Unsupervised Anomaly Detection in Multivariate Spatio-Temporal Datasets using Deep Learning

Yildiz Karadayi[1,2]

[1] Kadir Has University, Istanbul, Turkey
[2] Innova, Istanbul, Turkey
ykaradayi@innova.com.tr

Abstract. Techniques used for spatio-temporal anomaly detection in an unsupervised settings has attracted great attention in recent years. It has extensive use in a wide variety of applications such as: medical diagnosis, sensor events analysis, earth science, fraud detection systems, etc. Most of the real world time series datasets have spatial dimension as additional context such as geographic location. Although many temporal data are spatio-temporal in nature, existing techniques are limited to handle both contextual (spatial and temporal) attributes during anomaly detection process. Taking into account of spatial context in addition to temporal context would help uncovering complex anomaly types and unexpected and interesting knowledge about problem domain. In this paper, a new approach to the problem of unsupervised anomaly detection in a multivariate spatio-temporal dataset is proposed using a hybrid deep learning framework. The proposed approach is composed of a Long Short Term Memory (LSTM) Encoder and Deep Neural Network (DNN) based classifier to extract spatial and temporal contexts. Although the approach has been employed on crime dataset from San Francisco Police Department to detect spatio-temporal anomalies, it can be applied to any spatio-temporal datasets.

Keywords: Unsupervised anomaly detection, multivariate, spatio-temporal data, deep learning.

1 Introduction

By the advancement of the hardware technology for data collection, generation of contextually rich data has become part of many processes. Data from many applications of today's world are temporal in nature such as sensor data, financial data, sales transaction data, and system diagnostics data. In addition to time context, many temporal data have also another context called spatial. In such settings where a spatial attribute is also a contextual attribute, we face with a new type of anomalies: spatiotemporal anomalies. Anomalies and outliers are two terms used most commonly in the context of anomaly detection; sometimes interchangeably [1].

In spatial or temporal data domains, attributes are partitioned into contextual and behavioral attributes. In such cases, behavioral attribute values (e.g. temperature at current time, money spent in a specific location) are treated as dependent variables [2]. Contextual attributes (e.g., location and time) are used to build neighborhoods in which the model of the normal data is built to predict the dependent variables or quantify the outlier scores of each data point within the neighborhood. In a credit card fraud detection scenario, dependent variable might be the total amount of many spent in a given time period, whereas independent variables might be customer demographics data, location, etc. In some datasets, it is possible for both spatial and temporal attributes to be contextual attributes. Such data can be referred to as spatiotemporal data. In spatiotemporal datasets, behavioral attributes like sea-surface temperatures, car speed, and transaction amount are often measured in the context of specific time or location. In these cases, both spatial and temporal continuity plays important role in identifying anomalies. A spatial-temporal outliers (ST-Outlier) are objects whose behavioral (non-spatial and non-temporal) attributes are different from other objects in their contextual neighborhoods [5].

Spatio-temporal data is extremely common in many problem settings where collecting data from various spatial locations for the nature of the problem are important. We need to emphasize that spatial and temporal continuity may not be equally important in all problem settings. For example, in an application where water temperature of an ocean is measured every minute by sensors located in many different locations, spatial continuity may be more important than temporal continuity. On the other hand, time context might play more significant role or might have at least equal significance along with spatial context in finding irregular spending patterns in a spatio-temporal financial dataset. For example, a customer cannot use his or her credit card in two different stores in 5 minutes if there is 100 km. distance between two stores.

There have been many studies on finding anomalies in time-series data considering only temporal context, or finding anomalies in spatial data considering only spatial context. There are limited researches on finding spatio-temporal outliers (ST-Outlier) which considers both context at the same time. Most of the ST-Outlier detection techniques follow a similar approach: Find spatial outliers and then compare them with temporal neighbors to verify whether they are ST-Outlier or not. Spatiotemporal methods for outlier detection [4, 5] are significantly more challenging because of the additional challenge of modeling the temporal and spatial components jointly [2].

In the unsupervised scenarios, previous examples of interesting anomalies are not available. In such cases, modelling the normal behavior in the presence of noise and anomaly pose extra difficulty. Generally, unsupervised methods can be used for either noise removal or anomaly detection, and supervised methods are designed for application-specific anomaly detection. Unsupervised methods are often used in an exploratory setting, where the discovered outliers are provided to the analyst for further examination of their application-specific importance [2].

1.1 Related Work

Outlier analysis is an important research area in data mining and machine learning communities. Outliers are also referred to as abnormalities, deviants, or anomalies in the data mining and statistics literature. It has been studied extensively in the context of time series data analysis. Time-series outlier detection studies find outliers considering only temporal context [2, 3, 12, and 13]. Whereas some other researches focus on finding outliers with respect to spatial context only [6, 7, 8, 9].

