Detecting Abnormal Behavior in the Transportation Planning using Long Short Term Memories and a Contextualized Dynamic Threshold

Thananut Phiboonbanakit
Japan Advanced Institute of Science and Technology
Ishikawa, Japan
Sirindhorn International Institute of Technology,
Thammasat University
Pathum Thani, Thailand
thananut@jaist.ac.jp

Van-Nam Huynh
Japan Advanced Institute of Science and Technology
Ishikawa, Japan
huynh@jaist.ac.jp

Teerayut Horanont
Sirindhorn International Institute of Technology,
Thammasat University
Pathum Thani, Thailand
teerayut@siit.tu.ac.th

Thepchai Supnithi
NECTEC, National Science and Technology Development
Agency
Pathum Thani, Thailand
thepchai@nectec.or.th

ABSTRACT

Unsupervised anomaly detection in time-series data is crucial for both machine learning research and industrial applications. Over the past few years, the operational efficiencies of logistics agencies have decreased because of a lack of understanding on how best to address potential client requests. However, current anomaly detection approaches have been inefficient in distinguishing normal and abnormal behaviors from high dimensional data. In this study, we aimed to assist decision makers and improve anomaly detection by proposing a Long Short Term Memory (LSTM) approach with dynamic threshold detection. In the proposed methodology, first, data were processed and inputted into an LSTM network to determine temporal dependency. Second, a contextualized dynamic threshold was determined to detect anomalies. To demonstrate the practicality of our model, real operational data were used for evaluation and our model was shown to more accurately detect anomalies, with values of 0.836 and 0.842 for precision and recall, respectively.

CCS CONCEPTS

• **Computing methodologies → Machine learning**; **Anomaly detection**; **Neural networks**.

KEYWORDS

Anomaly detection, Neural networks, Recurrent Neural Network, Long Short Term Memory, Logistics, Time-series, Forecasting

UbiComp/ISWC '19 Adjunct, September 9–13, 2019, London, United Kingdom
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6869-8/19/09…$15.00
https://doi.org/10.1145/3341162.3349324

ACM Reference Format:
Thananut Phiboonbanakit, Van-Nam Huynh, Teerayut Horanont, and Thepchai Supnithi. 2019. Detecting Abnormal Behavior in the Transportation Planning using Long Short Term Memories and a Contextualized Dynamic Threshold. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2019 International Symposium on Wearable Computers (UbiComp/ISWC '19 Adjunct), September 9–13, 2019, London, United Kingdom.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3341162.3349324

1 INTRODUCTION

Unsupervised anomaly detection is important for both machine learning and industrial application in many areas such as cybersecurity [1] and image processing [21]. In the logistics industries, the rapid growth of freight transportation in urban areas results in many challenges for transit and logistics agencies in terms of controlling costs for each of their business units and creating higher efficiency routing plans in the logistical area. The vehicle routing problem (VRP) is the most important topic to support agencies' decision making in routing planning, including optimizing their operational cost and traveling distance [18] and satisfying their operational constraints [10][13][8]. However, when route planning is launched, it involves utilizing data to support the planning. Considering only distance and traveling time results in management addressing only one narrow aspect. The main challenge is that the current increase in business data complexity and the limits of data labelling cause ineffectiveness in interpreting data in existing study approaches. It is necessary to distinguish normal and abnormal behavior from the planning flow in operation. As a result, inaccurate identification of abnormal behavior can adversely impact overall operational performance. For abnormal behavior in this research context, we address abnormality in the operational work flow where fleet utilization and task assignment were not in a regular state or overutilized. Previous studies have continued to improve detection by adopting innovative anomaly detection models. However, these are subject to false positive rates during the detection process. It is not possible to make estimations using a density-based approach on time series data because it can miss anomalies that occurred in the specified limit or are characterized by temporal dependencies[6]. The real-world data from the system are usually highly non-stationary and dependent on the current context. Data being monitored are often heterogeneous, noisy, and high-dimensional [11]. Recent research has shown that to address temporal dependencies, one can utilize Long Short Term Memory (LSTM) for prediction and estimate the error based on the assumption that if the error is high, the data at that point of time is not dependent on other data points. In addition, a threshold is required to support the estimation process.

However, an unanswered question remains regarding the ability of the current anomaly detection model to address data temporal dependencies and the variety of anomalies in a business area. Our research question was defined as follows: how can we address temporally dependent data without losing information to detect abnormal business operational behaviors? Moreover, how we can contextualize the anomalies? Also, what is the root cause of the anomalies? The lack of a solution to bridging the gap in current study approaches causes a misunderstanding in how best to address a potential client's request because we cannot detect an abnormal pattern if essential information, such as temporal characteristics, is not considered and the context is not defined. This leads to a downturn in company profits and operational opportunities. To be able to support their business, agencies require research into a more effective data analytical approach to aid in their management strategies.

Contribution In this study, we adapt and extend the method from various domains to address and fill the gap in solving the aforementioned issues. This work presents, through the field of logistics industries, anomaly detection. However, in addition, our proposed approach can be applied to many other application domains. It is not specific only to logistics industries which involve anomaly detection in multivariate time-series. We describe our use of LSTM to achieve high prediction performance while maintaining model efficiency and data interpretability throughout the process. Once the model performs a prediction, we propose a context, dynamic, and unsupervised threshold, including a weighted average method for evaluating a series of data. This approach addresses diversity, non-stationarity, and noise issues associated with automatically setting thresholds for data streams characterized by varying behaviors, such as their context link to their behavior or deviations from the regular group or the defined thresholds to the approach also supports identification of the root cause of the anomalies. We then present experimental results using real-world data derived from a global positioning system (GPS) probe, logistics agencies' reports of operation, and an expert-interview report in an aggregated form. Finally, we highlight the essential point, improvements, and observations identified through our experiment and what should be considered in further research.

This paper is organized as follows. Section 2 focuses on the anomaly detection method. Section 3 provides a detailed statement of the problem, its significance, and the motivation driving this research. Section 4 presents the methods used to conduct data analysis processing and an experiment regarding fleet management anomalies is discussed. The proposed model is also described. Section 5 explains the factors involved in anomaly detection and a comparative experiment of fleet management systems. Section 6 summarizes our findings, draws conclusions based on our research objectives, and suggests potential improvements to this study.

2 RELATED WORK

2.1 Anomaly Detection

In our study, we discovered three types of anomalies consisting of point, contextualized, and collective anomalies. The definition of each type is as follows. First, for point anomalies, the data point is an anomaly if it is significantly different from the others, measured by the Euclidean distance, similarity, and dissimilarity. Second, for contextualized anomalies, the data point, or sub-sequence, is an anomaly if it is significantly different from the context rule and behavior. For example, contextual in terms of time, quarter, date, and action under the condition of demand, order, and number of available vehicles. Finally, collective anomalies are a sub-sequence within a long sequence significantly different from the others. This study shows that this anomaly can be detected by the sliding windows approach, which calculates the statistical threshold if the data point differs from others in each defined time window. [15][11]. The objective when developing an anomaly detection method is to detect unexpected events or behaviors using a probabilistic distance- (clustering), reconstruction-, domain- (classification), or theory-based informational approach on available data [12]. To be more specific, anomalies in the logistics industries for this research context are addressing operational management error where the task assignments or fleet utilization are not regularly defined and distributed. The anomalies were influenced by fleet utilization, client demand, driver, and asset availability. We used historical data to evaluate the pattern to determine the overall performance of the agencies' operation to assist the agencies in their decision making.

