ATD 2020: Traffic Anomaly Detection

Shan Huang, Xiao Wang

November 2020

1 Introduction

The goal of the challenge is to detect anomalous examples using limited observations in traffic flow data. In the dataset, anomalous examples are defines to be the outliers that are away from the majority. More specifically, considering the distribution of traffic flow on a particular weekday, within a particular hour of the day and at a particular location, the anomalies are defined to be the data points that lie outsider three standard deviations from the mean.

The challenge of this contest is to classify data points into normal observations and anomalous observations using only partially observed data. In the contest, several different fractions of observed data was given. Apparently, if we are given all of the data, we can perfect predictions as we know the exact distribution of the traffic flow of a particular hour, weekday and location. When data is observed only partially, we are forced to inference the underlying distribution using a sample of data of a limited size.

Since we know how anomalous data points are labeled (i.e. anomalous examples are those outside three standard deviation), the problem narrows down to estimate the population variance. A simple way to estimate population variance is to use the sample variance of the observed samples. Although sample variance is always an unbiased estimator of the population variance, the quality of this estimator (the variance of this estimator) is largely depend on the sample size. When the sample size is too small, we have poorly estimated population variance because the variance of the estimator is too large.

Therefore, we must leverage sampled data from other distributions (e.g. traffic flow of another hours, weekdays, locations etc.) if these distributions are similar to, or correlated to the one that we want to estimate variance on. It is important to note that using samples from other distributions is risky if these distributions are different from the one we estimate variance on. So we must try to use other samples which have the best similarity of distribution compared the one we are working on before using other samples. In other words, although we may reduce the estimator variance by increasing the sample size, the estimator tends to be more and more biased as we including examples from other distributions.

Consequently, the key to construct a good estimator is identifying how similar different distributions are so that we could use samples from similar distri-

Table 1: Priority queue of adding data to the sample

Priority	Weekday	Weekends
1	Adding adjacent weekdays, same hour	Adding the other weekend day, same hour
2	Adding rest of the weekdays, same hour	-
3	Adding adjacent hours of the same day	Adding adjacent hours of the same day
4	Adding second adjacent hours of the same day	Adding second adjacent hours of the same day
5	Adding data from closest station, same hour, same day	Adding data from closest station, same hour, same day

butions to increase the sample size, without sacrificing too much on the average quality of sample. For example, it is intuitive that distributions of two consecutive hours might be similar, as usually we do not expect drastic traffic flow change in one hour (but off course this could possibly happen). Another key element has to do with the sample size. As we increase sample size, we inevitably downgrade the sample quality. And the trade-off suggest that we must pick the optimal sample size when constructing the variance estimator. More technical details will be discussed in the method section.

For code, dataset and prediction results of our work, please refer to out Github repository: https://github.com/b19e93n/ATD2020TrafficAnomalyDetection.

2 Method

The idea of our method is gradually increase the number of data points n in the sample, until the size of the sample give us the most credible mean and standard deviation of the traffic flow. We use a priority queue when we add new data points in the sample. The priority is summarized in Table 1.

We applied our model on the detrended dataset.

2.1 Weekday and Weekends

Our Priority queue have completely separated the data from weekends and weekdays. We made this judgement from the observation that weekdays and weekends have very different behavior. Fig. 1 is an example of traffic flow distribution of different weekdays at 7 am, station 10. We also did a quantitative measurement of how different the traffic flow distribution between weekdays by calculating their normalized WasserStein Distance, shown in Fig. 2. We observe that the WasserStein Distance in between two weekdays and in between the two weekends are much smaller than WasserStein Distance in between a weekday and weekend.

Although further investigation shows that there is also noticeable difference between the data distribution of Saturday and Sunday, due to the general lack of data of weekends, we still put combining the data from the other weekends on the first place of priority queue for weekends.

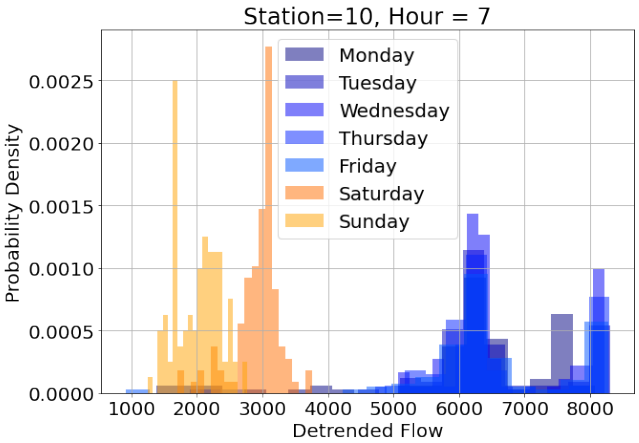

Figure 1: An Example of traffic flow distribution on different weekdays at 7 AM, station 10. We can see that weekdays and weekends form different groups