Birant and Kut [11] propose a neighborhood-based ST-Outlier detection mechanism. They propose a three-step approach to identify the spatio-temporal outliers. First, they use a modified version of DBSCAN algorithm to identify the spatial neighborhoods within the dataset. They define spatial outliers based on these neighborhoods. Then, they check the temporal context of spatial-outlier objects by comparing them to temporal neighbor objects.

Cheng and Li [5] propose a four-step approach to identify spatio-temporal outliers: classification (clustering), aggregation, comparison and verification. Their aim is to address the semantic and dynamic properties of geographic phenomena for ST-outlier detection. At the clustering step, the prior knowledge of the data is used to form some regions that have significant semantic meanings. The aggregation is also called filtering since the outliers (noises) will be filtered by changing the spatial scale. The main idea here is that if there are spatial outliers, they usually disappear if the scale of processing is reduced, clustering results will be different with different scales. With a decrease in scale, the difference between the objects and their spatial neighbors will decrease and the small regions that contain outliers will be removed. At comparison step, the results obtained at two spatial scales are compared in order to detect the potential spatial outliers. At the verification step, the outliers detected in the previous step can be suspected as ST-outliers. Therefore, the verification step checks the temporal neighbors of the suspected ST-outliers detected in the previous step. If the attribute value of such a ST-outlier is not significantly different from its temporal neighbors, this is not a ST-outlier. Otherwise, it is confirmed as a ST-outlier.

Adam et al. [10] propose a spatio-temporal outlier detection approach methodology based on Voronoi Diagrams. Their methodology is based on building micro and macro neighborhoods using the spatial and semantic relationships among the objects. They first build Voronoi diagrams using spatial properties of each object to find micro neighborhood. By using spatial and semantic relationships between objects, they find the Macro Neighborhood, which is an extended neighborhood, of each object. Using these neighborhoods, they detect outliers based on distance (Euclidean distance) values among various points. A data object is said to be a spatio-temporal outlier if it differs sufficiently from other points in the macro neighborhood. Here the Macro neighborhood consists of all the micro neighborhood merged into it under the spatial and the semantic relationship restrictions.

Gupta et al. [14] introduce the notion of context-aware anomaly detection in distributed systems by integrating the information from system logs and time series measurement data. In addition to temporal context, they use system specific performance metrics such as number of tasks running, memory usage, and CPU usage to create additional contextual data. They propose a two-stage clustering methodology to extract context and metric patterns using a PCA-based method and a modified K-Means algorithm. They instantiate their framework for Apache Hadoop platform. They extract additional context variables using the job history logs. They first cluster instances based on non-temporal context variables to extract context patterns, then apply time context using time-series metrics variables to detect outliers.

Aforementioned methods explained above have something in common: They first apply spatial (or non-temporal) context to find spatial outliers using a distance based technique. Then, spatial outliers are compared with other spatial objects using temporal neighborhoods to detect if they are temporal outliers too. They all use either spatial clustering base outlier detection algorithms such as DBSCAN [19], or locality based outlier detection algorithms such as LOF [20] to find neighborhoods. They can only detect simple anomalies like extreme cases, cannot detect collective anomalies. Another problem about distance-based methods are that they are well known to be computationally expensive and not suitable for multivariate datasets [2].

2 Proposed Model

The main inductive bias in this proposed model is the assumption that the same physics apply to all input sequences irrespective of which spatial neighborhood it comes from. The same model can be applied to any sequences which come from different geographical locations to extract useful representations to help find irregularities.

The proposed hybrid model composed of two main components: LSTM Encoder to extract temporal context and deep neural network classifier to learn the spatial context and detect anomalies. To learn the temporal representation that extracts all that is needed to predict the future sequence and reconstruct the input sequence at the same time, the combined framework idea proposed by [18] was employed. Their research was focused to unsupervised learning of video representations to predict the future frames. In this study, combined LSTM models were used to build encoder which can extract useful temporal context so that it can be used to build spatial classifier to extract spatial context and detect spatio-temporal anomalies. When the classifier is not successful in assigning correct spatial context label (location information) for the given sequence, we may assume that the sequence was generated by a process that do not comply with temporal and spatial regularities of the given world.

2.1 LSTM Encoder

The first step in the proposed spatio-temporal anomaly detection framework is to extract temporal context. The component responsible of doing this is Long Short Term Memory (LSTM) Encoder. It is similar to composite model proposed in [18]. It contains a LSTM Autoencoder and LSTM Future Predictor which trained in parallel to extract temporal context from dataset. There is one encoder, but two decoder LSTMs: one that decodes the representation generated by encoder into the input sequence, and another that decodes the same representation to predict the future multivariate time series.

