Anomaly Detection Using Unsupervised Learning

Shantanu Dave
11130387
Communication Systems & Networks(M.Sc.)
TH Köln
Köln, Germany
shantanu.dave@smail.th-koeln.de

***Abstract*—As we are moving towards the Industry 4.0 era where Artificial Intelligence(AI) and the Internet of Things(IoT) are crucial and integral parts of the revolution. In this transition phase from manual to the automation of work using different machines, sensors are a very important component and they play a vital role in the setup. The connectivity and flow of data/ information between sensors and devices leads us to witness rapid growth of time-based data are known as time series. In this project we will be implementing the techniques and applications of machine learning and statistical analysis, getting familiar with pandas, matplotlib, NumPy and various other libraries using Python on available sensor data from industries and extract useful information and make it possible to detect outliers and perform conditional monitoring which in-turn will help in reducing cost, optimizing manual labour capacity, increase productivity, availability, reliability and keep downtime minimum[7].**

The main aim of the Research Project is to develop online multivariate analysis tool which fetches the data, impute the missing data, eliminates outliers and non- compliant data, perform unsupervised learning and inform the user in case of abnormality ie, out of control situations.

I. Introduction

Many application domains are influenced by indoor/outdoor air quality, ranging from process industries to corporate offices. The factories tend to be a significant source of pollutant releases to the atmosphere in Europe. According to the study, the air is up to 100 times more toxic in factories than outside. Air quality within industry carries a lot of details about the machinery and can also affect machine endurance. Specifically considering the process industry, the machines are acquainted with the diverse fragments of air quality comprising ozone, dust particles, volatile organic compounds and numerous gases such as carbon, sulphur, nitrogen etc. These can have an effect on machine components eg rusting or decay. If we gather all of these data we will obtain a significant volume of multivariate time-stamped data sequences. Analysis of such data will help us understand the nature of the system and the processes involved that generate the time series. It helps to further predict the future trends, behaviours of systems and detects outliers.

For instance, in old times the manual labours were appointed in the plant to monitor the machines. They were capable of detecting machines and surroundings abnormal behaviour before it turned out to be a serious issue. In particular, they also tracked the procedures. They recognized several different odours so they used them to distinguish between normal and abnormal instances. However, with the availability of a lot of controlling software, in recent times as we are advancing towards automated industry solutions, manual labour is drastically reduced by a factor of four in the plant and increased by two in the control room depending on the safe automation, which indeed is not completely safe. Consider, for example, a boiler in which there is a steady flow of water and at the same time a leakage occurs. In this situation, the control room monitors the level of water and assumes that there is a slow rise in water level, but there is no data about the leakage occurring which indicates about the unsafe measure.

Now suppose we have different multilayered sensors collecting various data readings every minute and if there is a fault, then the sensors will start to behave inconsistently and the signals they convey will also get uncanny and by this, anything which is not normal will be detected and the control room will be alarmed. This is achievable by developing an intelligent application where we use artificial intelligence to perform similar cognitive functions like a human brain. Key specifications will include high computing power, the ability to use numerous Artificial Intelligence and Machine Learning Packages, Wi-Fi access to connect to the Access point created by Sensor Bot, digital outputs and the flexibility to use a Graphical User Interface (GUI)

To accomplish the project objectives, we used one Sensor Bot (small microcontroller powered sensor kit using Arduino and ESP 8266 as a Wi-Fi module and numerous sensors) The sensor bot collects several sensor data, like noise value, light, temperature, vibration in multiple directions (x, y, z), dust measurements, Co2, TVOC, humidity using different sensors to acquire air quality information. For computational purpose, we used powerful Raspberry Pi 3B+. It comes with robust Processor specification 1.2 GHz Quadcore Broadcom 64bit CPU, associated RAM is 1 GB, It also has 40 GPIOs pins, availability of Ethernet Port, availability of full size HDMI, HDMI Storage is possible through Micro SD, Wi-Fi: BCM43438 wireless LAN and last but not the least we used Python programming language for scripting. The considerable Reasons for opting Python were its Simple and consistent nature, an extensive selection of libraries and frameworks. Python is interface Independent, providing excellent community support and it is popular in today's trend.

