58.5163
	R^2	-1.6552	0.8921	0.7898	0.9393	0.9395	**0.9401**	0.5634
Kanazawa Bunko	***RMSE***	32.9158	18.0758	18.6139	13.0636	**12.7585**	12.8762	15.1841
	R^2	0.3819	0.8136	0.8023	0.9026	**0.9071**	0.9054	0.8685
Yokosuka Chuo	***RMSE***	55.4177	27.4148	32.8736	18.8222	**18.7386**	18.8089	26.6673
	R^2	0.2097	0.8066	0.7219	0.9088	**0.9096**	0.9090	0.8170

TABLE VII: Evaluation Results of Anomaly Detection Models (The best for each metric is indicated by bold face.)

Station	Metrics	Precision	r_1	r_0	F-measure	TP	FN	TN	FP
Sengakuzi	***IQR***	0.144	0.887	0.957	0.247	55	7	7296	328
	Interval	**0.388**	0.726	**0.991**	**0.506**	45	17	7553	71
	Hotelling	0.062	**0.984**	0.880	0.118	61	1	6709	915
Shinagawa	***IQR***	0.034	0.727	0.881	0.065	32	12	6735	907
	Interval	**0.409**	0.818	**0.993**	**0.545**	36	8	7590	52
	Hotelling	0.011	**1.000**	0.493	0.022	44	0	3770	3872
Haneda Airport Domestic Terminal	***IQR***	0.030	0.941	0.933	0.058	16	1	7153	516
	Interval	**0.455**	0.588	**0.998**	**0.513**	10	7	7657	12
	Hotelling	0.008	**1.000**	0.714	0.015	17	0	5475	2194
Keikyu Kamata	***IQR***	0.038	0.561	0.924	0.071	23	18	7064	581
	Interval	**0.279**	0.707	**0.990**	**0.400**	29	12	7570	75
	Hotelling	0.023	**1.000**	0.772	0.045	41	0	5900	1745
Keikyu Kawasaki	***IQR***	0.035	0.675	0.903	0.067	27	13	6908	738
	Interval	**0.361**	0.650	**0.994**	**0.464**	26	14	7600	46
	Hotelling	0.016	**1.000**	0.670	0.031	40	0	5124	2522
Yokohama	***IQR***	0.049	0.746	0.881	0.092	47	16	6716	907
	Interval	**0.500**	0.778	**0.994**	**0.609**	49	14	7574	49
	Hotelling	0.017	**1.000**	0.527	0.034	63	0	4019	3604
Kamiooka	***IQR***	0.037	0.659	0.902	0.071	29	15	6894	748
	Interval	**0.367**	0.750	**0.993**	**0.493**	33	11	7585	57
	Hotelling	0.014	**0.977**	0.613	0.028	43	1	4686	2956
Kanazawa Bunko	***IQR***	0.039	0.788	0.917	0.075	26	7	7020	633
	Interval	**0.309**	0.879	**0.992**	**0.457**	29	4	7588	65
	Hotelling	0.014	**1.000**	0.704	0.028	33	0	5387	2266
Yokosuka Chuo	***IQR***	0.038	0.583	0.907	0.071	28	20	6926	712
	Interval	**0.287**	0.646	**0.990**	**0.397**	31	17	7561	77
	Hotelling	0.018	**0.958**	0.676	0.036	46	2	5163	2475

Interval: Interval Estimation

TABLE VIII: The Time Difference between the Time when Anomaly State Firstly Detected by Our Proposed Method at Each Station and the Accident Announcement Time by Official Twitter (The differences less than or equal to 10 minutes are indicated by bold face.)