By combining the two tasks (reconstructing the input and predicting the future) to create a composite model as shown in Fig. 1, a powerful LSTM Encoder component can be trained to extract temporal context. Here the reconstruction part is LSTM Autoencoder Model and predictor part is LSTM Future Predictor Model. As explained in [3], composite model tries to overcome the shortcomings that each model suffers on its own.

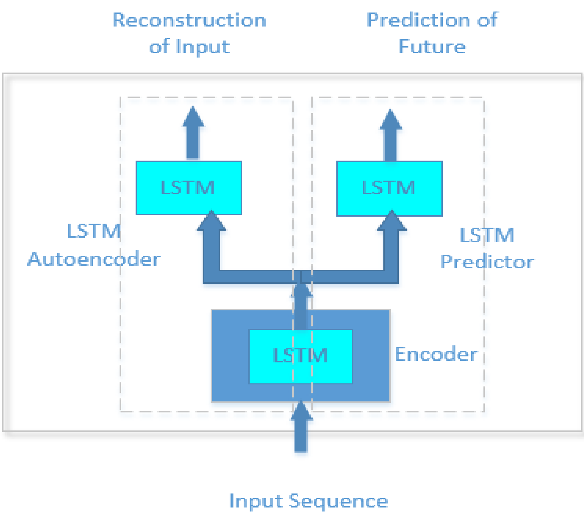

Fig. 1. Composite LSTM Encoder-Decoder Model.

2.2 Deep Neural Network Classifier

The second component of proposed framework is deep neural network (DNN) based classifier which is responsible for extracting spatial context and doing classification of input data to identify anomalous input sequences within given data. DNN based classifier and overall architecture of proposed hybrid framework is given in Fig. 2. Input sequences to this component first fed into the LSTM Encoder component built at first step. The output of the LSTM Encoder is the latent representation of temporal data. The classifier receives this latent representation as input and extracts spatial context from it. To be able to extract useful spatial context, the classifier was trained with the goal of predicting correct spatial location. If the unsupervised learning model built at first step, which is a LSTM Encoder, comes up with useful temporal representations then the classifier should be able to perform better.

The unsupervised anomaly detection problem is formulated as multiclass classification problem by training the classifier to learn regions which each input sequence comes from. The better spatial context extracted from data the better classification results have been achieved. The aim of this step is to build a final classifier which is successful in assigning the correct location label to each input sequence and at the same time would be able to detect spatio-temporal anomalies which do not confirm with overall trend within each spatio-temporal neighborhoods.

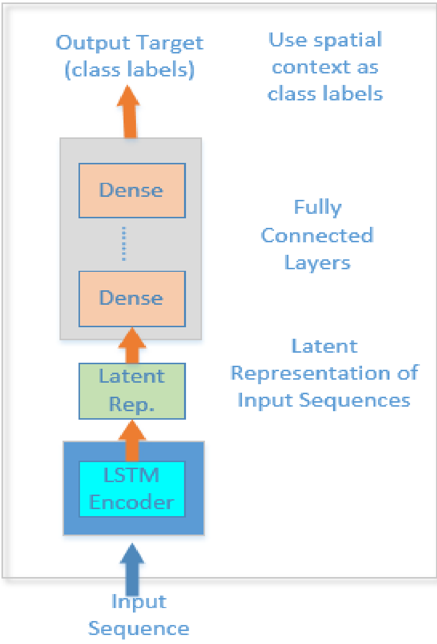

Fig. 2. Proposed Hybrid Framework with DNN Classifier

3 Case Study

To verify the proposed deep-learning based spatio-temporal outlier detection framework, we performed a case study on a real spatio-temporal dataset. Our goals in this case study are:

- To detect spatio-temporal outliers and analyze and study the results.
- Examine the spatio-temporal outliers detected by proposed approach, with base LSTM models (LSTM Autoencoder and LSTM Future Predictor) and LOF [20] algorithm.

3.1 Dataset Description

The dataset used in this study is the historical San Francisco Police Department Incident Report, which covers from January 1, 2003 to May 15, 2018. It is open and can be accessed through [16]. A different version of this data was made available through a Kaggle competition [17] on crime classification and has been used as sample dataset for various crime prediction and classification researches. The dataset contains 2215023 incident records, each consisting of date, time of day, one of 39 crime categories, a short description of the incident, the day of week, one of 10 police districts in which the incident occurred, the resolution of the incident, the address, the longitude, and the latitude fields. For this study, latitude, longitude and address information is ignored and only police district information is used as spatial context variable.