II. Preliminaries

This part of the paper deals with the terminologies used in further steps.

A. Anomaly

Anomaly is a pattern in the data which does not follow the expected behaviour. It is also known as "Outliers","Exceptional cases" etc.

There are three types of anomalies: point, contextual and collective. A data point is considered a point anomaly when it is far away from other points. On the other hand, contextual anomaly varies on the context and collective anomaly is when a set of information is not different from the entire data set[20]

B. Time Series

Time series is a sequence taken at a successive equally spaced interval in time. In our case, every 60 seconds. Thus, it is a sequence of discrete-time data. Time series data generally emerge when monitoring industrial processes or while measuring attributes of air quality. The time series analysis of temporal data indicates that with respect to variation in time, data points may have an underlying structure (e.g. autocorrelation, some general pattern or cyclical variation) that needs to be considered[17].

Applications: The use of time series models to detect anomaly on our temporal data is in two sections. Initially, obtain an understanding of the structure of the data, how they are correlated and then fit a machine learning model and proceed to the detection of outliers. Whether univariate analysis is a good pick, or better results are obtained through multivariate analysis?

C. Univariate Time Series

Univariate analysis of the time series relates to a time series comprised of only one observation linearly recorded over equal time increments. Some case studies are the temperature readings taken at different intervals of time in a city used for forecasting next values. Although the normal understanding of univariate time series data set is given as a single column of numbers but in computability, time is an implicit variable in the time series. In our use case, we performed univariate analysis on different sensor data individually. If the data are equispaced, the time variable does not need to be explicitly given. Thus, we used sequence/index numbers[19].

Time	Sequence Number	value 1
19/08/2019 22:39	1	529
19/08/2019 22:40	2	529
19/08/2019 22:41	3	528
19/08/2019 22:42	4	528
19/08/2019 22:43	5	529

Figure 1. Univariate Data were taken every 60 sec

D. Multivariate Time Series

As the name suggests, there are more than one time-dependent parameters in a multivariate time series. 10 different sensor values were used in our case. Each variable depends not only on its past values but it also shows some relation to other variables. In Excel, we employed a stepwise regression methodology to understand the concept in-depth and to learn about the inter-variable correlation[19].

Time	Sequence Number	value 1	value 2	value 3	value 4	value 5
19/08/2019 22:39	1	529	374	668	0	0
19/08/2019 22:40	2	529	388	668	0	0
19/08/2019 22:41	3	528	393	800	6	0
19/08/2019 22:42	4	528	401	680	0	0
19/08/2019 22:43	5	529	403	679	0	0

Figure 2. multivariate data with 5 values taken together every 60 seconds

E. Why Unsupervised Learning?

It is a machine learning technique responsible for determining the unknown patterns in the dataset without pre-existing labels. Deep Learning's amazing success has been accomplished mainly through supervised machine learning, but large "labelled" datasets are required to train these algorithms. In our case, however, we look forward to developing an Artificial Intelligence, that reacts with human senses ' ability. The human way of reacting to unexpected events is largely unsupervised and real-time. Given that the dataset that motivates this research is entirely unlabeled, this project aims to contribute to improving unsupervised anomaly detection.

III. Related work

Anomaly detection or finding any suspicious activity(outliers) are the fascinating topics in various papers in the recent years. Amongst them, Tung Kieu, Bin Yang, Christian S. Jensen have done detailed research of Outliers on time-series data (2018)[2] using statistical methods and also autoencoders; Another jaw-dropping survey related to time series data in Aggarwal C., Outlier Analysis(2016)[6] where he has completely focussed on theoretical aspects of Outlier detection. However, these papers do not focus on Unsupervised algorithms, which this paper supports as an alternative to the others.