2.1.1 Anomaly Detection using Long Short Term Memory (LSTM). Long Short Term Memory (LSTM) was adopted to determine the temporal correlation of time series attributes to estimate the current predictor time. The aim of this approach was to identify collective anomalies. [11] performed LSTM to detect abnormalities in spacecraft operation. They analyzed the data by adopting a dynamic threshold with a statistical method for detecting anomaly sequences which deviate from and exceed the limit. LSTM also was used to detect anomalies in the time-sequences of computer network usage by estimating the prediction error threshold calculated from the relative error, relative error threshold, minimum attack time, and danger coefficient [4]. However, these approaches can identify the majority of sequences and point anomalies but also have a high false positive rate in their detection results.

2.1.2 Anomaly Detection using a Reconstruction-Based Approach. The reconstruction-based approach assumes that anomalies were not effectively reconstructed from the low dimensional space. This approach aims to detect am outlier and rare event data point which does not conform with other data points. The method used in this approach was Principle Component Analysis (PCA). However, the disadvantage of using traditional PCA is that it reduces the data dimension and the critically important information that is required to detect outliers or anomalies is removed. Thus, during recent years, researchers have adopted an improved method such as Robust PCA (RPCA) that is less sensitive to noise [3]. In addition, some recent works have proposed to analyze the reconstruction error from deep auto-encoders and the experimental results show that it is efficient in detecting anomalies [5][12][20]. However, the detection is based on reconstruction error, which covers only one aspect to detect anomalies. In a further study, the abnormalities did not always have a high reconstruction error, which means that the data points may appear with common data points which have a low reconstruction error [3].

2.1.3 Multi-level Anomaly Detection. From the survey in this research area, multilevel anomaly detection is widely adopted on the assumption that data have a high dimension and it is not convenient for clustering based approaches such as Kmean, Support Vector Machine (SVM), or Gaussian Mixture Model (GMM) to perform density estimation and anomaly detection. Therefore, a two-step approach was adopted [6] in which the dimensional reduction was first conducted and followed by clustering analysis. This approach has a drawback wherein there is separate learning between the dimension reduction and clustering. The key information of importance for clustering analysis could be lost during the reduction operation. Joint learning is also of growing interest which aims to improve detection accuracy, making the detection more reliable and improving the computational time by dividing a detection phase into hierarchical levels and then completing an ensemble prediction. The model is a combination of an auto-encoder for dimensional reduction and an ensemble k-nearest neighbor for clustering [17]. Another research study also used auto-encoder as a base model and performed density-based spatial clustering of applications with noise (DBSCAN) to perform density estimation [2]. The combination of supervised and unsupervised learning also has been adopted to detect Denial of Service (DoS), Probe, and Normal in the hierarchical level. It is a combination of Catsub, K-Point, and the outliner detection method. [9]. Another approach is the multilevel hybrid model where a combination between support vector machine, extreme learning machine, and modified Kmean is used in the computer network [1]. A combination of neural and decision tree has been used to detect in a cyber-network. The hybrid and multilevel model aims to improve detection accuracy and reduce the false positive rate [16].

However, none of these studies have addressed the perspective of anomaly detection in logistics research. More specifically, the anomaly detection approach, which can detect point, contextualized, and collective anomalies in multi-level dimensions, including identified their root-cause, has not been widely discussed. In addition, it is essential to aggregate temporal dependencies to support anomaly detection where anomalies need to be considered in various aspects as in a real operation, which presents a challenge. This challenge motivated us to address this problem; thus, during recent years, LSTMs have demonstrated that it is possible to learn and detect temporal dependencies over various domains. In this study, we propose an unsupervised model to detect anomalies using multivariate time-series data as an input to LSTM to determine the temporal dependencies of time-sequence and detect collective and contextualized abnormalities leveraged from a dynamic threshold and anomaly contextualization approach.

3 PROBLEM DEFINITION

Logistics agencies have faced a challenge in managing their operations because of the uncertainty of the demand in the market. Both business and data analytic problems exist. The business problem is defined as the lack of understanding how best to address a potential client's request leading to a downturn in company profits and operational opportunities. To be able to support their business, agencies require research into more effective data analytic approaches to aid their management strategies. By analyzing historical data from logistics agencies, it is shown that during 2 years of operation, agencies held unexpected incident requested consisting of 7% of the overall work, which appears as a low percentage and also the number of abnormal behavior seem to be unbalance between regular behavior. However, if the impact were calculated in terms of losses, it would demonstrate that the profitability of the logistics agency operation can be materially impaired. These problems have been a link to the data analytical problem as an inefficiency to distinguish between non-anomaly and anomaly data points using the current detection approach on high-dimensional data. The existing detection approaches require modification to preserve the essential information. It has been required to detect data points which are characterized by a temporal dependency element. The non-linearity and high variability of the data decrease the current detection approaches performance. Therefore, in this study, we proposed an unsupervised anomaly detection model that leverages from the dynamic threshold and anomaly contextualization approach to solve the gap of existing detection. It also addresses the issues introduced at the beginning, which are based on the assumption that an anomaly that does not have temporal dependencies and rarely occurs is subject to a high prediction error rates when compared to that of the other data points. In addition, it is also imbalanced when compared to the overall dataset and the data contains a high degree of diversity. This study, therefore, involves the processing of data as it is collected and utilizes the data to perform analysis and anomaly detection. All anomalies in this study were defined as point, contextualized, and collective as described in section 2. For point anomalies, we can set straightforward single values that fall within low-density value regions. Collective and contextualized points were determined by the context, in terms of the day as a context, and the behavior between the request received from the client was compared to action in fleet management such as the number of available vehicles in the fleet and incident rate. An example of anomalies in the logistical area is shown in Figure 1. The figure on the left shows the abnormalities which independently occurred from the other attributes and the second is combined (in the yellow square) and context with thresholds (in the red square) with another characteristic to define as anomalies. To be more specific, anomalies in the logistics industries for this research context are addressing operational management error where the task assignments or fleet utilization are not regularly defined and distributed. The anomalies were influenced by fleet utilization, client demand, driver, and asset availability. We used historical data to evaluate the pattern and determine the overall performance of the agencies' operation in an effort to assist the agencies' decision making.