3.2 Data Preprocessing

In order to convert incident report data more convenient for time series analysis, the data is aggregated on daily basis based on crime categories. As a result, a multivariate dataset has been constructed covering 5613 days of data for each 39 crime categories. See Figure 3 for time series multivariate data example from one of the districts, for daily crime counts of 6 different crime categories taken from January 1, 2017 to December 31, 2017.

In this study, to decide the correct temporal window size for analysis, we looked at the weekly crime pattern for each district. See Figure 4 for weekly crime pattern for each district. The weekly crime count analysis shows a clear pattern for each district that one week time frame can be used for anomaly analysis as it shows strong predictability.

Figure 5 shows weekly crime pattern for district Northern for 6 years, from the beginning of 2005 until the end of 2010. It shows that each year has its own weekly pattern. Figure 6 shows this weekly crime count broken-down to top 12 occurring crime categories. For each district we can observe this clear weekly pattern for each type of crimes.

Data in this dataset is divided into standard weeks, 7 day time frames, which begin on a Sunday and end on a Saturday. Dataset which starts on January 1, 2003 to May 15, 2018 was divided into a total of 801 weeks. Spatio-temporal anomaly detection analysis were conducted on weekly data windows for each district.

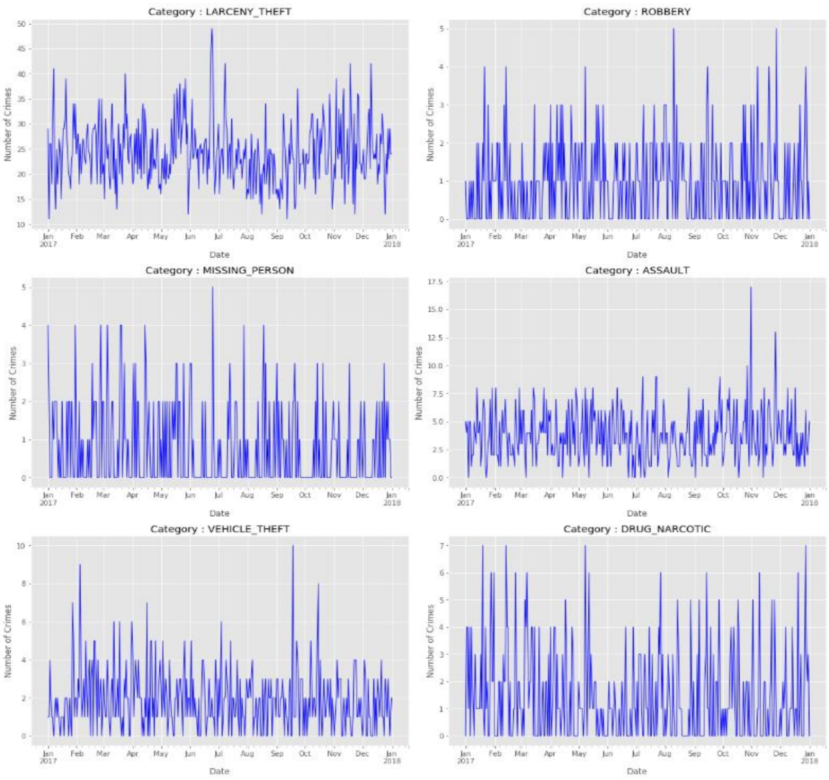

Fig. 3. Daily Crime counts for district Northern. 6 crime categories were shown as time series data example: Larceny/Theft, Robbery, Missing Person, Assault, Vehicle Theft, and Drug/Narcotic.

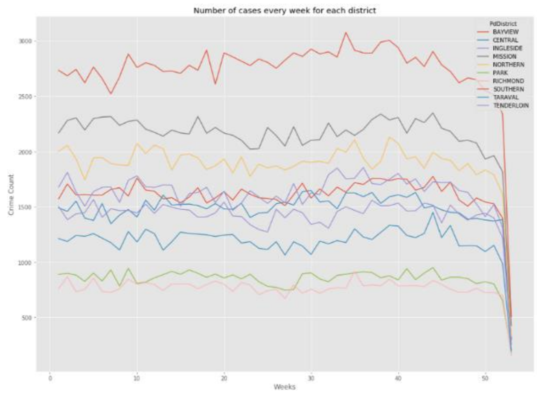

Fig. 4. Weekly crime pattern for each district from 2005 to 2011.

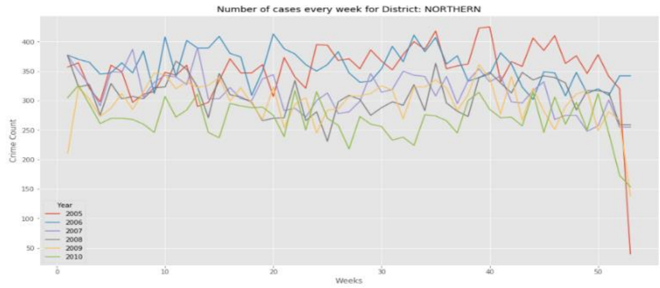

Fig. 5. Weekly crime pattern for district Northern from 2005 till by the end of 2010.

