Deep learning based Urban Anomaly Prediction from Spatiotemporal Data

Bhumika[0000−0002−4391−8634] and Debasis Das[0000−0001−6205−4096]

Indian Institue of Technology Jodhpur, India
{bhumika.1,debasis}@iitj.ac.in

Abstract. Urban anomalies are unusual occurrences like congestion, crowd gathering, road accidents, natural disasters, crime, etc., that cause disturbance in society and, in worst cases, may cause loss to property or life. Prediction of these anomalies at the early stages may prevent significant loss and help the government to maintain urban sustainability. However, predicting different kinds of urban anomaly is difficult because of its dynamic nature (e.g., holiday versus weekday, office versus shopping mall) and presence in various forms (e.g., road congestion may be caused by blocked driveway or accident). This work proposes a novel integrated framework *UrbanAnom* that utilizes a data fusion approach to predict urban anomaly data using gated graph convolution and recurrent unit. To evaluate our urban anomaly prediction framework, we utilize multi-stream datasets of New York City's urban anomalies, points of interest (POI), roads, calendar, and weather that were collected via smart devices in the city. The extensive experiments show that our proposed framework outperforms baseline and state-of-the-art models.

Keywords: Urban anomaly, deep learning, data fusion, spatio-temporal data, gated graph convolution network, gated recurrent unit

1 Introduction

The term *anomaly* refers to a deviation from the normal or expected pattern, and examples of anomalies include fraud, real-world events, criminal activity, traffic congestion, crowding, etc. One such type of anomaly is the *urban anomaly*, which we see around us in the form of traffic congestion, fairs, market promotions, fire incidents, criminality, etc., and may pose hazards to the general public's safety or result in financial losses. Statistics show that the annual cost of traffic congestion in four major Indian cities—Delhi, Mumbai, Bengaluru, and Kolkata is Rs. 1.47 lakh crore [4]. Therefore, reducing life or economic losses might be possible with early and accurate urban anomaly prediction. The local government, for instance, can organize transportation and mobility management during the festival season to avoid an unneeded stampede. With the help of this study, we hope to promote sustainable urbanization by foreseeing various types of urban anomalies.

Predicting urban anomalies traditionally involves a lot of effort. For instance, feature-based techniques rely on extracted features [20], which necessitate domain expertise to accurately capture the intricate dynamics of the urban

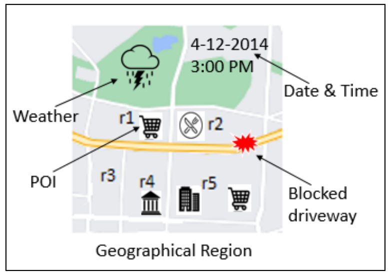


Fig. 1: Blocked driveway anomaly in urban areas

anomaly. While other research [9, 10] attempts to forecast anomalies but are only capable of coarse-grained prediction with low accuracy, we are aiming to predict different types of anomaly in a region bordered by roadways. An example of anomalies in urban areas is shown in Fig. 1, where blocked driveway anomaly affects region $r2$ and $r5$ directly and inferred from different contextual information like weather, point of interest (POI), and date & time. Difficulties in modelling anomaly prediction from several viewpoints are dynamic nature, rare occurrences, area dependency, and direct-indirect influencing factors; these difficulties drive a novel framework design.

We present the integrated framework where multiple deep neural networks capture different aspects of data and fuse their output to achieve common objective. To address the challenge of the dynamic nature of the urban anomaly, we extract spatial and temporal insights. Next, to solve the challenge of region dependency, we form regions with the help of road network with the intuition that it may pay attention towards illegal parking and blocked driveway. To incorporate the influence factors of urban anomaly, we use weather-related features. Spatial and temporal aspects are learned by the framework separately. Later, to join these modules, we fuse the output as input to global attention layer. To get the relevance of different anomalous events, the attention layer is used to predict future events more accurately. Finally, the hidden states from the attention layer are passed as input to multi-layer perceptron to predict the anomaly category in a region. We utilize a number of real-world datasets, including those gathered from New York City's 311 complaints, POIs, and weather stations. Our contributions are summarized as below:

- A novel framework, namely UrbanAnom is proposed to predict urban anomaly of specific categories in particular region.
- We propose a GatedGCN based method to capture inter-region relationships in the city. A Stacked GRU based modeling approach is chosen to take advantage of long-term and short-term temporal dependency.
- The extensive experiments on real-world urban anomaly dataset shows that UrbanAnom outperforms in terms of different metrics like F-measure, macro-F1, and micro-F1 of 83%, 85%, and 83%, respectively, from baseline as well as state-of-the-art models.

2 Related Work

A review of the literature from spatio-temporal perspective is presented in this section with two aims in mind: (1) Deep learning based methods, and (2) Hybrid learning (graph + deep learning) based methods.