4 DEVELOPMENT OF THE LSTM-BASED ANOMALY DETECTION MODEL

4.1 Data Collection and Pre-processing

First, data were collected from GPS tracker equipment and agencies' operational reports. It consists of a GPS probe from sixty trucks with installed GPS tracker equipment, a client order. The collected data from these two modules we then divided into four data formats consisting of an order confirm, vehicle statistic, driver statistic, and order. These data were subject to feature extraction using a business intelligence framework. To obtain all necessary features, first, we performed a data acquisition to extract features and in preparation for analysis. Then, we loaded the data to store in data storage. A Power BI was used to determine the data relationships for measuring the fundamental statistic of each feature and creating a linkage to other data attributes within various perspectives, as shown in Figure 3. From the previously described operation, we obtained the statistics on the number of vehicles used, available vehicles, drivers, relationship between vehicle and assigned driver including their orders, and so on, as the form of multivariate time-series shown in Table 1 and Figure 4.

From Table 1, the definition of import and export are described as follows:

- Imports are tasks which deliver goods or products from local agencies to overseas destinations.
- Exports are tasks which deliver goods or products from overseas to local destinations.

4.2 Preliminary Experiment on Anomalies Contextualization

The contextualized anomalies required a context link with a behavior [6] for detection. Therefore, the motivation to conduct this experiment was that the Bayesian network can define the linkage between data attributes. It is also among the methods in the graphical model used to represent the dependency of the event and the evidence in the dataset. The model formulation is described in Section 4.2.1. The assumption is if the linkage has a low probability, then that data connection rarely occurs, and thus we can use the origin and destination of that linkage to be the context and the threshold to define the anomalies.

4.2.1 *Define data attribute dependency using Bayesian Network.*

After we obtained the data to prepared for analysis, we then determined the dependencies and set of rules to contextualize anomalies from the dataset. Bayesian Network-Based Approaches were used for anomaly contextualization as in Equation (1):

$$\Pr(e|m) \tag{1}$$

where e is an event (or evidence for an event) and m is the model. To determine context of an anomaly, we need to specify the threshold (t) as represented in Equation (2):

$$\Pr(e|m) < t \rightarrow anomalous \tag{2}$$

For time-series or data that have a sequence of events, a process to contextualize the anomaly (aggregated)is required. The equation is modified based on [14] as Equation (3):

$$\frac{1}{N}\sum_{i}\Pr(e|m) < t \rightarrow anomalous \tag{3}$$

where N is a time-step i and m is the model. If we would like to determine conflicts within a set of evidence, we use "conflict measure" [14] to detect possible incoherence in evidence $\mathbf{E} = \{E_1 = e_1, ..., E_m = e_m\}$ as in Equation (4):

$$C(\mathbf{E}) = \log\frac{\Pr(E_1 = e_1) \times ... \times \Pr(E_m = e_m)}{\Pr(\mathbf{E})} \tag{4}$$

After using the Bayesian Network, we were able to obtain context and behavior for the defined anomalies. We set it from the low probability of the linkage. Further discussion is provided in the results section.

For the context, vehicle (V) usage not in the threshold of a vehicle usage range. We set it up as the $threshold_v$. It is represented as Equation (5).

$$y_1 = \begin{cases} 1, & \text{if } V \neq threshold_v \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

The condition where the value for the received request (R) exceeds the threshold of a vehicle usage. We set it up as the $threshold_v$ for vehicle usage is represented as Equation (6).

$$y_2 = \begin{cases} 1, & \text{if } R > threshold_v \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

The condition where the number of incidents (I) has increased and exceed the threshold of an incident is represented as Equation (7).

$$y_3 = \begin{cases} 1, & \text{if } I > threshold_I \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

If any of these conditions is matched, we consider anomalies to have occurred during the operational process. From this context, we group the definition to define possible anomaly and not-anomaly points as shown in Table 3.

4.3 Proposed model for anomaly detection

The proposed novel unsupervised anomaly detection method was used to determine and detect a fleet management transaction and whether it results in prediction error and the defined contextualized threshold represents an operation anomaly from the multivariate time-series data. This study also was motivated by a previous study's challenge to address temporal dependency data and reduce the false positive rate. As mentioned by [11], a single model of LSTM with dynamic threshold was able to detect the majority of the anomalies. However, the false positive rate remained high. The main reason is that the model was efficient in detecting some anomalies but other variants remained limited. In other words, anomalies can be influenced by environmental factors. Thus, the model could not detect all types of anomalies in the system. We contributed to the contextualization threshold in the model which increases the capability of detecting specific anomalies from an upper-level perspective in an urban logistics transportation operation in which one