IV. Brain Tasks

A. Fetching the Data

Using the WiFi module ESP8266, the data collected from the sensor bot is sent and uploaded to a Web server. The Sensor Bot is responsible for collecting data every 60 seconds and the concurrent function is to upload the data to the web server. The first aspect of brain tasks is to extract the data from the webserver and save it in the device or database in a local directory. The web scraping methodology is being used to retrieve data from the website. Web Scraping is the process of downloading and extracting valuable information from websites. Several other empirical processes and decisions are dependent on these results.

The process of collecting IoT data from industries in real-time will certainly help to optimize their processes. An emerging subdivision of technology is Industrial IoT (IIoT). In a very short span of time, it has received considerable popularity. Industrial plants employ sensors to measure the fitness of their conveyor belts and machinery. To foresee any system malfunction, you can obtain data from those sensors and connectors. Scrapping real-time data is useful as it can help to take proactive and mitigating action to avoid damages and therefore boost the industry's efficiency and performance.

For this process, numerous python packages are available ranging from Pattern, Scrapy, Mechanize, Beautiful Soup, Requests etc. Python is open source and has extensive libraries for almost everything There are several approaches in this process to collect the data from a web server, but the Beautiful Soup framework and Requests in python are a commonly used tool. Beautiful Soup is basically referenced from a song which is cited in high school literature [24]. It is a Python library that allows quick responses to web scraping operations. Beautiful Soup's latest version(Beautiful Soup 4) which is compatible with both Python 2.7 and Python 3, creates a parse tree from structured HTML and XML documents.

Beautiful soup module is used to extract data from both HTML and XML format, offering simple and easy ways to navigate, scan, and update data during the web-scrap process. Requests module is used to send and receive HTTP requests and make human-readable data available. You can integrate your Python programs with web services through the use of the Requests module.

It should be noted that the webserver data is being updated every minute and is fetched via time function or crontab using raspberry pi. In both instances, we run the python script every 60 seconds. Crontab is a Raspberry Pi special function that allows you to run any script according to the directive without using it manually. The key reasons for using Python are: it is one of the best and simplest to go programming language, it provides significant community support and use cases, comprise n number of libraries, and is mainly platform independent In the end, we stored the text we obtained in a CSV document. We therefore continuously collected data for 2 weeks to make our application more robust

B. Data Preprocessing

The data we received from the sensor bot is in raw format. When we deal with numerous amounts of data then there is a high probability of dealing with incorrect data due to sensor mistake or network issues. Data pre-processing is the method of cleaning/correcting the raw data into an organised format. The data we fetched from the sensor kit was partially random and comprised of missing data, false values. They are commonly ciphered as NaNs (not a number) or false values. Hence it is necessary to clean the data for further use. Data pre-processing has to be performed in such a way that we can implement any sort of Machine learning algorithm to the cleansed data.

Testing a model with plenty of missing values degrades the efficiency of machine learning capabilities. It is, therefore, necessary to understand the data flow, data type, etc. Steps to perform Data Pre-processing: import the appropriate libraries, read the data from the file, check for missing values and handle them, also find categorical data and convert to numerical values, Standardize data. Transformation and Data splitting.

Ways of dealing with missing data: the first way is to do nothing. Second, remove the missing data rows. Though, a lot of useful information gets lost when there are large numbers of missing values. Instead of eliminating the data points, the best method is to impute the missing values. It ensures the missing values are predicted from the previous values and other correlated values from available data. There are three types of missing data available and they are as follows:
(1)Missing Completely at Random(MCAR): In this case, the data presence or absence is completely independent of the variables or parameters under observation[15].
(2)Missing in Random(MAR): In this case, the data available is not arbitrary and may be related to the parameter with complete information[15].
(3)Not missing at Random(NMAR): this scenario is coped with when the missing values do not represent any of the two subgroups (MCAR or MAR) above[15].
Additionally, we concentrated on three different imputation strategies in this experiment; Mean / Median, Mice and k-NN, which are further explored in more depth.

a)Mean/Median: This method is carried out by performing the mean of non-missing values in a column and replacing it with the missing value (Figure1 and 2).
Advantages: it works well with a small data set and is easy to execute.
Disadvantages: it is not possible to correlate the data since it is centred on the column, nor does it account for the uncertainty of the imputation[14].
For our dataset, this imputation failed to give the expected result as our dataset vary with function of time and many other parameters.