2.1 Deep learning based methods

Recently, deep learning based algorithms have been utilised with promising results in a variety of anomaly detection and prediction tasks, including crowd gathering, traffic accidents, criminal prediction, etc. Among deep learning techniques, recurrent neural networks have demonstrated superior performance in a variety of spatial-temporal tasks, including weather forecasting, stock market forecasting, accident forecasting, etc. In particular, Jiang et al. [12] predicted crowd dynamics from video data. They made predictions about future crowd density and flow using a multi-task convLSTM encoder-decoder. Another work done by Zhou et al. [29] have suggested utilising deep learning to predict crimes like robberies and burglary by combining spatial, temporal, and semantic data into latent space. Huang et al. [11] suggested a multi-view multi-model spatial-temporal learning (MiST) framework used a recurrent neural network and pattern fusion module to forecast city-wide anomalous events. For predicting traffic accidents, they suggested a dynamic fused network framework that makes advantage of hierarchical deep learning. Additionally, Shimosaka et al. [18] proposed mixed-order poisson regression from GPS data to find nationwide abnormal events. In order to predict urban anomalies, Huang et al. [10] created a hierarchical deep neural network that combined geographical, temporal, and category aspects. We also use the concept of integrating different deep learning models since an integrated framework can better capture the dynamic behavior of events. The performance of deep learning-based approaches is usually improved by the use of attention mechanisms in current trends. We also employ this concept to enhance the performance of the entire framework.

2.2 Hybrid learning (graph + deep learning) based methods.

In general, several factors influence urban anomalies, and data analysis using a single dimension does not reveal any underlying correlations. Therefore, Zhang et al. [26] suggested employing graph embedding and neural network to detect anomaly from spatio-temporal data. A multi-modal fusion model for urban anomaly prediction from a spatial and temporal perspective was put out by Liu et al. [16]. They obtained spatial information using a graph convolution network, and temporal features using a gated recurrent unit. On the other hand, we change the general architecture and take into account extra contextual information like the calendar and weather data. Zhao et al. [27] solution to the traffic prediction problem, which incorporates both spatial and temporal relationships, used a temporal graph convolution network. In a different article, authors Liu et al. [15] also suggested a system that used adaptive graph convolution and temporal convolution to solve the challenge of urban anomaly detection. To efficiently capture various inherent information, the integrated architecture has been applied to numerous prediction tasks. Urban anomalies can be predicted using spatial and temporal clues, but we also need to take the context into account. Therefore, we employ an integrated framework to capture different aspects.

3 Preliminaries

In this section, we first define the terms and then formally define the problem statement for urban anomaly prediction. Particularly, we consider R geographical regions of an urban area and A anomaly categories with T time window.

3.1 Notation

Definition 1. ***Region Graph*** *In this study, we use the map segmentation method [24] on the road network, such as highway and arterial roads, to split the city into regions* $R = r_1, r_2, \ldots, r_n$. *In the proposed architecture, we take inter-region graph formulation into account. Each region* $r_i \in R$ *functions as a node* $v \in V$ *of the graph* $G = (V, E)$ *in the inter-region case, where* V *indicates the set of all disjoint regions and* E *is the set of all connecting pathways of the regions. If two regions are close to one another, an edge* $e_k \in E$ *exists between* v_i *and* v_j *such that* $i \neq j$ *where* $(u, v) \in V$. *The region graph's adjacency matrix can be defined as* $\mathcal{RG} \in \mathbb{R}^{V*V}$. *In* $\mathcal{RG}$, *we specifically set the element* $\mathcal{RG}_{ij} = 1$ *if a connecting path exists between two regions and* $\mathcal{RG}_{ij} = 0$ *if there isn't a connecting path.*

Definition 2. ***Point of interest*** *The point of interest (POI) dataset includes latitude and longitude positions for hospitals, businesses, educational institutions, retail locations, etc., that serve as a feature of graph nodes. The rationale behind taking POI into account is to identify correlations between various places depending on how they function (such as a hospital or commercial area). For instance, a similar functioning region in an urban location would have a similar anomalous pattern, according to Yuan et al. [23] study. If* F *is the number of POI categories, then the adjacency matrix for POI can be written as* $\mathcal{PI} \in \mathbb{R}^{V*F}$. *The element* $\mathcal{PI}_{ij}$ *is set to* 1 *in* $\mathcal{PI}$ *if a specific POI category is present in a region and* $\mathcal{PI}_{ij}$ *to 0 in all other cases.*

Definition 3. ***Temporal Anomaly Stream*** *Data in the prediction of urban anomalies shows a temporal stream that varies over time. This temporal stream is represented for a region* r_i *at time step* k *as* $\mathcal{TS} = (Y_i^{1k}, Y_i^{2k}, \ldots, Y_i^{lk})$, *where* $\mathcal{TS} \in \mathbb{R}^{n*l*k}$ *is the record of an anomaly of* l *category in* k *time slots at* r_i *region. When an anomaly of category* a_l *occurs at the* k^{th} *time step at region* r_i *the adjacency matrix for the temporal stream has the value* $\mathcal{TS}_i^{lk} = 1$ *and* $\mathcal{TS}_i^{lk} = 0$ *otherwise.*

Definition 4. ***Weather and Calendar Context*** *It stands to reason that weather has an impact on anomalous occurrence because obstructed driveway reports are more often in adverse weather. As a result, we incorporate the weather as a crucial component, which is denoted as* $\mathcal{W} \in \mathbb{R}^{1\times f_w}$*. The urban anomaly changes with time as well. For instance, because most individuals tended to sleep at night, there are fewer complaints at night. Additionally, there is a different pattern of complaints during the week and on vacations. We therefore divide the given day, which consists of 24 hours, into six-hour periods and represent this as a one-hot encoding vector,* $\mathcal{CL} \in \mathbb{R}^{1\times f_{cl}}$*.*