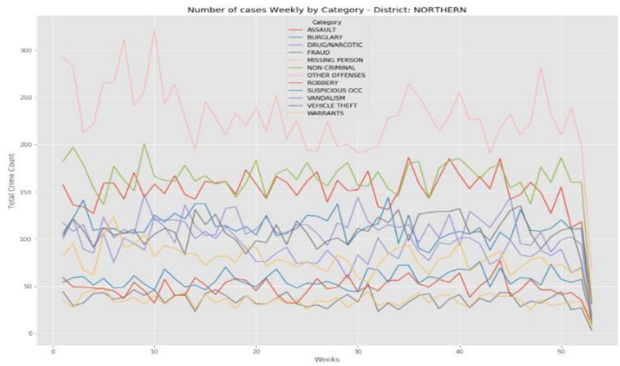

Fig. 6. Weekly crime pattern for district Northern broken down into the top 12 occurring crime categories.

4 Experimental Setup

Experiments were implemented in the Keras framework [21] using the Tensorflow [22] backend. All deep learning models were trained using backpropagation algorithm and Rectified Linear Unit (ReLU) activation function on all layers, except the output layer of the classifier which softmax function was used. ADAM optimizer algorithm was used to optimize the loss function, which was “mean squared error” for all LSTM based models and “categorical crossentropy” for multiclass classifier. For LSTM based future prediction model, two hidden layers with 200 and 100 hidden units each. For LSTM Autoencoder, one hidden layer with 100 hidden units gives the best result for this dataset. Deeper LSTMs did not improve results significantly. For all LSTM models 10 epochs with batch size of 64 were used. First 14 years (730 standard weeks starting from January 5th, 2003 till December 31st, 2016) of dataset were used for training of all models, rest of the data (71 standard weeks starting from January 1st, 2017 till May 12, 2018) were used as test set.

4.1 Base LSTM models

LSTM Autoencoder models are used to detect multivariate anomalies based on reconstruction errors. They try to build the input sequence using small dimensional latent representation, and if the result varies greatly from the original input then the input can be labelled as anomaly. The size of the input sequence, which is the input window, is 7, which is equivalent to the size of standard week. The size of the output sequence, which is prediction window, is equal to the input window as it tries to reconstruct the input sequence.

LSTM Future Predictor models are used to detect multivariate anomalies based on prediction errors. They try to predict future sequence given the input sequence, and if the prediction varies greatly from the real data, then the input sequence can be labelled as anomaly. The size of prediction window and input window was kept same throughout the experiment as 7, which is equivalent the number of days in a standard week. LSTM Future Predictor model tries to predict the next week's number of crimes for each 7 days and for each 39 different crime categories for each district given the current week's data.

To quantitatively measure the prediction and reconstruction performance of base LSTM models, the root mean squared error (RMSE) is calculated for each 39 variable for 7 time-steps as follows:

$$RMSE = \sqrt{\frac{1}{N*T}\sum_{n,t}\left(y_{n,t} - \hat{y}_{n,t}\right)^2} \qquad (1)$$

where N is the total number of features (crime categories), T is the number of time steps (size of input window) considered for this problem, $y_{n,t}$ ve $\hat{y}_{n,t}$ is the exact crime count and the predicted (or reconstructed based on the type of LSTM decoder used) crime for given time step and crime category. The RMSE for all input sequences (test weeks) were calculated and interquartile range for error values was defined to select threshold for anomaly detection. Test weeks whose errors fall outside of 1.5 times of the interquartile range above the 3rd quartile were flagged as anomaly. Table 1 shows number of anomalous weeks and their indexes detected by LSTM models.

Table 1. Anomalous weeks detected by base LSTM models.

Model Name	District - # Anomalous Weeks	Anomalous Week Indexes
LSTM Future Predictor	Mission – 1	1
	Tenderloin – 1	13
	Northern – 1	5
	Richmond – 2	27, 37
	Bayview – 2	30, 55
	Central – 3	5, 15, 63
	Park – 1	1
	Taraval – 0	
	Southern – 0	

	Ingleside – 0	
LSTM Autoencoder	Mission – 0	
	Tenderloin – 0	
	Northern – 2	5, 40
	Richmond – 2	27, 37
	Bayview – 1	55
	Central – 2	5, 15
	Park – 2	1, 36
	Taraval – 2	9, 34
	Southern – 1	11
	Ingleside – 1	48

Out of 71 test weeks, LSTM Future Predictor model flags 11 weeks as anomalous, LSTM Autoencoder flags 13 weeks as anomalous. For some districts (Richmond, Bayview, Central and Park), weeks flagged as anomalous are matching.