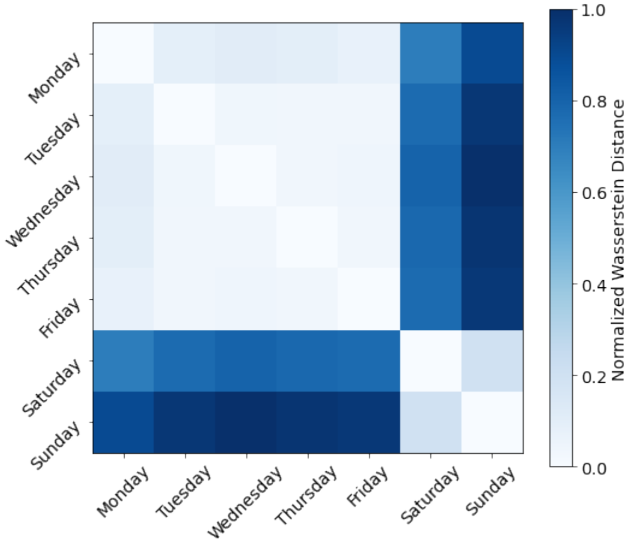

Figure 2: The WasserStein Distance between different weekdays. The bigger the distance, the more different the two probability distributions

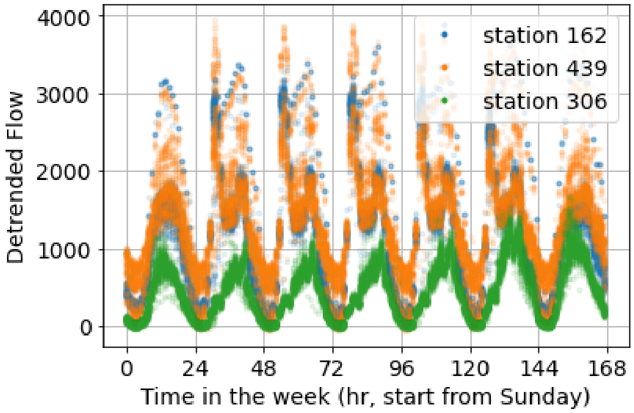

Figure 3: An example of traffic flow drawn from 3 stations next to each other in city 1. Station 439 is in between the other two stations, on the same road, appearing in the same "Fraction Observed" group.

2.2 Spatial data vs. Temporal Data

We put the priority of adding geographical information to be lower than adding temporal information. We made this judgement based on the observation that the average distance between two stations are far. These 3 datasets (City1,2,3) each contains 500 stations, each of the "Fraction Observed" group contains 100 stations. We have observed considerable changes in the distribution of traffic flow even between the closest two stations that is likely on the same road.

We also believe that the stations given in the datasets only covers a small portion of all roads on the actual map, as we see considerable rises or drops on two immediate adjacent stations on the same road, that could only be explained by side-track roads where no stations are set up.

To further illustrate the above two points, Figure 3 shows three stations that appears to be on the same road, immediate adjacent to each other. (Note that we can only use stations from the same "Fraction Observed" group, according to the challenge). We cal see that station 162 and station 439 has similar distributions, but station 306 has very different distribution than the station 439, although they are both adjacent station of 439. We believe that this is because there are out-branching roads in between station 306 and station 439, which was not shown in the map. This greatly reduced the correlation in between two adjacent stations, therefore lowered the priority of adding data using spatial information. In fact, in section **??** we will show that indeed spatial information should have a much lower priority than temporal information to be added to the sample using Wasserstein distance.

We have tried to incorporate the spatial information in a higher priority. In that experiment, we tried both simply adding data from adjacent station (same weekday, same hour), or adding these data with an adjustment so that

Table 2: F1 score on City 1, comparing spatial information first and temporal information first under near-optimal threshold choice n = 20.

Fraction	F1: Spatial Information First	F1: Temporal Information First
0.01	0.125	0.308
0.02	0.184	0.367
0.05	0.399	0.374
0.10	0.540	0.584
0.20	0.698	0.698

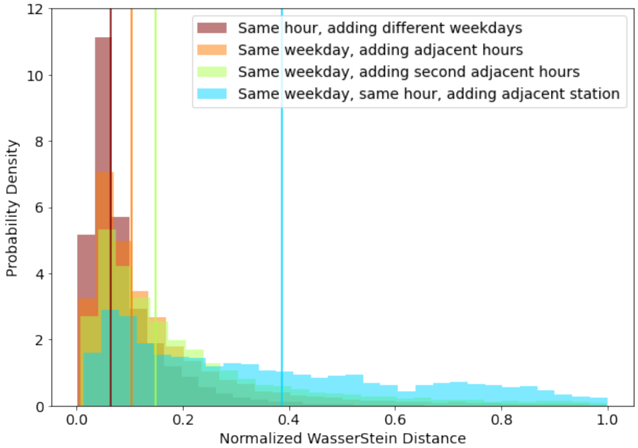

Figure 4: Wasserstein distance between samples before and after adding different categories of data. The histogram is drawn over different [station, weekday, hour] combination. Solid lines refers to the median value.

the distribution looks similar to the target station. Both of these experiment dragged down our overall F-1 scores by an average of 0.08 (see Table 2).