3.2 Problem Statement

Solutions for predicting urban anomalies typically focus on extracting spatio-temporal data without taking context into consideration. On the other hand, we take into account semantic, spatial, and temporal data. The goal is to learn a predictive function that predicts l anomaly categories across n regions in s future time steps given historical data with l anomaly categories $A = (a_1, a_2, \ldots, a_l)$ over n region $R = (r_1, r_2, \ldots, r_n)$ and k time step $T = (t_1, t_2, \ldots, t_k)$. The formal representation of problem is given as:

$$y_n^{l,(k+s)} = \Psi(\mathcal{RG}, \mathcal{PI}, \mathcal{TS}, \mathcal{W}, \mathcal{CL}); \tag{1}$$

where $\Psi(\cdot)$ is a approximation function that we want to learn with input arguments region graph ($\mathcal{RG}$), point of interests ($\mathcal{PI}$), temporal anomaly ($\mathcal{TS}$), weather information ($\mathcal{W}$), and calendar data ($\mathcal{CL}$). The outcome is $y_n^{l,(k+s)}$, which is a prediction of all anomaly categories l in every region n over the next s time steps.

4 Framework: UrbanAnom

In this section, UrbanAnom framework is described in detail with introduction of the model input and the motivation for proposed framework. As Fig. 2 shows the architecture of UrbanAnom that consist four major modules: Semantic Spatial Module, Context Aware Temporal Module, Global Attention, and Multi-Layer Perceptron.

Definition 5. ***Anomaly context tensor*** *The input for the model is adjacency region matrix ($\mathcal{RG}$), point of interest matrix ($\mathcal{PI}$), temporal anomaly stream ($\mathcal{TS}$), calendar ($\mathcal{CL}$), and weather embedding ($\mathcal{W}$). As shown in Fig. 2, context aware temporal module have extracted* $\mathcal{RG}$*,* $\mathcal{PI}$ *and* $\mathcal{CL}$*,* $\mathcal{W}$ *data along with* $\mathcal{TS}$*. In case of semantic spatial dependency adjacency matrix* $\mathcal{RG}$ *and* $\mathcal{PI}$ *are fed into GatedGCN.*

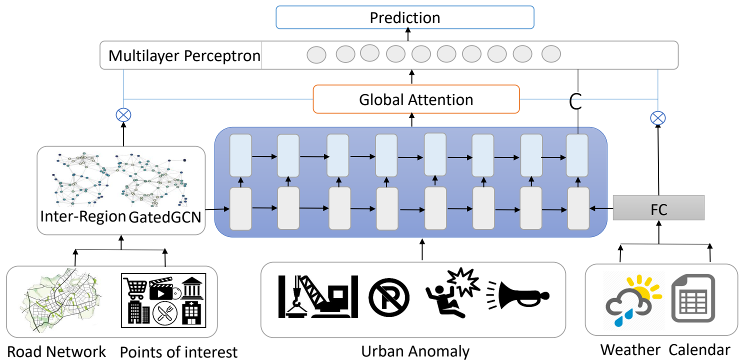


Fig. 2: Graphical representation of proposed architecture

4.1 Semantic Spatial (SS) Module

To consider spatial information from the geographical region along with semantic signals, we also include POI details. The intuition behind including POI is that similar functioning regions may have similar kinds of anomalies. Traditionally, convolution neural network (CNN) is used to capture the local spatial information, but it does not work well with non-euclidean space such as graphs. We divide the city into regions according to road network, and considering each region as node of the graph, CNN is unable to capture complex topological relationships from the graph network. Recently, CNN variant that works over graph has been come into existence, such as graph convolution network (GCN). A benchmark over graph neural network given by [5], shows gatedGCN works better in the node classification, graph classification, and link prediction, which is anisotropic variant of GCN. Therefore, we decide to use gatedGCN [1] in the task of extracting spatial information of geographical region.

Let h_i^l is a hidden unit at layer l attached with node i. The updated unit h_i^{l+1} at next layer $l+1$ in GatedGCN uses bath normalization, edge gates and residual connections which is represented by the equations:

$$h_i^{l+1} = h_i^l + ReLU(BN(U^l h_i^l + \sum_{j \in N_i} e_{ij}^l \odot V_l h_j^l)), \tag{2}$$

where $\odot$ is a Hadamard product, $U^l, V^l \in \mathbb{R}^{d*d}$ and edge gates e_{ij}^l is used as soft attention represented in equation as:

$$e_{ij}^l = \frac{\sigma(\hat{e}_{ij}^l)}{\sum_{j' \in N_i} \sigma(\hat{e}_{ij'}^l) + \epsilon}, \tag{3}$$

$$\hat{e}_{ij}^l = \hat{e}_{ij}^{l-1} + (BN(A^l h_i^{l-1} + B^l h_j^{l-1} + C^l \hat{e}_{ij}^{l-1})), \tag{4}$$

where ϵ is hyperparameter for numeric stability, σ represent activation function, $A^l, B^l, C^l \in \mathbb{R}^{d*d}$. The difference between GCN and gatedGCN is that the later one take cares edge feature at each layer. In summary, we use GatedGCN [1] to learn semantic spatial relationship of anomalies.