4.2 Spatio-Temporal Classifier

The proposed deep learning based spatio-temporal anomaly detector is composed of a LSTM Encoder component and a fully connected deep neural network classifier as depicted in Figure 2. LSTM Encoder component was trained in a composite framework where an LSTM Autoencoder and an LSTM Future Predictor were trained in parallel with a common encoder component. After the encoder has been trained, it was put in front of the DNN classifier as a temporal context extractor for the input data. The classifier is designed to predict the location label of each 7-day long multivariate input sequence in a supervised training settings. Each 10 different district constitutes 10 different labels, and the anomaly detection problem converts to multi-class classification problem. The number of inputs is equal to the dimension of LSTM Encoder. The first hidden layer has 200 neurons, and second hidden layer has 50 neurons. ReLU activation function was used for hidden layers. The final layer of the deep neural network classifier is the softmax classifier using "categorical crossentropy" as loss function with neuron numbers equals to number of spatial labels. Cross entropy loss function formula can be given as following:

$$MCCE = -\frac{1}{N}\sum_{i}^{N}\sum_{l}^{C}[y_{il}\log(\hat{y}_{il})] \qquad (2)$$

where C is the number of class labels and N is the number of test sequences (test weeks for the given dataset).

Although the second component of the hybrid framework is a classifier, the ultimate purpose is to detect spatio-temporal anomalies. We train the classifier using district information of input sequences as labels to enforce the deep neural network classifier to learn spatial context. If the classifier can learn useful representation of spatial context, it gets higher accuracy on classification problem. The input sequences which are classified wrongly would be flagged as potential spatio-temporal anomalous sequences.

Metrics. Precision and recall were used to measure the accuracy of the classifier. Precision is the number of true positive results divided by the number of all positive results (true positives + false positives), whereas recall is the number of true positive results divided by the number of actual positive results (true positives + false negatives). Accuracy is the total number of true positive and true negative cases divided by all number of cases. The classifier gave the total accuracy of 77.18%. Table 2 shows the performance of the classifier for each class label.

Table 2. Precision and recall results for each class labels.

Class Labels	Precision	Recall
Taraval	76.36%	59.15%
Mission	77.90%	94.36%
Bayview	70.77%	64.79%
Ingleside	100%	66.20%
Central	64.77%	80.28%
Northern	75%	46.48%
Southern	64.42%	94.37%
Park	97%	91.55%
Tenderloin	63.86%	74.65%
Richmond	100%	100%

As a sample case, district Taraval was investigated for all 39 features. The week numbers 59 and 61 from test weeks (which are 71 in total) were misclassified by spatio-temporal classifier and not detected by LSTM models as anomaly. We would flag those weeks as spatio-temporal anomalies. The Fig.7 shows crime counts for selected crime types occurred on week 59 and 61 for district Taraval. The crime values corresponding to weeks 59 and 61 were colored red.

Dimension reduction using LSTM Autoencoder was employed to input data to graphically visualize the detected spatio-temporal anomalies. Data from districts Central, Richmond, and Southern were projected into 3-dimensional space along with detected spatio-temporal anomalies. The Fig. 9 shows spatio-temporal anomalies detected by proposed framework visualized in 3D.

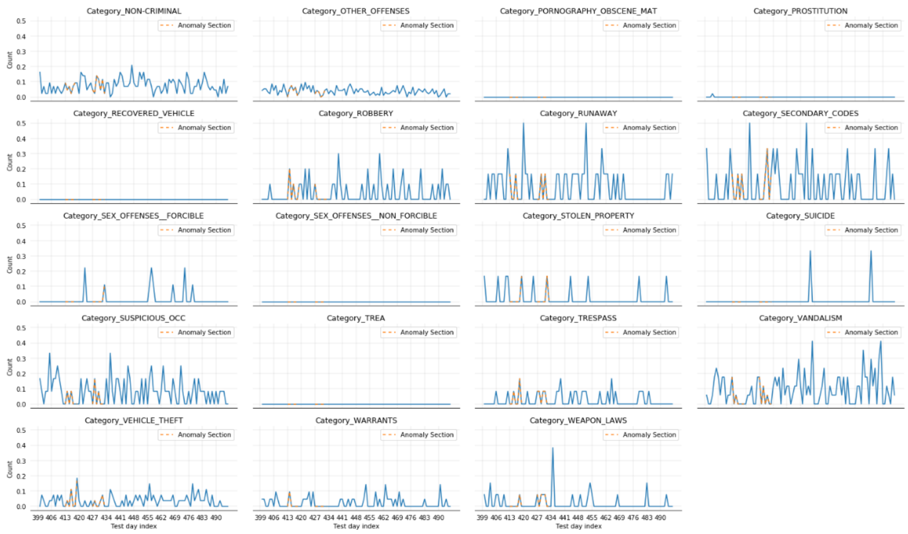

Fig. 7. Crime count analysis for different crime categories for test weeks 59 and 61 which were detected as spatio-temporal anomaly by the hybrid model for district Taraval.