Figure 3. Mean Imputation

(B)Multiple imputations using MICE (Multiple Imputation by Chained Equations): most methods generally involve the single imputation. MICE algorithm runs several regression models at once and every missing value is imputed through the non-missing values (Figure 3). This functional algorithm is based on three distinct phases. 1.Imputation: imputation of incomplete values(m) from data which is not missing
There is no uncertainty when simulating random draws since it relies on the distributed data[14]. 2.Analysis: Analyse each of the data sets generated in m.
3.Pooling: Impute the findings of the algorithm m into the

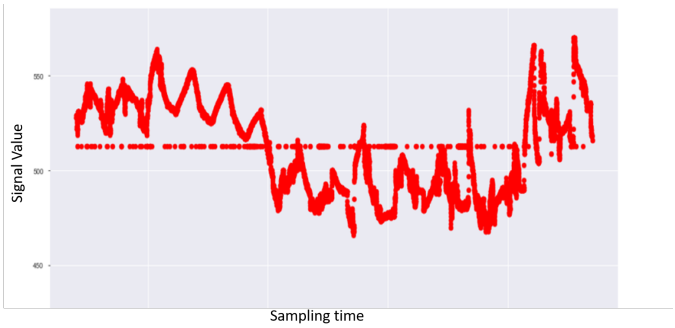

Figure 4. Mean Imputation Result

end result
This is a new technique which is highly used for several different types of dataset but again through MICE imputation as well we could get the expected outcome.

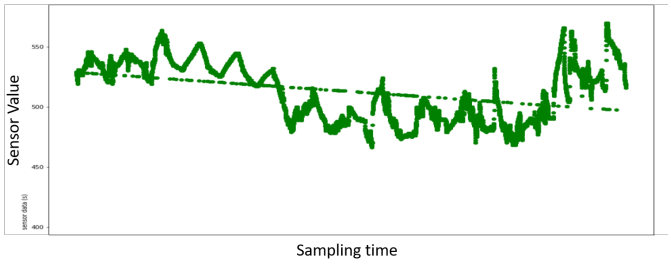

Figure 5. Mice Imputation Result

C) Imputation by k-NN: Popularly known as the k-th nearest neighbour. The algorithm predicts missing values on the basis of similar characteristics (Figure 4). The idea behind using KNN for missed values is that a point value may be approximated by the nearest point values depending on past values and on other variables. Firstly, a mean imputation is performed and then KD tree is constructed with the resulting values. The nearest neighbours are computed further using this KD tree.The value of K is very important and deciding factor, there is not fix value available. The idea is to get the best possible result with the minimum k value. we took k = 3 in our case. Once it's done it takes its weighted average and then imputes the new data [22].
In our dataset it worked perfectly and we got the desired result. It also shows that the different variable are correlated and it is possible to get good processed data if we use knn imputation in time-series data.

V. Experimental Setup

In this project, we have tried to implement both Statistical and Machine learning methods for time-series data.

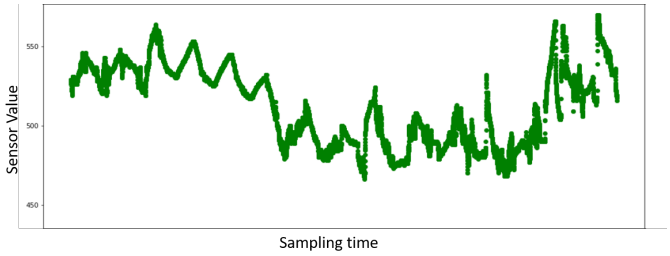

Figure 6. k-NN Imputation Result

A. Statistical Analysis

Regression Analysis is responsible for predicting the correlation between two or more variables in Statistical Modelling. Primarily, we have two kinds of variables; dependent and independent. The dependent variable is the one you would like to predict or forecast about. Independent variables are those that could affect the dependent ones. Application of regression analysis helps you understand how the dependent variable differs when one of the independent variables shifts, which enables you to statistically evaluate which of those variables affects[16].