4.2 Context Aware Temporal (CAT) Module

In the context-aware temporal module of UrbanAnom, we aim to encode sequential anomaly patterns with the context of anomaly occurrence in previous days or weeks. We fed the details of anomalies that occur a month ago, a week ago, a day ago. For each region r_i, we generate anomaly occurrence vector A_l^t that reflect the anomalies in region r_i. Given the generated anomaly vector, we leverage GRU [3], which is one of the recurrent neural network (RNN) that work with time-series data. RNN is widely applied in time-series data. Different RNN variants are there with various recurrent units such as vanilla RNN, Long short term memory (LSTM), and gated recurrent unit (GRU). Both LSTM and GRU have gating mechanism to deal with the vanishing gradient problem of the traditional recurrent neural network, but GRU is less complex and more efficient in small training data. Our framework is flexible to change recurrent units; the effect of change is also explored.

In GRU, on each timestep t, we have input x^t, i.e., features and hidden state h^t. The hidden state also acts as a memory block, and operations of the memory block are controlled by two gates, namely the update gate and reset gate. Update gate controls what part of the hidden state is updated versus preserved, and reset gate controls what part of the previous state are used to compute new content. The operation on memory block performed by gates using following equations at each timestep:

$$u^t = \sigma(W_u h^{t-1} + U_u x^t + b_u) \tag{5}$$

Update gate (u^t) controls what part of hidden state are updated versus preserved.

$$r^t = \sigma(W_r h^{t-1} + U_r x^t + b_r) \tag{6}$$

Reset gate (r^t) controls what part of previous state are used to compute new content.

$$h'^t = tanh(W_h(r^t \cdot h^{t-1}) + U_h x^t + b_h) \tag{7}$$

(h'^t) represent the next hidden state.

$$h^t = (1 - u^t) \cdot h^{t-1} + u^t \cdot h'^t \tag{8}$$

(h^t) is the current hidden state. Here W_u, W_r, W_h are weights and b_u, b_r, b_h are biases. While σ and tanh are activation functions, Dot (.) represent element wise product.

4.3 Global Attention Module

Limitation of neural network based architectures is that they represent fixed length internal representation, which is not good for representing long dependencies. In our case, for a specific region r_i, complex dependencies exist among spatial and temporal anomaly occurrences. In our UrbanAnom architecture, we use attention mechanisms that pick the most essential signals to capture short and long distance dependencies, in order to avoid the situation where only the last hidden vector is used to represent spatial and temporal patterns [17]. The ability of the attention mechanism to selectively focus on a portion of crucial information has been demonstrated in machine translation and image analysis tasks. This inspires us to prefer global attention above representations of hidden spatial and temporal states. Attention mechanism is given by the equations as:

$$attn_n = tanh(W_{attn} h_n + b_{attn}) \tag{9}$$

$$\Lambda_n = \frac{exp(attn_n^T W_m)}{\sum_{n'} exp(attn_n^T W_m)} \tag{10}$$

$$\hat{a} = \sum_{n=1}^{N} \Lambda_n W_{attn} h_n \tag{11}$$

where W_{attn}, b_{attn}, W_m are training parameters, h_n shows the hidden state learnt from lower layer and the number of input vectors represented by N. Learned importance weight represented by α_n and $\hat{a}$ represent new hidden representation called attention vector. In our case we utilize the attention over hidden states of both semantic spatial module and context aware temporal module.

$$\Upsilon = \Delta(\hat{a}(SS_{h^t}), \hat{a}(CAT_{h^t})) \tag{12}$$

where Υ represents the global attention, SS_{h^t} and CAT_{h^t} are the hidden states of semantic spatial module and context aware temporal module. The symbol Δ is the fusion function between SS and CAT attention output.

4.4 Multi-Layer Perceptron based Prediction module

The multi-Layer perceptron is used as a last layer in the prediction phase of the proposed UrbanAnom architecture to generate the presence of anomalies in various categories of each unique region r_i. The MLP is able to describe anomaly occurrence probabilities using a softmax function. The output of SS, CAT, and Υ is dynamically fused into a multilayer perceptron network to generate the final anomaly prediction represented as an equation below:

$$y_n^{l,k} = fc(W_{ss} * SS + W_{CAT} * CAT + W_{at} * \Upsilon_t) \tag{13}$$

where W_{ss}, W_{CAT}, W_{at} are learnable parameters and $fc(\cdot)$ represent the fully connected layer of perceptrons. The cross entropy loss function is defined as:

$$L = \sum_{n,l,k \in A} y_n^{l,k} log \hat{y}_n^{l,k} + (1 - y_n^{l,k}) log(1 - \hat{y}_n^{l,k}) + \lambda R_{reg} \tag{14}$$

where $\hat{y}_n^{l,k}$ denotes the predicted anomalous event of the l category in region r_i in k^{th} time slot. We use L_2 norm as regularization R_{reg} function and λ is adjustable hyperparameter. The model parameters are learned during minimization of loss function.

5 Evaluation

We conduct experiments to determine the efficiency of the proposed framework using datasets from New York City, including NYC-Urban Anomaly, NYC-POI, NYC-Road Network, NYC-Weather, and NYC-calendar. In this section, we provide a description datasets used, parameter settings, performance validation, parameter sensitivity, and evaluation of variants.