Day & Time of Accident Announcement in Twitter	Tweeting Frequency	Num of Tweets	Accident Occurred Place or Station	Anomaly State Firstly Detected (min. from the announcement)									The Station Closest to Accident Place among the 9 Stations
				SG	***SH***	***KM***	***HA***	***KW***	***YK***	***KO***	***KB***	***YC***	
2017/6/2 9:22	Every 20 minutes	5	Keikyu Kamata	+18	**+8**	+28	+48	**+8**	+38	**+8**	+68	+18	KM
2017/6/12 8:16	Every 30 minutes	6	Idogaya	UD	+44	+24	UD	+44	**+4**	+14	UD	+14	KO
2017/6/13 11:58	Every 20 minutes	5	Toei Asakusa Line	**-18**	**-8**	UD	UD	+52	UD	UD	+62	UD	SG
2017/6/27 8:10	Every 2 hours	5	Keisei Line	+60	+50	+50	UD	+50	+40	+50	UD	+60	SG
2017/7/4 19:18	Every 30 minutes	6	Zoushiki	+32	**+2**	**+2**	UD	UD	UD	**+2**	UD	UD	KM
2017/7/20 0:00	Every 20 minutes	2	Rokugoudote	UD	UD	**+10**	UD	**+10**	UD	+60	+70	UD	KW
2017/7/28 22:04	Every 5 minutes	8	Nakakido Station	UD	**+6**	**+6**	+26	+16	**+6**	+56	+26	+26	YK

SG: Sengakuzi, SH: Shinagawa, KM: Keikyu Kamata, HA: Haneda Airport Domestic Terminal, KW: Keikyu Kawasaki, YK: Yokohama, KO: Kamiooka, KB: Kanazawa Bunko, YC: Yokosuka Chuo, UD: Undetected.