SUMMARY OUTPUT	
Regression Statistics	
Multiple R	0,999966115
R Square	0,999932232
Adjusted R Square	0,999885602
Standard Error	4,224245261
Observations	21466

Figure 7. Linear Regression Analysis

Theoretically, the framework of regression analysis is based on the sum of squares, a statistical method of determining the distribution of data points. The model aims to get the smallest sum of squares practicable and make a distinction that comes closest to the results. R Square. It is the determination coefficient, which is used as an indicator of the quality of fit. It demonstrates how many points are falling on the line of regression. The value of R2 is computed from the total sum of squares, more accurately it is the sum of the squared deviations from the mean of the raw data[23]. We tried to inspect the outliers present in temperature in our dataset. We added a few nonlinearities where the standard values were drifted. In the first step after regression, we searched for P values for variables which were comparatively less, then removed them to check R square value and then followed again the same steps. R2 (R square) is 0.9999 (rounded to 4 digits) in our example which is excellent result. It means that the regression

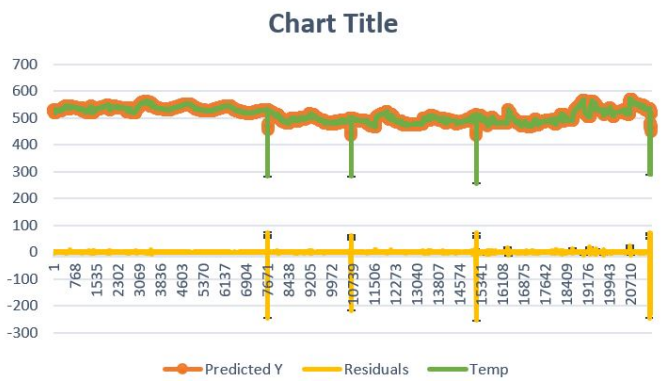

Figure 8. Linear Regression Result

analysis model fits 99 per cent of our values. Typically, R Squared is considered a decent fit of 95

B. Machine Learning Algorithms

1) Isolation Forest: : mainly responsible for detecting the anomalies in multidimensional space and it is almost similar to the Random forest algorithm. In general, it is not easy to distinguish between normal and anomalous data without labelled data. In this case, we're trying to find anomalies using the Isolation Forest. We used 12 metrics of sensor data including date and time in the data set. We focused on temperature data in particular. In the event of isolation forest, all values in the data set are segregated. Instead of monitoring normal data points, it specifically detects anomalies. Isolation forest is built on the basis of decision trees. It is important to acknowledge that anomalous points can be separated in a few stages while normally linked points can take significantly additional steps to be separated[18, 20].