5.1 Dataset

In developed nations, the anomaly reporting system also emerged along with the rise in urbanisation. As a result, we run independent experiments to predict anomalies of different categories while validating our proposed framework using various real-world datasets from New York City [1]. The city of New York has a 311 emergency service platform [2] that lets residents file complaints by phone call, text message, or mobile app. Traffic congestion, crime, fire events, and other anomalies can occur in metropolitan areas, but we have chosen the data given by the 311 emergency service in New York City as our dataset for anomalies.

Brief explanation of the datasets is given as: 1) NYC-Urban Anomaly: Dataset contains latitude, longitude, complaint type, and timestamp information. Four types of anomalies, including blocked driveways, noise, illegal parking, and building use, have been the subjects of our experiments. The reason for selecting only these categories is because they are common occurrences and simple to compare with prior research.The distribution of urban anomalies is depicted in Fig. 3, with darker colours denoting more anomalies in a given area. 2) NYC-POI: The dataset containts geo-coordinated information on different categories are grouped into six main categories, including education, food & dinning, health & beauty etc. The POI data is extracted from OpenStreetMap API [3]of year 2017 and assume there is no major change in POI information. Table 1 shows the statistics of the dataset. 3) NYC-Road Network: The main component of the transportation system is the road network. To segment the road network dataset of New York City into regions, a map segmentation method. The road network information is given in the the website [4]. 4) NYC-Weather: We acquired meteorological information for New York City from WunderGround[5], which included temperature information as well as 18 characteristics of various weather conditions, such as sunny, rain, and haze, etc. 5) NYC-Calender: The calendar information, including the days of the week, the weekend, and the holidays, was retrieved from the Holiday library [6].

---

[1] https://opendata.cityofnewyork.us/
[2] https://portal.311.nyc.gov/
[3] https://www.openstreetmap.org/
[4] https://figshare.com/articles/dataset/Urban_Road_Network_Data/2061897
[5] https://www.wunderground.com/
[6] https://pypi.org/project/holidays/

Table 1: Dataset statistics of urban anomaly and POI from NYC

Urban Anomaly from NYC	
Category	**Instances**
Blocked Driveway	74,698
Noise	134,690
Illegal Parking	57,374
Building Use	24,319

Point-of-interest (POI) from NYC	
Category	**Instances**
Business to Business	3717
Education	1062
Government & Community	3116
Food & Dinning	3385
Health & Beauty	4336
Real Estate & Construction	4675

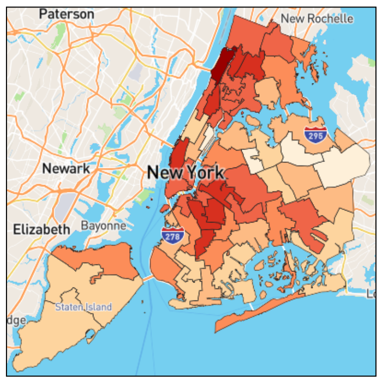


Fig. 3: Anomaly distribution in NYC

5.2 Parameter Settings

All experiments were conducted on Google Colab using a GPU specification of Tesla P100-PCIE-16GB and an Intel Xeon 2.20GHz processor. The Colab environment allotted 12 GB RAM and 34 GB of hard drive space for memory. We use the PyTorch toolkit to create the model in Python. We divided the dataset into three sets: a training set (8 months), a validation set (2 months), and a test set (4 months) (2 months). The hyper-parameters are fine-tuned using the validation set, and test data is used for the final performance assessment. Adam optimizer to train it with a $1e-3$ learning rate. In stacked GRU, we set the hidden dimension size to 32, and 4 GRU layers are used with the topmost layer as attention. In GateGCN, 2 layers are used, the global attention dimension is set to 32, and MLP layers are 3. Batch size for the experiment is set to 64, and regularization parameter λ is set to 0.01.

5.3 Performance Validation

We compare the performance of UrbanAnom with various baselines; Table 2 describes the results of urban anomaly prediction in terms of accuracy, precision, recall, and F-measure. In the urban anomaly prediction, UrbanAnom outperforms the existing baseline approaches, according to the evaluation results.

Table 2: Comparison with different baselines in percentage (%)

Methods	**Accuracy**	**Precision**	**Recall**	**F-Measure**
SVR [2]	69	68	65	66
LR [8]	67	66	67	66
ST-RNN [14]	72	69	70	69
LSTM [7])	75	71	73	72
GRU [3]	74	73	72	72
ARM [6]	79	77	76	76
UrbanAnom	**85**	**83**	**84**	**83**

Table 3: Comparison with different state-of-the-art models in percentage (%)

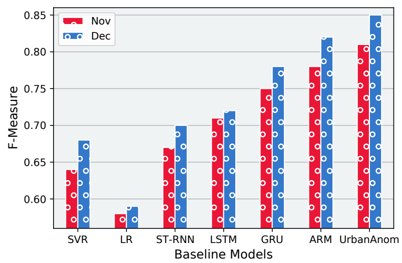
Methods	**Accuracy**	**Precision**	**Recall**	**F-Measure**
DCA [28]	-	75	62	70
CUAPS [9]	66	70	76	73
UAPD [21]	66	69	74	71
ind+int [25]	69	68	77	73
DAUAD [26]	-	70	75	74
ST-MFM [16]	74	73	80	79
DST-MFN [15]	-	77	84	81
UrbanAnom	**85**	**83**	**84**	**83**

This illustrates the advantage of taking into account the integration of semantic spatial, context-aware temporal modules. Second, methods based on neural networks perform better than standard machine learning because neural networks are better at learning hidden and non-linear correlations. Third, attention-based techniques capture long-term dependencies more effectively than a simple recurrent neural network. Finally, just using time-series data is inefficient compared to recurrent neural networks; in order to accurately forecast anomalies, we must include spatial, temporal, and semantic information.