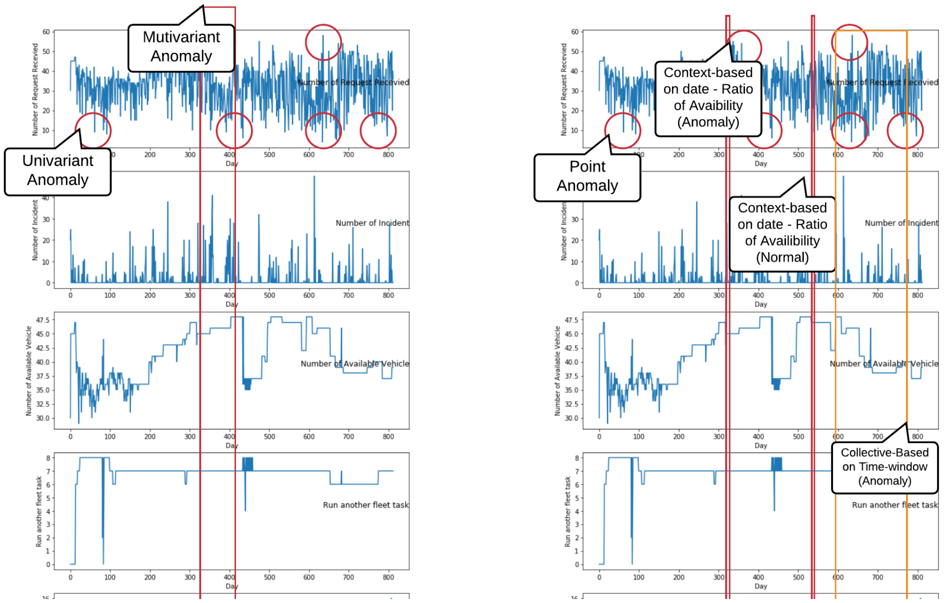

Figure 1: Example of anomalies held by logistics agencies

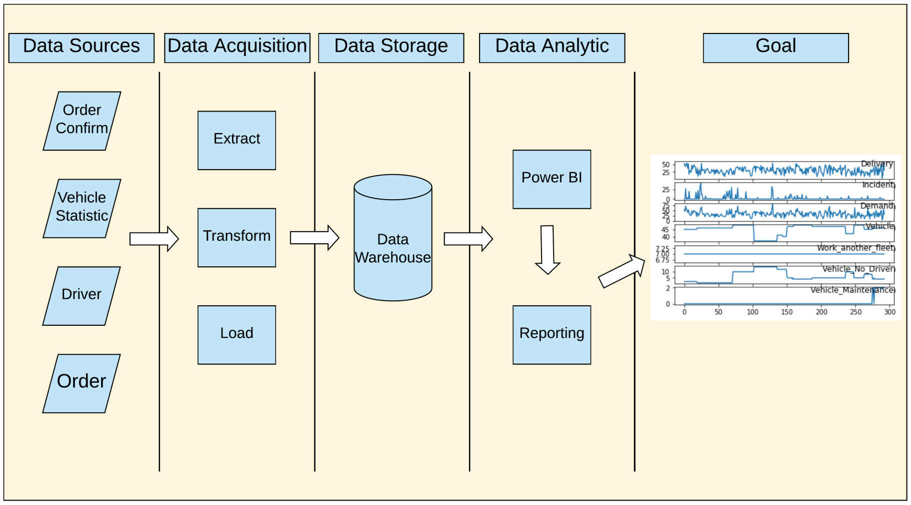

Figure 2: Framework of business intelligence

could not only determine a prediction error and statistical threshold, or use a density-based approach, but could also detect influences from environmental factors. For example, from the client, operation, asset, and human resource factors. Moreover, these contexts helped to identify the root cause of the anomalies in the fundamental step. The overall proposed model is shows as Figure 5.

4.4 Multivariate time-series prediction using LSTM

At this stage, we took the data which were prepared in Section 4.1 as inputs. The basic principle behind a recurrent neural network (RNN) is to leverage the following information in the input to make a prediction. The LSTM was used to train the sequences of the time-series data. However, in a traditional neural network, inputs and outputs are independent of each other. Therefore, when making a prediction, it is important to know the previous steps. This type of neural network is termed recurrent because it performs the same

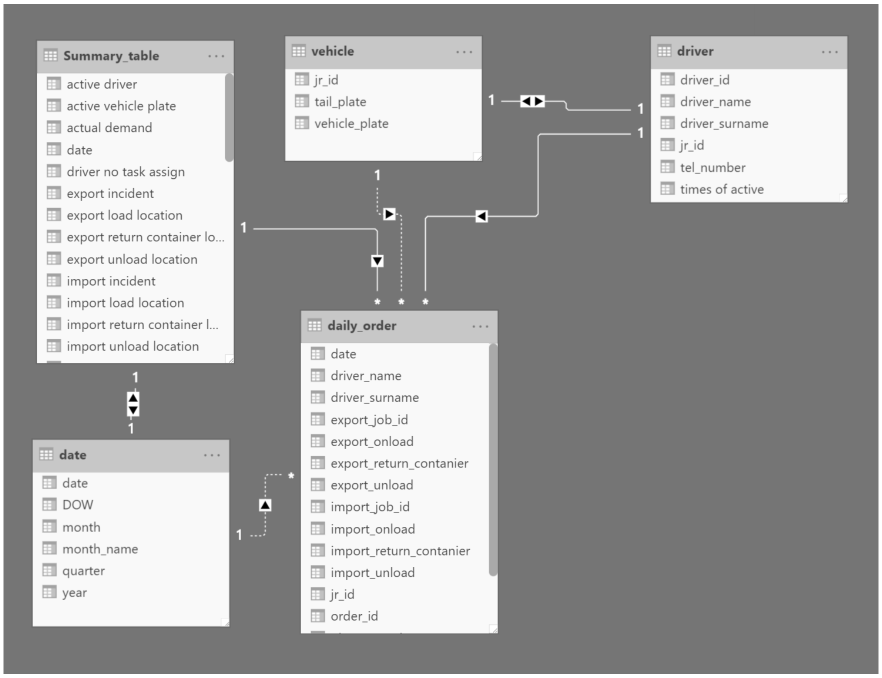

Figure 3: Data relationship from multi-data sources

Table 1: Data specification of daily reports transformed from a GPS probe and company report

Attribute(s)	Example of Data	Unit
Date	2/10/2018	-
Number of Available Vehicles	45	Vehicle
Number of Occupied Vehicles	7	Vehicle
Number of Vehicles with No Assigned Driver	8	Vehicle
Number of Vehicles with Back Order Work	0	Vehicle
Number of Vehicles in Maintenance	0	Vehicle
Number of Vehicles with Driver Taking Leave	0	Vehicle
Number of Total Requested from Client	45	Order
Number of Requests Received	45	Order
Number of Orders Cancelled or Postponed (Import)	0	Order
Number of Orders Cancelled or Postponed (Export)	0	Order
Quarter of the Year	Q4	-

computation for all elements in a sequence of inputs, and the output of each element depends, in addition to the current input, on stored state data. [19]. The core principle is to improve the network by providing it with explicit memory. These frameworks are equipped with special hidden units. The resultant behavior is that previous input can be remembered for a long time. Multivariate time-series forecasting with LSTM was used to predict the received request from the client and vehicle availability in the fleet. We evaluated the model's root mean squared error (RMSE) and the absolute error as in Equation (8). The prediction reflected the temporal dependencies based on the assumption that the time sequences where a high error rate of prediction meant the time-step did not regularly occur because it did not have influence from the previous time-step attributes to make the prediction.

$$e^{(t)} = |y_t - \hat{y}_t| \tag{8}$$

Where e^t denotes the prediction error of each time-step, y_t denotes a true observation, and $\hat{y}_t$ denotes the prediction result from the input features.

4.5 Contextualize Dynamic Threshold

After the prediction result was obtained. We utilized and modified the dynamic threshold approached proposed by [11] for use in our study. The prediction error of each time-step was representative of

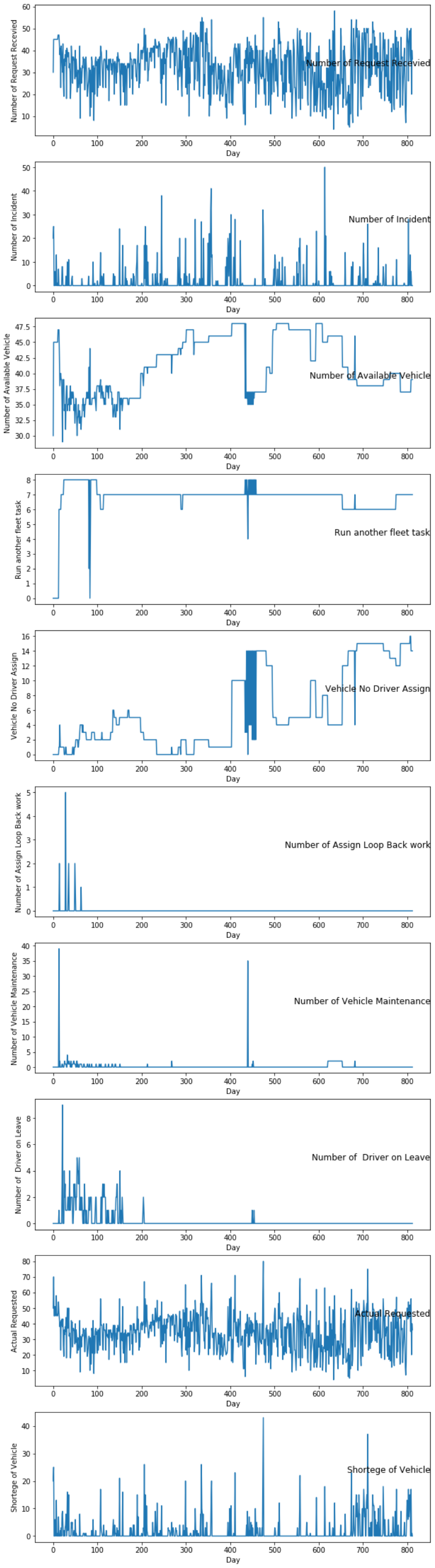

Figure 4: Visualization of collected data

one-dimensional vector errors:

$$e = [e^{t-h}, ..., e^{(t-l_s)}, ..., e^{(t-1)}, e^{(t)}]$$

where h is the historical error of the previous time step. The set of errors e is now smoothed to reduce the spike error generated from LSTM. Sometimes it was not perfectly predicted when the data point was in the not-abnormal state as shown in the [11] experiment. We used the exponentially weighted average (EWMA) to generate smoother errors where each weight of the data point was determined from the previous time-step $(t-1)$. Then, we obtained the smoothed error:

$$e = [e_s^{t-h}, ..., e_s^{(t-l_s)}, ..., e_s^{(t-1)}, e_s^{(t)}]$$

To evaluate whether values are non-anomaly, we set a threshold for their smoothed prediction errors. The values corresponding to smoothed errors above the threshold were classified as anomalies. In our study, the proposed thresholds which were defined as the context from section 4.2 and the smoothed error previously described were used to fill the gap of specific anomalies where it required a context of behavior to make a detection. We propose an unsupervised method without the use of labeled data. First, the $threshold_p$ (smooth error threshold) was defined as Equation (9).

$$threshold_p = \mu(e_s) + \mathbf{z}\sigma(e_s) \tag{9}$$

Where $\mathbf{z}$ represents a positive value of standard deviation above $\mu(e_s)$ We discover from our experiment that number of z greater than two was increasing the detection rate and reducing the number of positive errors. Next, the threshold for an incident context in each time step (t) was defined as Equation (10). It was used to determine the period that contained the number of incidents over the regular transaction in operation.

$$threshold_I = \mu(i) + \mathbf{z}\sigma(i) \tag{10}$$

where $\mathbf{z}$ represents the positive values of standard deviation above $\mu(i)$. The threshold for the vehicle usage context in each time step (t) was defined as Equation (11). It was used to determine the period that contained several vehicle usages over the regular transaction in operation and for decision a defining an anomaly in Equations (5), (6), and (7).

$$threshold_v = \mu(v) \pm \mathbf{z}\sigma(v) \tag{11}$$

Finally, we utilized Equations (5), (6), (7) and (9) to determine the anomaly score by using a weighted average method [7] as Equation (12).

$$score_i = \frac{\sum_{i=1}^{n}(y_i w_i)}{n} \tag{12}$$

where n is the number of contexts for a specific anomaly, w is the assigned weight for each context (for this study we assigned one where it was equally important), and y is the decision in each context represented in Equations (5), (6), and (7). If the score exceeded the typical score obtained from the regular data point, then we classified the data point as an anomaly. We also applied a sliding window approach to make a detection in the series of anomalies that occurred in the time-sequence. It was based on the assumption that if the majority of data points in the windows have a $score_i$ exceeding the average score of the regular data point, then all of

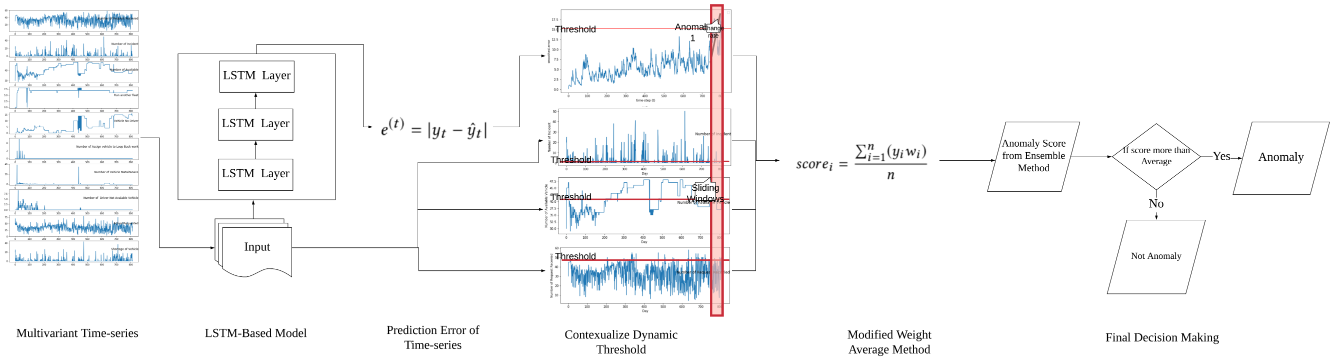
Number of Request Recevied
Number of Incident
Number of Available
Run another fleet
Vehicle No Driver
Number of Assign vehicle to Loop Back work
Number of Vehicle Mataitanace
Number of Driver Not Avalable Vehicle
Actual Requested
Shortege of Vehicle
LSTM Layer
LSTM Layer
LSTM Layer
Input
$e^{(t)} = |y_t - \hat{y}_t|$
Threshold
Anomaly 1
Change rate
smoothed-error
time-step (t)
Threshold
Number of Incident
Day
Threshold
Sliding Windows
Number of Available Vehicle
Day
Threshold
Number of Request Recevied
Day
$score_i = \frac{\sum_{i=1}^{n}(y_i w_i)}{n}$
Anomaly Score from Ensemble Method
If score more than Average
Yes
Anomaly
No
Not Anomaly
Multivariant Time-series
LSTM-Based Model
Prediction Error of Time-series
Contexualize Dynamic Threshold
Modified Weight Average Method
Final Decision Making

Figure 5: Demonstration on the flow of data which input into the model, first it is determined the temporal dependencies and prediction error from LSTM and then evaluate the prediction error and supported context with the proposed contextualize thresholds. Finally, the evaluation results were combined with weight average method and return the final decision for anomaly detection.

the points were classified as anomalies. In our experiment session, it was shown how the detection performance impacted the size of the window. In addition, we conducted an experiment on the dependencies of the previous data points with the assumption that previous data points can influence the current data in becoming anomalies. The percentage of different (d) of $score_i$ and $score_{i-1}$ was defined as Equation (13).

$$d_i = \frac{|score_i - score_{i-1}|}{score_i} \tag{13}$$

where i is the time-step; if d_i exceeded the average of all sets of d_i in the specific windows, then we classified all of the data points as anomalies. The summary of the procedure is shown in Figure 6. We first slide the window from the beginning of the series. Then, we calculated the anomaly score as defined in Equation (12). This equation was supported by threshold and decision conditions which were previously described. Then, one can determine the change between time-step (t) and the previous time-step (t-1). Whether the values of each factor violate the threshold was assessed and then an anomaly score was assigned to each data point in the windows. We repeated the procedure until the end of the time-sequence.