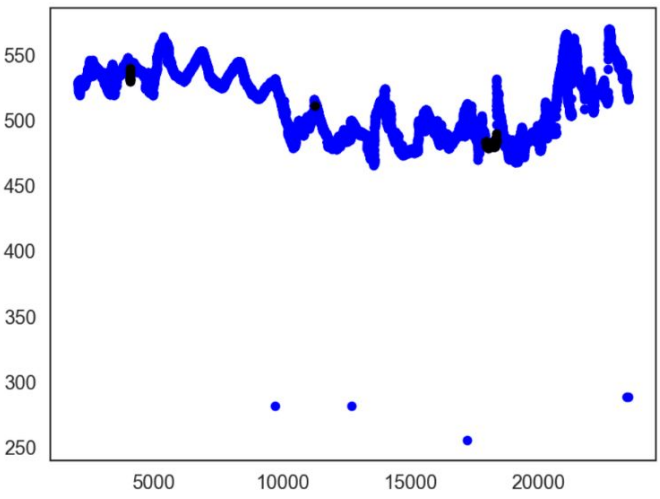

Figure 9. Result of Isolation Forest

The value of the contamination that refers to the percentage of outliers present in the data is yet another important factor in isolation forests. We used the isolation forest of sklearn and have tried to explore anomalies in our high volume data set. With a hit and trial method, we tried to find the value of contamination. Generally, when anomalous points are to be identified, isolation forest is regarded as one of the prominent unsupervised learning methods. But unfortunately, we have had to increase the contamination in our case in order to detect all the anomalous points. But there were also many "false positive" cases that had been predicted.

2) One class SVM: A further interesting approach to anomaly detection is through state vector machines[14]. SVM was initially discussed by Vapnik in his research [11]. SVMs is an approach to solve classification problem addressing it to be a quadratic optimization problem. SVM architecture lets us identify the right vector machine based on our data in such a way that machine reliability for training data is enhanced which in fact is very important for the test case, and the anomaly can be later preciously identified. State vector machine typically works for supervised problems where we already have the data set labelled. In supervised problems, SVM is generally two-class based.

It was later adapted to one-class SVM by methods outlined in [12]. One-class SVM is used for unsupervised learning model where we train only the normal data. The negative examples in general. The model learns the limits of these points and when we detect any sort of noise in the positive data which means if we find any data point which lie outside the boundary then we term it as outliers. The one class SVM is represented as follows [12] [13]:

$$f(x) = \begin{cases} +1, if\ x \in S \\ -1,\ if\ x \in \overline{S} \end{cases}.$$

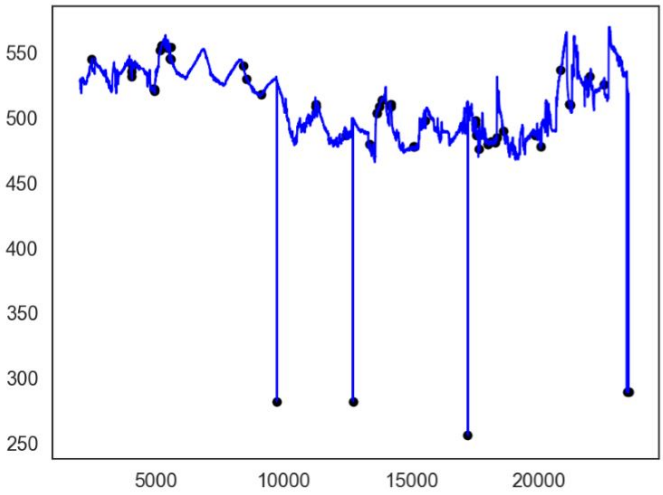

Figure 10. Result of One-Class SVM

The two important things to consider are the nu parameter and the gamma parameter. The nu parameter specifies how many outliers we anticipate in the data set and the gamma parameter measures the smoothing of the contour lines. We change the value of the Gamma factor by hit and trial method to see how close we can get with minimum error margin[20].

3) RNN: A different approach to detect anomaly is Recurrent Neural Networks (RNN)[20]. They differ from the typical neural network as they use the output, we received from hidden layer neurons as feedback and provided the input to the next neurons. The previous records are thus used to grasp and predict the next sequence which we can then test to detect an anomaly

Long Short-Term Memory (LSTM) enables the large-term processing of data dependencies [10]. The LSTM framework comprises of a collection of recurrently connected structures termed as memory blocks. That block has one or more self-connected neurons each with its own cell state. The memory block is constructed all across the neuron cells which, by using an identity function and always having incoming unit weights, can ensure continuous error flow into them. Therefore, these units resolve the regression problem [9],[10] widely seen in conventional recurrent neural networks.