4.3 Comparison of Proposed Model and LOF

Local Outlier Factor (LOF) [20], is one of the most popular algorithms that quantifies the outlierness of an object. To evaluate the effectiveness of the proposed framework, detected anomalies by LOF and proposed framework were projected in lower dimensional space and visualized. Fig. 8 shows the visualization of data and outliers. A more distinctive decision boundary for outliers can be seen compared to not so obvious boundary of LOF algorithm.

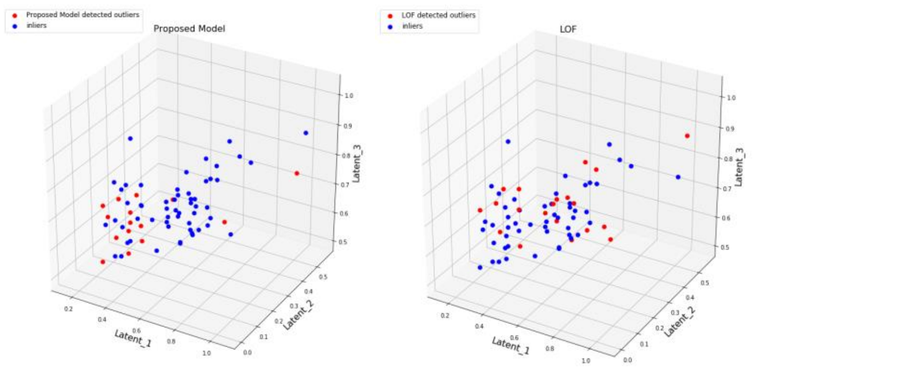

Fig. 8. 3D graphic visualization of multivariate time series data and detected spatio temporal anomalies by proposed framework for districts Central, Richmond, and Southern.

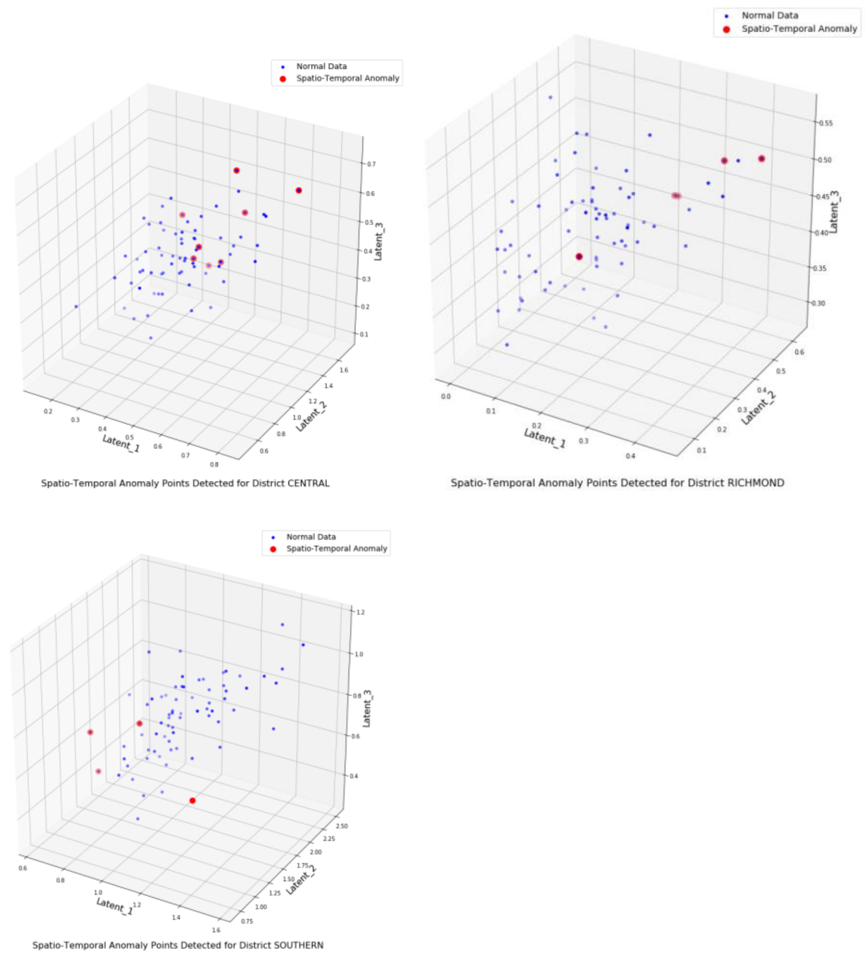

Fig. 9. 3D graphic visualization of multivariate time series data and detected spatio-temporal anomalies by proposed framework for districts Central, Richmond, and Southern.