4.5.1 Performance Evaluation. During this stage, we evaluated the detection capability of our proposed model using an Area Under the Roc Curve (AUC), precision, and recall. The AUC measures the entire two-dimensional area under the entire receiver operating characteristic curve (ROC). The ROC was defined by applying a different threshold in a comparison between True Positive Rate (TPR) and False Positive Rate (FPR)). The equations for calculating TPR and FPR are Equation (16) and Equation (17), respectively.

$$TPR = \frac{TP}{TP + FN} \tag{14}$$

where TP denotes the number of true positives and FN denotes the number of false negatives.

$$FPR = \frac{FP}{FP + TN} \tag{15}$$

where FP denotes the number of false positives and TN denotes the number of true negatives.

The result is also validated using ground truth data in terms of its detection accuracy, precision, and recall including a confusion matrix. The precision and recall evaluation matrices are defined as Equation (18) and Equation (19, respectively.

$$Precision = \frac{TP}{TP + FP} \tag{16}$$

where TP denotes the number of true positive and FP denotes the number of false positives.

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

where TP denotes the number of true positives and FN denotes the number of false negatives.

5 RESULT AND DISCUSSION

Given the lack of data labelling from the data set, it is necessary to define a context for the anomaly in the urban logistics operation. As in the procedure in Section 4.2, we derived a result using the Bayesian Network as shown in Table 3.

From the analysis, anomalies in the fleet management system mainly occur from vehicle issues. The calculation result in section 4.2 demonstrated that the linkage between the vehicle to received request(Actual delivery) and the incident most likely cause anomalies in operation. The results showed that there is a low possibility to have a regular link and when we calculate Equations (3) and (4) a high anomaly score occurs, showing that the anomaly scores were higher than those of other linkages as shown in Table 3. We conclude that the root cause of the anomalies in urban logistics transportation mainly occurred from vehicle issues, which led to abnormal business behavior to address potential requests from clients and incidents occurred as a consequence. From this outcome, we set vehicle usage as a context in each time step and combined it with behavioral decision in Equations (5), (6) and (7), respectively. The summaries of context for defined anomalies are shown in Table 4.

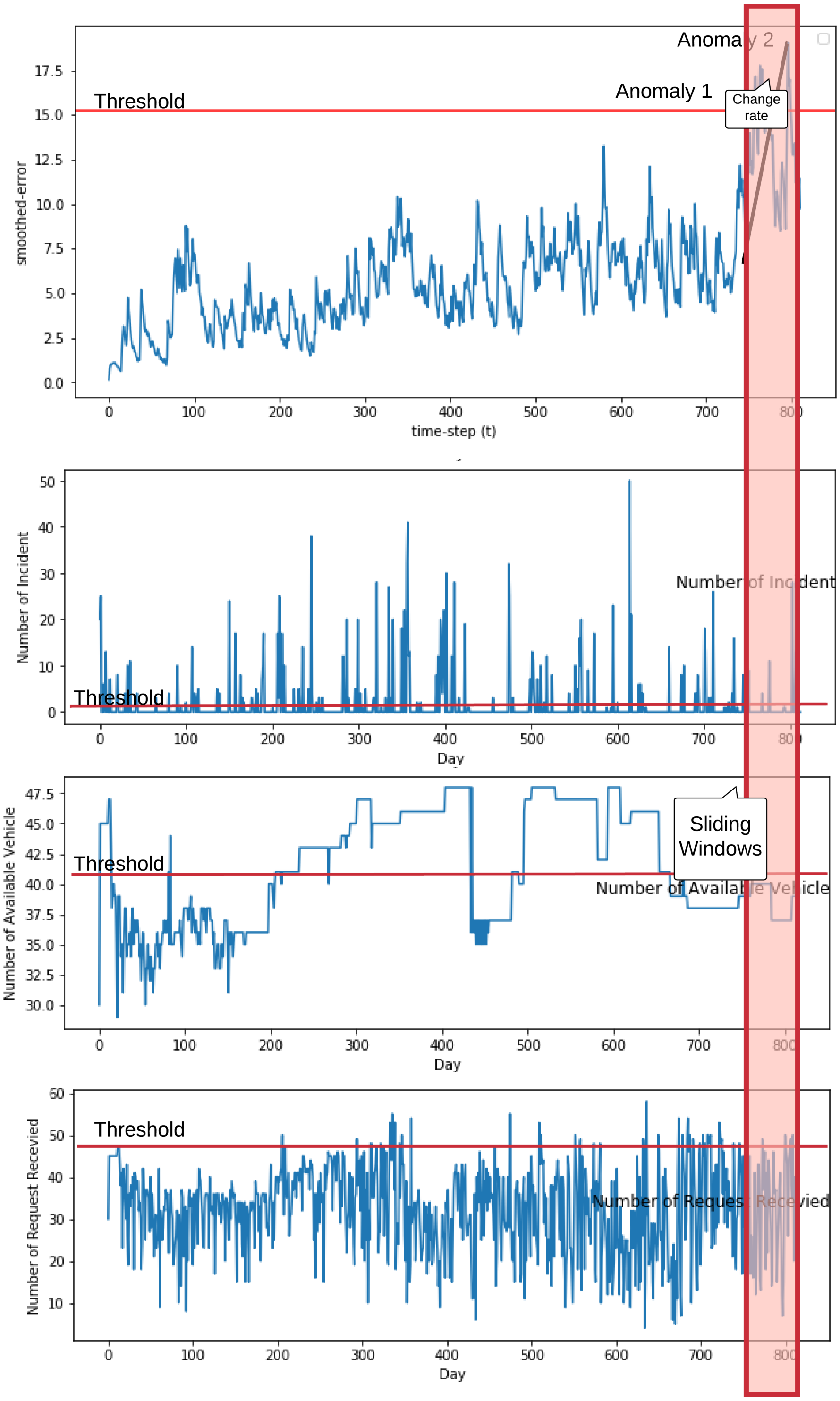

Figure 6: Demonstration of the sliding window with contextualized dynamic threshold showing that the smoothed error rate exceeded the threshold in the specific window size which was supported by the number of incidents. It received the requests as they exceeded a regular setting and the insufficient availability of vehicles in the fleet. In other words, these factors were the root causes that influenced the anomalies.