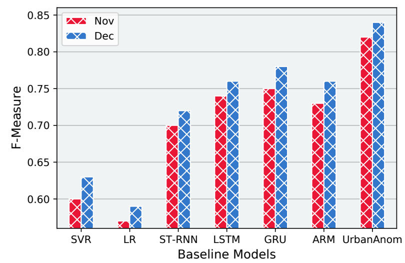
Additionally, we look into how well UrbanAnom predicts various categories of anomalies; the results are given in Fig.s 4a, 4b, 4c, and 4d. Noise, illegal parking, blocked driveways, and building use are some of the different categories we include for evaluation. We find that UrbanAnom performed better than baseline methods in predicting individual category anomalies and is capable of modelling region, time, and category data efficiently. Last but not least, we compare our model to current state-of-the-art urban anomaly prediction models. The results are reported in Table 3. As shown in Table 3, UrbanAnom predicts urban anomaly more accurately in terms of different performance metrics.

5.4 Parameter Sensitivity

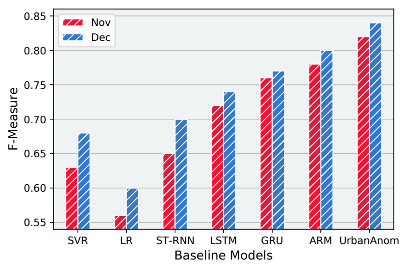
In this subsection, we examine the robustness of UrbanAnom, we examine the effect of different hyperparameter settings (i.e., attention dimension, hidden state



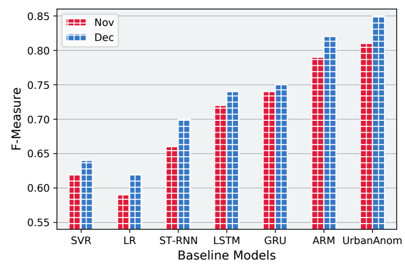
(a) Noise



(b) Illegal Parking

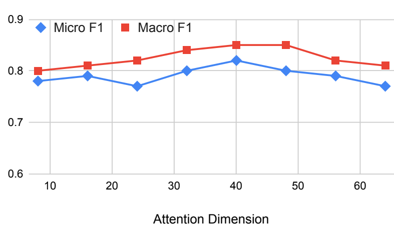


(c) Blocked Driveway

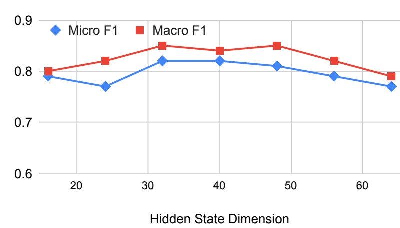


(d) Building/Use

Fig. 4: Predicting results for individual anomaly category



(a) Attention dimension effect



(b) Hidden state dimension effect

Fig. 5: Hyper-parameter sensitivity results

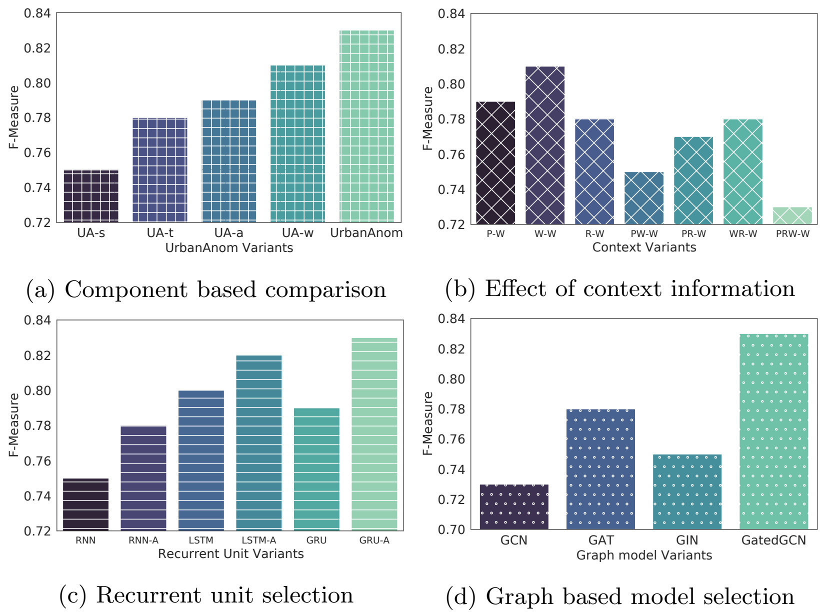
dimension), and the results are shown in Fig. 5a, and Fig. 5b. Other parameter are set to the default value, except the parameter being tested. We observe from Fig. 5a that increasing the dimension size increases the performance initially, but it also occurs with extra computation cost. Therefore, we set the dimension size as 32 to provide a balance between efficiency and computation cost. From Fig. 5b, it is also observed that we got peak performance when hidden state dimension set to 32. We also observe that both hyperparameters, i.e., attention dimension and hidden state dimension have less effect on the performance, which shows the robustness of the proposed framework.