2.3 Priority of Adding Data

We gave our order of adding in data in Table 1. How we decide we will take took such an order is by comparing the overall Wasserstein Distance of traffic flow distribution of the target sample (given station, weekday, hour) to the expanded sample after adding data. We compute such WasserStein Distance for all combinations of Station, Weekday and Hour, and draw a histogram shown in Figure 4. We see the distributions of same station, same hour, but different weekdays have much more similarity than same hour, same weekday but adjacent stations. Same conclusion applies to same station, same weekday but adjacent hours. Therefore, adding data from adjacent stations should have lowest priority.

Table 3: F1 score on City 1, comparing choosing a threshold of 2.6 std and 3.0 std under near-optimal threshold choice n = 20. A smaller threshold is favored in a smaller fraction.

Fraction	F1: threshold: 2.6 standard deviation	F1: threshold: 3.0 standard deviation
0.01	0.392	0.308
0.02	0.452	0.367
0.05	0.509	0.374
0.10	0.617	0.584
0.20	0.681	0.698

Table 4: Fine-tuning result of optimal threshold and optimal number of data points n on City1.

Fraction	Optimal # of data n	Optimal Threshold (std)	F1-score (City 1)
0.01	15	2.40	0.4498
0.02	20	2.50	0.4595
0.05	20	2.70	0.5183
0.10	30	2.70	0.6296
0.20	30	2.90	0.7037

2.4 Anomaly Threshold

Although anomaly in this challenge is clearly defined as "Outliers that is more than 3 standard deviation from the mean value of the population", we find from preliminary experiments that using a smaller threshold than 3 stds in small fraction datasets may give us better result. Although the mean value from the sparsely drawn sample is the best un-biased estimation of the mean value, the standard deviation will be over-estimated. As is shown in Table 3, smaller threshold favors small fractions. We will show this is true later with the fine-tuned results of thresholds we choose for different fractions.

2.5 Fine-Tuning

We conducted fine-tuning on finding both the optimal threshold of number of data points and the optimal threshold (in std) for each fraction on dataset of City1. the result is shown in Table 4

3 Experimental Results

With the fine-tuned parameters, we apply our model to City2 and City3, we got following results, shown in Table. 5 and Fig. 5. Table. 6 shows comparison of our result to the current leaderboard results. It shows that we are better than the current 1st place winner. Unfortunately we were not allowed to submit our results since we already submitted one result (issue 17) before the original deadline of Oct. 1.

Table 5: F1 score for City2 and City3 under optimal parameters (threshold and number of data points).

Fraction	F1: City 2	F1: City3
0.01	0.4232	0.3839
0.02	0.5187	0.4803
0.05	0.5757	0.5493
0.10	0.6232	0.6632
0.20	0.6987	0.7066
Macro-F1	0.5699	0.5567

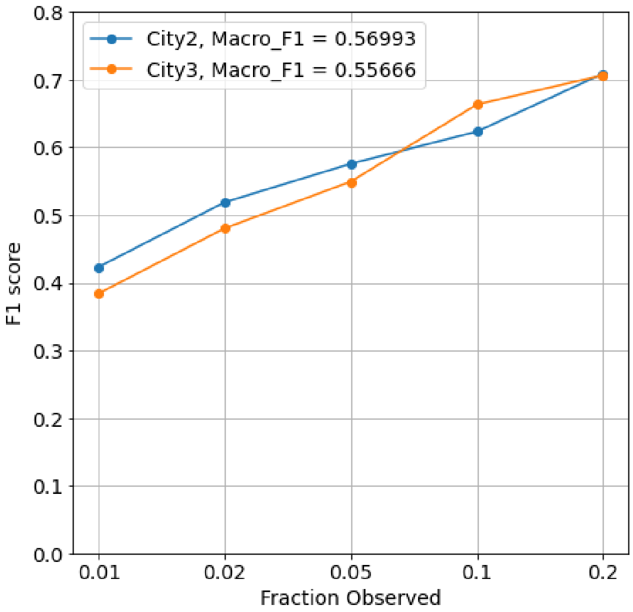

Figure 5: Result of our model on City2 and City 3

Table 6: Our Result compared to the leading board results

Place	Gitlab Issue #	F1: City 2	Gitlab Issue #	F1: City3
N/A	Our Model	**0.569926**	Our Model	**0.556660**
1	6	0.551284	20	0.553277
2	9	0.536932	13	0.540105
3	7	0.486214	19	0.530534
4	5	0.479521	17	0.485046
5	11	0.453511	18	0.478363