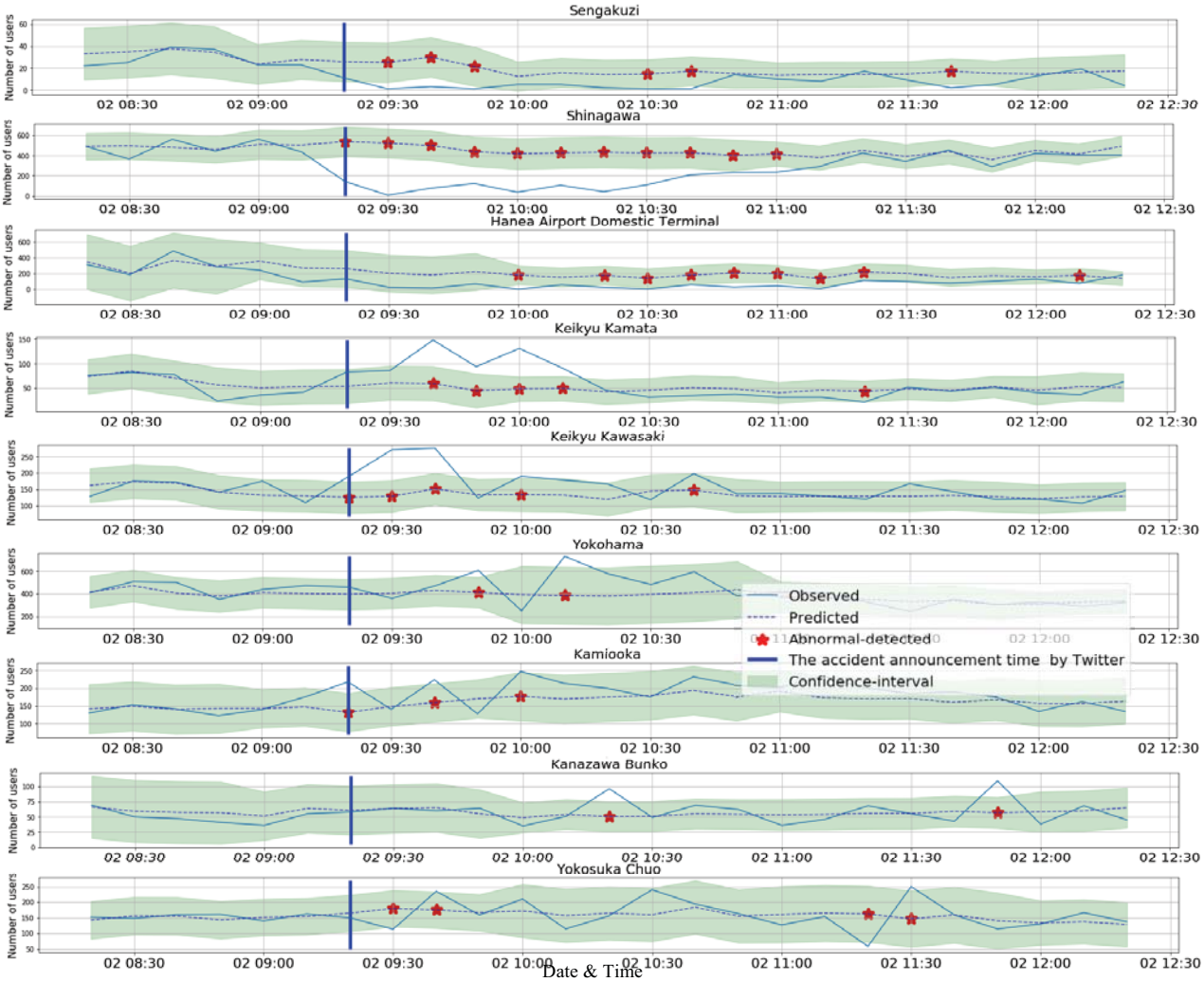

Fig. 4: States of 9 Stations before 1 Hour and after 2 Hours of Accident on June 2, 2017

VI. Conclusion and Future Plan

We propose a method to detect anomaly state from normal state by using the data of smart card for public transportation, where the number of passengers every 10 minutes in the 9 major stations in Keikyu Line is utilized. As the results, regression-tree-based Gradient Boosting and XGBoost are the best in the prediction models for some stations, and the interval estimation is the best in the detection models. The comparison of the results with the official Keikyu Line Twitter announcement shows that our proposed method is better in terms of update frequency, geographic detail and better in some station in terms of realtimeness, with remaining issues of the threshold optimization as a future work.

Acknowledgment

We would like to thank to Keikyu Corporation for providing us with the PASMO smart card data.

References

[1] Ramon Brasser, “Tokyo's rush hour by the numbers”, 2015, http://www.elsi.jp/en/blog/2015/11/blog1126.html (accessed 2 August 2019).

[2] Ministry of Land, Infrastructure, Transport and Tourism, “Delay Visualization of Railways in the Greater Tokyo Area (FY 2017)”, 2019 (in Japanese), https://www.mlit.go.jp/common/001269352.pdf (accessed 30 July 2019).

[3] The official Twitter of Keikyu Line, https://twitter.com/keikyu_official (accessed 2 August 2019).

[4] Yuan Y., Lint HV., Wageningen-Kessels FV. and Hoogendoorn S., “Network-wide traffic state estimation using loop detector and floating car data”, Journal of Intelligent Transportation Systems, vol. 18, no. 1, pp. 41-50, 2014.

[5] Ozkurt C. and Camci F., “Automatic traffic density estimation and vehicle classification for traffic surveillance systems using neural networks”, Mathematical and Computational Applications, vol. 14, no. 3, pp. 187-196, 2009.

[6] Zheng Y., “Methodologies for Cross-Domain Data Fusion: An Overview”, IEEE Transactions on Big Data, vol. 1, no. 1, pp. 16-34, 2015.

[7] Wang Y., Siriaraya P.,Yukiko K. and Toyokazu A., “Twitter-based Traffic Delay Detection based on Topic Propagation Analysis using Railway Network Topology”, Personal and Ubiquitous Computing, vol. 23, no. 2, pp. 233-247, 2019.

[8] Takuya Y., Hedetaka M., Hoichi Y.,Eiji A., and Hiroshi N., “Aoutomatic Notification for Train Delay from Twitter”, IPSJ SIG Technical Report, vol. 2013-DD-89, no.1, 2013.

[9] Tonlierera E., Baskiotisa N., V. Guiguea, P. Gallinaria, “Anomaly detection in smart card logs and distant evaluation with Twitter: a robust framework”, Neurocomputing, vol. 298, pp. 109-121, 2018.

[10] Optuna, https://optuna.org/ (accessed August 2 2019).

[11] Bai Z., and Saranadasa H., “Effect of high dimension: By an example of a two sample problem”, Statistica Sinica 6, pp.311-329, 1996.