5.5 Evaluation of Variants

In addition to evaluating UrbanAnom with baseline and state-of-the-art techniques, we also assess variants by doing an ablation study to determine the impact. For a deeper understanding of UrbanAnom, we also assess the framework from other aspects, including how the choice of various recurrent units affects performance and how various context factors, such as POI and weather, affect the outcome.

Ablation study of components Ablation study checks for the impact of an individual component on the framework performance. It can be inferred from the result that semantic spatial, temporal, attention and weather embedding module provides additional context regarding prediction. Specifically, spatial and temporal components plays significant role in correct prediction. 1) *Effect of semantic spatial module (UA-s)*: A simplified version of UrbanAnom that do not include semantic spatial data into consideration for evaluation. As shown in Fig. 6a, F1-measure of UA-s is 0.75 which is comparatively less than UrbanAnom. 2) *Effect of context aware temporal module (UA-c)*: Another variant of UrbanAnom which do not cover the temporal aspect of the urban anomaly prediction problem. As observed from the results shown in Fig. 6a that UA-t have F1- measure of 0.78, which is less than UrbanAnom. This effect raised beacause anomaly changes with respect to time and temporal component plays an important role in prediction. 3) *Effect of global attention module (UA-a)*: Model prediction score is observed in the absence of global attention; it helps to understand how attention is affecting the accuracy. Attention mechanism help to improve the performance of anomaly prediction correctly, and results shown in Fig. 6a validates this statement. UA-a has F1-measure of 0.79, its a significant reduction in accuracy of the proposed framework. Therefore, adding attention helps in predicting urban anomaly more accurately because it helps us to capture long term dependency in temporal and spatial dimensions. 4) *Effect of weather embedding module (UA-w)*: Weather information can be an important aspect on anomalies, so to check its effect, we evaluate our model without using it. As the intuition that weather may affect urban lives, results show its applicability. It is clear from Fig. 6a, UA-w performs less than UrbanAnom with F1-measure of 0.81.

Context, recurrent unit, and graph model selection effect The effect of context information, recurrent unit, and graph model selection is shown in Fig. 6. Insights of the Fig. 6 are discussed as: 1) *Context information effect:* To improve the accuracy of predictions, we incorporate context information such as POI (P), weather (W), and calendar (R) into our framework. The impact of contexts on the model's performance is important to know the effect of individual context contribution in total prediction accuracy. In 6b, the F-measure is shown for the emphcontext-W (without $POI/W/R$) information. For instance, $P-W$ stands for model performance without POI data. As shown, extra information improves the model's accuracy by 10%, and weather information has a large influence



(a) Component based comparison

(b) Effect of context information

(c) Recurrent unit selection

(d) Graph based model selection

Fig. 6: Evaluation of variants

across all situations. The removal of all the context categories $PRW-W$ decrease the prediction accuracy significantly. 2) *Recurrent unit selection effect:* We utilize different recurrent units, i.e., RNN, RNN-A, LSTM, LSTM-A, GRU, GRU-A for temporal insights, where A denotes local attention. It is clear from the results shown in Fig. 6c that selection of the recurrent unit makes a huge difference in the accuracy of the model. The first observation that we have drawn is that RNN has lower performance among all the variants and GRU-A has the best one. Second, local attention also increases the F1-measure, as all the variants RNN-A, LSTM-A, and GRU-A perfoms better than their base model. 3) *Graph based model selection effect:* We experimentd with different variants of GCN, i.e., GCN [13], graph attention network (GAT) [19], graph isomorphic network (GIN) [22], and GatedGCN for spatial relation extraction. It is clear from the results shown in Fig. 6d that GatedGCN have shown better performance than other graph based models. We can also infer that GCN has lower performance among all the variants.

6 Conclusion

To minimise the economic loss to our society, urban anomaly prediction is a crucial endeavour. In this research, we proposed a solution to the problem of urban anomaly prediction. In order to represent semantic spatial relations, the metropolitan area is separated into regions based on the road network. We employ the gated recurrent unit and take the weather into account to add context-aware temporal information. We successfully predict anomalies in the real-world dataset with an F-measure of 0.83 and compare the proposed method's performance with various baselines and state-of-the-art techniques to verify the model's performance. Additionally, we assess our suggested framework from a variety of angles, including the contributions made by each module, the significance of the contextual information, effect of graph based model and the influence of the recurrent unit on performance.

The dataset for anomaly collected from NYC 311 services which inherits all the issues of crowdsource data collection. For example, all the complaints are not registered in the portal, and all registered complaints are not validated. Multi-domain incorporation may increase the accuracy of the model; we will try to add accidents data, crime data, human mobility data, etc., so that better domain knowledge can be provided to the model. In the future, we try to use proposed architecture in other prediction problems also and want to incorporate more context information to simulate the real-world.