Table 2: Summary of anomaly scores using the Bayesian Network generated from multivariate time-series

Attribute(s)	Received req.	Incident	Request	Vehicle	Swap Fleet	No Driver	Maintenance
Received req.	0	1	1	**9**	2	3	1
Incident	2	2	2	**12**	2	1	3
Request	1	0	0	0	2	0	1
Vehicle	**11**	**12**	3	0	3	4	4
Swap Fleet	2	2	3	1	0	2	2
No Driver	2	2	2	0	2	0	2
Maintenance	1	2	2	0	2	1	0

Table 3: Criteria to Determine Anomalies

Description	Status
Received Request(R) and Vehicle (V) not exceed $threshold_v$ and incident not exceed $threshold_I$	normal
Received Request(R) and Vehicle (V) exceed $threshold_v$ and incident exceed $threshold_I$	anomaly
Received Request(R) exceed $threshold_v$ and Vehicle (V) not exceed $threshold_v$ and incident not exceed $threshold_I$	anomaly
Received Request(R) not exceed $threshold_v$ and Vehicle (V) exceed $threshold_v$ and incident exceed $threshold_I$	anomaly

Table 4: Parameter setting for the LSTM model

Model Parameter(s)	Values
hidden layer	2
unit in hidden layer	80
sequence length	730
training Iteration	1000
dropout	0.3
batch size	72
optimizer	adam
input dimension	12

5.1 Model Construction and Parameter Evaluation

After we defined the contexts for anomalies, then we constructed an LSTM model and set up parameters from hyper-parameter tuning as in Table 5.

5.2 Experimental Result

Next, we input all the data features into the model for training and testing. We performed the procedures described in Section 4.5 to calculate prediction error and defined a set of thresholds. In our experiment, we divided it into sub-experiments, where we detected only point , contextualized, and collective anomalies. The last was combined with both types of anomalies. First, table shows the experiment on point and collective anomaly detection when applying only a prediction error threshold and sliding windows approach.

From the result, it shows that when the window size increased, the AUC also increased. However, the values of the AUC were not significantly high, a consequence of the false positive and true positive rate. This shows that the data points contained dependencies to change the previous time-step as the estimate in Equation (13) to influence the status of the current time-step data point in the window for defining anomalies denoted by the anomaly score in Equation (12). The anomaly score was based on a threshold defined earlier in section 4.5 and was combined as anomaly scenarios as in Table 3. However, this also causes a high false positive where it detected regular data points as anomalies as the window size increased. We chose windows with a size equal to 1 as our result in this experiment where there is a high chance to correctly predict versus other window sizes. From this experiment, we set an assumption as it required a context and estimated the different changes in which tricked the next time step to be anomalous to support the detection. We then completed the next experiment only specifying the context and did not consider the previous time-step values, as shown in Table 6.

Table 5: Experiment 1 on the changes between each time window and the dynamic threshold

Window size	AUC	Precision	Recall	FPR
1	**0.572**	**0.714**	**0.189**	**0.047**
2	0.573	0.673	0.209	0.063
3	0.582	0.673	0.234	0.070
4	0.588	0.667	0.253	0.078
5	0.588	0.646	0.266	0.090
6	0.590	0.638	0.278	0.098
7	0.588	0.646	0.266	0.090
8	0.597	0.632	0.304	0.109
9	0.593	0.615	0.304	0.117
10	0.593	0.615	0.304	0.117

Table 5 shows that our set of data points does not contain the change dependencies of time-step. Thus, in this experiment, we then changed the model to the context of behaviors that described the decision condition and shown in Table 3. This was supported by the threshold in section 4.5 and the anomaly score in Equation

Table 6: Experiment 2 on the contextualized threshold of each time window

Window size	AUC	Precision	Recall	FPR
1	**0.504**	**0.88**	**0.696**	**0.059**
2	0.689	0.619	0.608	0.230
3	0.658	0.549	0.639	0.324
4	0.597	0.471	0.627	0.434
5	0.563	0.437	0.614	0.488
6	0.534	0.409	0.614	0.547
7	0.525	0.402	0.633	0.582
8	0.526	0.403	0.595	0.543
9	0.522	0.399	0.614	0.570
10	0.504	0.385	0.614	0.605

Table 7: Experiment 3 on the contextualized threshold combination to change between each time-step and the dynamic threshold

Window size	AUC	Precision	Recall	FPR
1	**0.799**	**0.831**	**0.684**	**0.086**
2	0.728	0.633	0.709	0.254
3	0.730	0.599	0.785	0.324
4	0.691	0.556	0.753	0.371
5	0.690	0.553	0.759	0.379
6	0.697	0.554	0.785	0.391
7	0.677	0.567	0.671	0.316
8	0.684	0.571	0.684	0.316
9	0.680	0.561	0.696	0.336
10	0.670	0.551	0.684	0.344

Table 8: Experiment 4 on the contextualized threshold in combination with the dynamic threshold

Window size	AUC	Precision	Recall	FPR
1	**0.870**	**0.836**	**0.842**	**0.102**
2	0.754	0.647	0.766	0.258
3	0.713	0.646	0.646	0.219
4	0.692	0.574	0.709	0.324
5	0.693	0.583	0.690	0.305
6	0.698	0.614	0.646	0.25
7	0.699	0.587	0.703	0.305
8	0.693	0.592	0.671	0.285
9	0.688	0.596	0.646	0.270
10	0.672	0.561	0.665	0.320

(12). However, the result in Table 6 demonstrates that it did not significantly change except in a window size equal to 1 where the precision increased to 0.88 and the recall to 0.696. The optimal window size for this experiment was 1 because it had a high rate of precision and recall. Also, the FPR was the lowest. Unfortunately, the AUC result decreased by 4 % on average from the previous experiment. This experiment supported our assumption that our data contained contextualization or anomalies, which required a specific context to support the detection. This assumption led to our experiments in which we combined the dependencies between time-step with the context for data point behavior as shown in Table 7.

From the result, combining these approaches improved the AUC by 17% versus the first experiment and 20% for the second experiment. The experiments also show that the window size of 1 was suitable for detecting anomalies and it had the highest precision and AUC compared to that of the other window sizes. It also shows that if the window size increases, then it degrades the detection performance because each data point does not have to depend or support each other where the approach takes the mean anomaly score within the window to define the threshold. If the majority of the data points in the window exceeded the threshold, then they were classified as anomalies; otherwise, they were classified as regular transactions. However, the recall still needs to be improved and was a sign that the data point was not influenced by the previous time-series. Therefore, we then performed an experiment where we removed the dependencies of time-step (t) and time-step (t-1), using Equation (13) from our model.

The results are shown in Table 8. This proved of our assumption that the data point of the previous time-step did not have any influence on the current prediction time-step where the precision and recall were a significant improvement over the previous experiment. We discovered that as the size of the window was set to 1, it had the most efficient detection when compared to that of the other window sizes. This means that each data point independently occurred and could be considered as point not collective anomalies in this case. The suggested improvement of this model would be the factors to be considered. Anomalies could originate from various perspectives as our study focused on operational anomalies. Thus, in our future work, we would expand the capability to detect anomalies in more aspects and discover the root cause that influences anomaly occurrence during the process. In addition, the data points that appear with regular data points are also important and comprise one of the limitations of this study. The second limitation is the reliability, as the statistic thresholds are not always detecting abnormal patterns, if the value does not exceed the threshold. In another case, sometimes a false positive detection is caused when the value exceeds the threshold but it was not influenced to occur as an anomaly. Combining multi-factors together to evaluate the operational behavior was required. In our future work, we suggest combining the LSTM-contextualized dynamic threshold approach with the multi-level-density-based approach to fill the gap of this limitation and increase the model reliability. We can leverage the benefit of distance and statistical approaches together to improve the detection. Our assumption is that anomalies would not always result in a high prediction error and also occur as regular data points. This is the reason why the study was so significant, necessary, and comprehensive to find a feasible solution.