5 Conclusion

In this study a hybrid framework was proposed to detect spatio-temporal anomalies of multivariate datasets in an unsupervised way. In this unsupervised settings, no labeled dataset is available to train modules for anomalous cases. The first component is LSTM Encoder which was trained to extract temporal context from input sequences. The second component of the framework is deep neural network based classifier to extract spatial context from encoded data. The framework learn temporal and spatial contexts separately and uses those representations to identify spatio-temporal anomalies. If the classifier classify the input sequence based on spatial labels, then the input can be considered as anomaly. The hybrid model was able to persistently detect spatio-temporal anomaly sequences well beyond the LSTM based prediction models and LOF algorithm. To further get improvements on spatial classifier, the model can be extended by applying convolutional neural network base spatial context extractor using finer grained neighborhood data.

References

1. V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey", ACM Computing Surveys, Vol. 41(3), Article 15, July 2009.

2. C. C. Aggarwal, Outlier Analysis, Springer Publishing Company, 2017.

3. M. Gupta, J. Gao, C. C. Aggarwal and J. Han, "Outlier Detection for Temporal Data: A Survey", IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 9, pp. 2250-2267, Sept. 2014.

4. T. Cheng and Z. Li. A Hybrid Approach to Detect Spatial-temporal Outliers. International Conference on Geoinformatics, 2004.

5. T. Cheng and Z. Li. A Multiscale Approach for Spatio-temporal Outlier Detection. Transactions in GIS, 10(2), pp. 253–263, March 2006.

6. S. Shekhar, C. T. Lu, and P. Zhang, “A Unified Approach to Detecting Spatial Outliers”, Geoinformatica, 7(2), pp. 139–166, 2003.

7. C.-T. Lu, D. Chen, and Y. Kou, “Algorithms for Spatial Outlier Detection”, ICDM Conference, 2003.

8. S. Shekhar, C. T. Lu, and P. Zhang, “Detecting Graph-based Spatial Outliers: Algorithms and Applications”, ACM KDD Conference, 2001.

9. Y. Kou, C. T. Lu, and D. Chen, “Spatial Weighted Outlier Detection”, SIAM Conference on Data Mining, 2006.

10. N. R. Adam, V. P. Janeja, and V. Atluri, “Neighborhood-based Detection of Anomalies in High-Dimensional Spatio-temporal Sensor Datasets”, ACM SAC Conference, 2004.
11. D. Birant and A. Kut, "Spatio-temporal outlier detection in large databases," 28th International Conference on Information Technology Interfaces, 2006, Cavtat/Dubrovnik.

12. K. Yaminshi and J. Takeuchi. A Unifying Framework for Detecting Outliers and Change Points from Time Series Non-Stationary Data. ACM KDD Conference, 2002.

13. Cheng, Haibin et al. “Detection and Characterization of Anomalies in Multivariate Time Series.” SDM (2009).

14. Gupta, M., Sharma, A.B., Chen, H., & Jiang, G. (2013). Context-Aware Time Series Anomaly Detection for Complex Systems.

15. K. Smets, B. Verdonk, E. M. Jordaan, "Discovering Novelty in Spatio/Temporal Data Using One-Class Support Vector Machines", International Joint Conference on Neural Networks, 2009.

16. Police Department Incident Reports: Historical 2003 to May 2018, https://data.sfgov.org/Public-Safety/Police-Department-Incident-Reports-Historical-2003/tmnf-yvry, last accessed 2019/06/19.

17. San Francisco Crime Classification Data, Kaggle, https://www.kaggle.com/c/sf-crime/data.

18. Srivastava, N., Mansimov, E., and Salakhutdinov, R.: Unsupervised learning of video representations using lstms. International Conference on Machine Learning (ICML), 2015.

19. Ester, M., Kriegel, H. P., Sander, J., and Xu, X.: "A density-based algorithm for discovering clusters in large spatial databases with noise," in KDD'96 Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 1996, pp. 226-231.

20. Breunig, M. M., Kriegel, H. P., Ng, R. T., and Sander, J.: "LOF: identifying density-based local outliers," in Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, 2000, pp. 93-104.

21. Keras Homepage, https://keras.io/.

22. Tensorflow Homepage, https://www.tensorflow.org/.