References

1. Bresson, X., Laurent, T.: Residual gated graph convnets. arXiv preprint arXiv:1711.07553 (2017)
2. Chang, C.C., Lin, C.J.: Libsvm: A library for support vector machines. ACM transactions on intelligent systems and technology (TIST) **2**(3), 1–27 (2011)
3. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)
4. Dash, D.K.: Traffic congestion costs four major indian cities rs. 1.5 lakh crore a year. https://timesofindia.indiatimes.com/india/traffic-congestion-costs-four-major-indian-cities-rs-1-5-lakh-crore-a-year/articleshow/63918040.cms (2018)
5. Dwivedi, V.P., Joshi, C.K., Laurent, T., Bengio, Y., Bresson, X.: Benchmarking graph neural networks. arXiv preprint arXiv:2003.00982 (2020)
6. Feng, J., Li, Y., Zhang, C., Sun, F., Meng, F., Guo, A., Jin, D.: Deepmove: Predicting human mobility with attentional recurrent networks. In: Proceedings of the 2018 world wide web conference. pp. 1459–1468 (2018)
7. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
8. Hosmer, D.W., Lemeshow, S.: Applied logistic regression. Wiley New York (2000)
9. Huang, C., Wu, X., Wang, D.: Crowdsourcing-based urban anomaly prediction system for smart cities. In: Proceedings of the 25th ACM international on conference on information and knowledge management. pp. 1969–1972 (2016)
10. Huang, C., Zhang, C., Dai, P., Bo, L.: Cross-interaction hierarchical attention networks for urban anomaly prediction. In: IJCAI (2020)
11. Huang, C., Zhang, C., Zhao, J., Wu, X., Yin, D., Chawla, N.: Mist: A multiview and multimodal spatial-temporal learning framework for citywide abnormal event forecasting. In: The World Wide Web Conference. pp. 717–728 (2019)
12. Jiang, R., Song, X., Huang, D., Song, X., Xia, T., Cai, Z., Wang, Z., Kim, K.S., Shibasaki, R.: Deepurbanevent: A system for predicting citywide crowd dynamics at big events. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 2114–2122 (2019)

13. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
14. Liu, Q., Wu, S., Wang, L., Tan, T.: Predicting the next location: A recurrent model with spatial and temporal contexts. In: Thirtieth AAAI conference on artificial intelligence (2016)
15. Liu, R., Zhao, S., Cheng, B., Yang, H., Tang, H., Li, T.: Deep spatio-temporal multiple domain fusion network for urban anomalies detection. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management. pp. 905–914 (2020)
16. Liu, R., Zhao, S., Cheng, B., Yang, H., Tang, H., Yang, F.: St-mfm: A spatiotemporal multi-modal fusion model for urban anomalies prediction. In: Proceedings of the Twenty-fourth European Conference on Artificial Intelligence (2020)
17. Qin, Y., Song, D., Chen, H., Cheng, W., Jiang, G., Cottrell, G.: A dual-stage attention-based recurrent neural network for time series prediction. arXiv preprint arXiv:1704.02971 (2017)
18. Shimosaka, M., Tsubouchi, K., Chen, Y., Ishihara, Y., Sato, J.: Moire: Mixed-order poisson regression towards fine-grained urban anomaly detection at nationwide scale. In: 2020 IEEE International Conference on Big Data (Big Data). pp. 963–970. IEEE (2020)
19. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017)
20. Wang, Y., Xu, J., Xu, M., Zheng, N., Jiang, J., Kong, K.: A feature-based method for traffic anomaly detection. In: Proceedings of the 2Nd ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics. pp. 1–8 (2016)
21. Wu, X., Dong, Y., Huang, C., Xu, J., Wang, D., Chawla, N.V.: Uapd: Predicting urban anomalies from spatial-temporal data. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 622–638. Springer (2017)
22. Xu, K., Hu, W., Leskovec, J., Jegelka, S.: How powerful are graph neural networks? arXiv preprint arXiv:1810.00826 (2018)
23. Yuan, J., Zheng, Y., Xie, X.: Discovering regions of different functions in a city using human mobility and pois. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 186–194 (2012)
24. Yuan, N.J., Zheng, Y., Xie, X.: Segmentation of urban areas using road networks. Microsoft Corp., Redmond, WA, USA, Tech. Rep. MSR-TR-2012-65 (2012)
25. Zhang, H., Zheng, Y., Yu, Y.: Detecting urban anomalies using multiple spatio-temporal data sources. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **2**(1), 1–18 (2018)
26. Zhang, M., Li, T., Shi, H., Li, Y., Hui, P.: A decomposition approach for urban anomaly detection across spatiotemporal data. In: IJCAI. pp. 6043–6049 (2019)
27. Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., Li, H.: T-gcn: A temporal graph convolutional network for traffic prediction. IEEE Transactions on Intelligent Transportation Systems (2019)
28. Zheng, Y., Zhang, H., Yu, Y.: Detecting collective anomalies from multiple spatio-temporal datasets across different domains. In: Proceedings of the 23rd SIGSPATIAL international conference on advances in geographic information systems. pp. 1–10 (2015)
29. Zhou, B., Chen, L., Zhou, F., Li, S., Zhao, S., Das, S.K., Pan, G.: Escort: Fine-grained urban crime risk inference leveraging heterogeneous open data. IEEE Systems Journal (2020)