6 CONCLUSION AND FUTURE WORK

This paper presents and defines an essential challenge in detecting abnormal behavior in a fleet optimization process which leverages the benefits from innovative anomaly detection approaches. We demonstrated that the LSTM approach was suitable to predict operational management values. While addressing the challenge involved and the remaining research question associated with unlabeled multidimensional datasets, and their interpretability to detect anomalies from non-stationary data series with too many anomaly scenarios in urban logistics transportation, we also proposed a novel contextualization with dynamic threshold approach that does not rely on any labels that limit real-world data sources. The capability of this approach extended to detecting anomalies with multi-dimensional factors and identifying their root causes. We discovered that our data do not contain any dependencies between time-step; furthermore, a specific context was required to detect anomalies as our last experiment showed that the AUC, precision, and recall rate significantly increased to 0.870, 0.836, and 0.842 respectively. The essential factors for improvement and further evaluation have also been identified as we look to expand the capabilities of the proposed approach and implement its framework not only to logistical areas but for all areas that involve series or streaming of data to support and enable more reliable and efficient decision making.

ACKNOWLEDGMENTS

This research was supported in part by an Excellent Thai Student Scholarship from Sirindhorn International Institute of Technology, Thammasat University, the SIIT-JAIST dual-degree scholarship program and the US Office of Naval Research Global under grant no. NG2909-19-1-2013. We would like to thank our colleagues from Jirajareon Co., LTD. who provided insight and expertise that greatly assisted our study.

REFERENCES

[1] Wathiq Laftah Al-Yaseen, Zulaiha Ali Othman, and Mohd Zakree Ahmad Nazri. 2017. Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. *Expert Systems with Applications* 67 (2017), 296–303. https://doi.org/10.1016/j.eswa.2016.09.041

[2] Tsatsral Amarbayasgalan, Bilguun Jargalsaikhan, and Keun Ryu. 2018. Unsupervised Novelty Detection Using Deep Autoencoders with Density Based Clustering. *Applied Sciences* 8, 9 (2018), 1468. https://doi.org/10.3390/app8091468

[3] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. 2018. Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection. *Iclr2018* 2016 (2018), 1–13. https://doi.org/10.1002/dvdy.21778

[4] Lo ÌĹÄśc Bontemps, Van Loi Cao, James McDermott, and Nhien-An Le-Khac. [n. d.]. Collective Anomaly Detection based on Long Short Term Memory Recurrent Neural Network. ([n. d.]).

[5] Andrea Borghesi, Andrea Bartolini, Michele Lombardi, Michela Milano, and Luca Benini. 2018. Anomaly Detection using Autoencoders in High Performance Computing Systems. Ml (2018). https://doi.org/arXiv:1811.05269v1

[6] VARUN CHANDOLA, ARINDAM BANERJEE, and VIPIN KUMAR. 2009. *Anomaly Detection: A Survey.* Technical Report September. https://doi.org/10.1089/lap.2006.05083

[7] AKHILESH GANTI. 2019. Weighted Average. https://www.investopedia.com/terms/w/weightedaverage.asp

[8] Luigi De Giovanni, Nicola Gastaldon, Massimilano Losego, and Filippo Sottovia. 2018. Algorithms for a Vehicle Routing Tool Supporting Express Freight Delivery in Small Trucking Companies. *Transportation Research Procedia* 30 (2018), 197–206.

[9] Prasanta Gogoi, D. K. Bhattacharyya, B. Borah, and Jugal K. Kalita. 2014. MLH-IDS: A multi-level hybrid intrusion detection method. *Computer Journal* 57, 4 (2014), 602–623. https://doi.org/10.1093/comjnl/bxt044

[10] Nathalie Helal, FreÌĄdeÌĄric Pichon, Daniel Porumbel, David Mercier, and EÌĄric LefeÌĂvre. 2018. The capacitated vehicle routing problem with loading constraints. *International Journal of Approximate Reasoning* 95 (2018), 124 –151. https://doi.org/10.1016/j.ijar.2018.02.003

[11] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. 2018. Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding. In *KDD 2018, August 19-23, 2018, London, United Kingdom.* 387–395.

[12] Doyup Lee. 2018. Anomaly detection in multivariate non-stationary time series for automatic DBMS diagnosis. In *Proceedings - 16th IEEE International Conference on Machine Learning and Applications, ICMLA 2017*, Vol. 2018-Janua. 412–419. https://doi.org/10.1109/ICMLA.2017.0-126

[13] Canhong Lin, K.L. Choy, G.T.S. Ho, S.H. Chung, and H.Y. Lam. 2014. Survey of Green Vehicle Routing Problem: Past and future trends. *Expert Systems with Applications* 41, 4 (2014), 1118–1138. https://doi.org/10.1016/j.eswa.2013.07.107

[14] Steven Mascaro, Ann E. Nicholso, and Kevin B. Korb. 2014. Anomaly detection in vessel tracks using Bayesian networks. *International Journal of Approximate Reasoning* 55, 1 (1 2014), 84–98. https://doi.org/10.1016/j.ijar.2013.03.012

[15] Nawaz Mohamudally and Mahejabeen Peermamode-Mohaboob. 2018. Building An Anomaly Detection Engine (ADE) For IoT Smart Applications. In *The 15th International Conference on Mobile Systems and Pervasive Computing.* 10–17.

[16] Sahar Selim, Mohamed Hashem, and Taymoor M. Nazmy. 2011. Hybrid Multi-level Intrusion Detection System. *International Journal of Computer Science and Information Security* 9, 5 (2011), 23–29.

[17] Hongchao Song, Zhuqing Jiang, Aidong Men, and Bo Yang. 2017. A Hybrid Semi-Supervised Anomaly Detection Model for High-Dimensional Data. *Computational Intelligence and Neuroscience* 2017 (2017), 1–9. https://doi.org/10.1155/2017/8501683

[18] Yiyong Xiao and Abdullah Konak. 2017. A genetic algorithm with exact dynamic programming for the green vehicle routing & scheduling problem. *Journal of Cleaner Production* 167 (2017), 1450–1463.

[19] Giancarlo Zaccone. 2016. *Getting Started with TensorFlow.* 143–147 pages.

[20] Shuangfei Zhai, Yu Cheng, Weining Lu, and Zhongfei Zhang. 2016. Deep Structured Energy Based Models for Anomaly Detection. 48 (2016). https://doi.org/10.1109/IJCNN.2000.861302

[21] Chong Zhou and Randy C. Paffenroth. 2017. Anomaly Detection with Robust Deep Autoencoders. (2017), 665–674. https://doi.org/10.1145/3097983.3098052