In this particular approach of our case, we equate the results estimated to the actual data. We can come to the conclusion that the data point is anomalous when the next predicted point is on a large distance with the real data point. Collection, size of block sequence and how many points we are predicting for anomaly detection are very crucial parameters to get relevant data comparison.

We consider an anomaly when the next data points are distant from RNN prediction. In our particular case, We used it on univariate data. we focussed majorly on temperature value. we made RNN learn the sequence from the last 50 values, and we predict just the 1 next value. This process continued for the whole set of data.

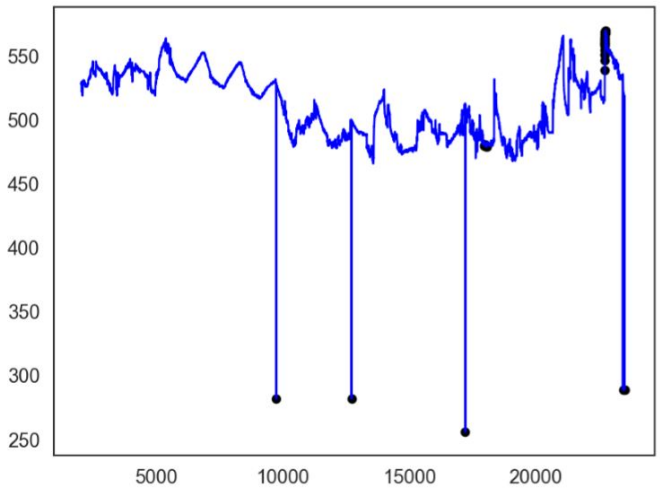

Figure 11. Result of Recurrent Neural Networks

VI. Conclusion

In addition to the above algorithms, we referred to many other algorithms during the research as well such as K means clustering methods which are very popular in unsupervised machine learning. We were not able to get close to the desired result so we didn't mention them in depth. We have had very varied results based on our experiments. Technically it is difficult to determine which algorithm is superior to the others. For example, in the standard regression model, there is no data-cleaning, i.e. pre-processing of data. However, we used the cleaned data and evaluated statistics. It gave us very comprehensive information about the correlation between several parameters in our dataset and we were able to extract information that worked for us while experimenting on other machine learning algorithms. Isolation forest is perceived to be among the prominent machine learning algorithms and it is strongly in trend when investigating outliers but it explicitly failed to detect point anomalies for our dataset. We may come to one major conclusion that Univariate Analysis is of no benefit when dealing with a broad dataset where data points are closely linked to each other. So, Multivariate analysis should be preferred to detect anomaly in time series dataset where data points indicate deviation when altering other data points. Lastly, When we used one-class SVM and RNN we got impressive results. Those two could be used as a base point for further research.

The primary objective of this research was on detecting anomalies. In the near future, we will intensify the system's transitions and behaviour based on time. Since we are working on unsupervised models, the challenge is encouraging to get into more detail without the labelled data set. In analyzing data from the process industry, certain data points, which are abnormal at a specific time and day, can be normal on another day and vice versa. The emphasis would be on identifying hidden neural network chains that can lead us to designated results using libraries such as Keras and TensorFlow, as well as creating a human-machine interface to track and analyze real-time data.

References

[1] M. Munir, S. A. Siddiqui, A. Dengel and S. Ahmed. *DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series*. IEEE Access, vol. 7, pp. 1991-2005, 2019.

[2] Kieu, Tung and Yang, Bin and Guo, Chenjuan and S.Jensen, Christian. *Outlier Detection for Time Series with Recurrent Autoencoder Ensembles*. IJCAI '19, 2019.

[3] A. Cook, G. Mısırlı and Z. Fan. *Anomaly Detection for IoT Time-Series Data: A Survey*. IEEE Internet of Things Journal.

[4] H. Izakian and W. Pedrycz. *Anomaly detection in time series data using a fuzzy c-means clustering*. 2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), Edmonton, AB, 2013, pp. 1513-1518.

[5] H. Wu. *A survey of research on anomaly detection for time series*. 2016 13th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, 2016, pp. 426-431.

[6] J. Qi, Y. Chu and L. He. *Iterative Anomaly Detection Algorithm Based on Time Series Analysis*. 2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Chengdu, 2018, pp. 548-552.

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

[8] S. Hochreiter, J. Schmidhuber. *Long Short-Term Memory*. Neural Comput., vol. 9, no. 8, pp. 1735-1780, Nov. 1997.

[9] S. Hochreiter, Y. Bengio, P. Frasconi, J. Schmidhuber. *Gradient Flow in Recurrent Nets: the Difficulty of Learning Long-Term Dependencies*. 2001.

[10] Y. Bengio, P. Simard, P. Frasconi. *Learning Long-Term Dependencies with Gradient Descent Is Difficult*. Trans. Neur. Netw., vol. 5, no. 2, pp. 157-166, Mar. 1994.

[11] B. E. Boser, I. M. Guyon, and V. N. Vapnik. *A training algorithm for optimal margin classifiers*. In 5th Annual ACM Workshop on COLT, pages 144-152, Pittsburgh, PA, 1992.

[12] J. Shawe-Taylor A. J. Smola B. Schölkopf, J. Platt and R. C. Williamson. *Learning Long-Term Dependencies with Gradient Descent Is Difficult*. Estimating the support of a high-dimensional distribution. Neural Computation, 13:1443-1471, 2001.

[13] Y. Wang, J. Wong and A. Miner. *Anomaly intrusion detection using one class SVM*. Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, 2004., West Point, NY, 2004, pp. 358-364.

[14] dawson, c. (2020). Outlier Detection with One-Class SVMs. [online] Medium. *Available at: https://towardsdatascience.com/outlier-detection-with-one-class-svms-5403a1a1878c*.

[15] Badr, Will. (2019). 6 Different Ways to Compensate for Missing Values In a Dataset. [online] Medium. *Available at: https://towardsdatascience.com/6-different-ways-to-compensate-for-missing-values-data-imputation-with-examples-6022d9ca0779*.

[16] V. Flovik. (2019). How to use machine learning for anomaly detection and condition monitoring. [online] Medium. *Available at: https://towardsdatascience.com/how-to-use-machine-learning-for-anomaly-detection-and-condition-monitoring-6742f82900d7*.

[17] M. Tadayon,Comprehensive Analysis of Time Series Forecasting Using Neural Networks, DeepAI. [Online]. *Available: https://deepai.org/publication/comprehensive-analysis-of-time-series-forecasting-using-neural-networks*.

[18] A. Krishnan,Anomaly Detection with Isolation Forest Visualization. [Online]. *Available: https://towardsdatascience.com/anomaly-detection-with-isolation-forest-visualization-23cd75c281e2*

[19] S. Parera,Introduction to Anomaly Detection: Concepts and Technique, My views of the World and Systems. *Available: https://iwringer.wordpress.com/2015/11/17/anomaly-detection-concepts-and-techniques/*.

[20] V. Ambonati,Unsupervised Anomaly Detection, Kaggle.com,[Online]. *Available: https://www.kaggle.com/victorambonati/unsupervised-anomaly-detection*.

[21] Galante, Luca Banisch, Ralf. (2019).A Comparative Evaluation of Anomaly Detection Techniques on Multivariate Time Series Data. 10.13140/RG.2.2.18638.72001.

[22] Y. Obadia, The use of KNN for missing values,Medium, [Online].*Available:https://towardsdatascience.com/the-use-of-knn-for-missing-values-cf33d935c637*

[23] Amral, N. Ozveren, Cuneyt King, David. (2007).Short term load forecasting using Multiple Linear Regression. Proceedings of the Universities Power Engineering Conference. 1192 - 1198. 10.1109/UPEC.2007.4469121.

[24] Mitchell, R. (2018). Web scraping with Python.