Vrije Universiteit Amsterdam

Bachelor Thesis

Unsupervised learning for IoT Anomaly Detection in Smart Buildings

Author: Andres Latorre 2708338

1st supervisor: Kees Verstoep
2nd reader: Mauricio Verano Merino

A thesis submitted in fulfillment of the requirements for the VU Bachelor of Science degree in Computer Science

April 17, 2024

Contents

Abstract

As sensor networks proliferate across a wide range of applications, efficient anomaly detection techniques have become increasingly essential. This project presents a comprehensive approach for detecting anomalies in multivariate real-time series data generated by sensors situated in various rooms of Vrije Universiteit's NU building. The proposed methodology leverages data clustering, data visualization techniques, and room-type specific analysis to not only identify deviations from normal patterns but also categorize them, distinguishing between various types of anomalies. The sensor network setup includes various room types, such as meeting rooms and two-person rooms, each exhibiting unique patterns and requirements that need customized analysis.

1 Introduction

In a wide range of application sectors, the Internet of Things (IoT) has made extensive digitization possible. Currently, there are more than 12 billion IoT devices, and by 2030, the number of deployed IoT devices is expected to reach 125 billion [1]. The new opportunities offered by the IoT are resulting in what experts call the "Fourth Industrial Revolution," also called Industry 4.0. This new type of industry is characterized by the integration of data, artificial intelligence algorithms, and IoT devices to create an intelligent and efficient industrial ecosystem. This thesis focuses on multivariate time series data, particularly those derived from IoT sources. Time series analysis consists of extracting information from points arranged in chronological order. One common objective is to discover correlations between time series. Focusing on multivariate time series data from IoT devices allows correlations between various systems variables, revealing IoT technology's complex interactions. Many other objectives explain the popularity of time series analysis, such as the search for trends, cycles, seasonal variations, or the detection of unusual behavior [2].

The dataset used for this project is unlabeled raw data from sensors that are positioned strategically in different rooms, each presenting different conditions. Therefore, the use of unsupervised learning algorithms steps up as a feasible way of detecting anomalies in the provided data ecosystem. Since they are designed to deal with unlabeled data, they are able to learn and identify interesting patterns from the data's own internal structure, meaning that they can also be used to point out anomalous patterns even when the labels are unknown. However, there are several difficulties in developing an automated model in an IoT environment. It is challenging and not always possible to define and categorize all types of anomalous data correctly. In many fields, the notion of normal behavior is constantly changing and evolving. Furthermore, data often contains noise, which can result in uninteresting and, therefore, unwanted anomalies. The focus of this project is not solely on anomaly detection but on extracting actionable intelligence. Through unsupervised learning methodologies, we seek not just anomalies but the nature within the patterns, offering a glimpse into the dynamics of each room. This paper will address the following research questions:

1. How should data be processed in order to get relevant information from the underlying raw data?

2. Is it possible to find an explanation for the different patterns and anomalies embedded in the sensor readings?

3. To what extent can we relate patterns and anomalies in the sensor data of the building to patterns in the usage of its facilities?

2 Related Work

Many anomaly detection methods have been developed for various application sectors in the last few years. A recent study by [3] demonstrated the effectiveness of a multivariate analysis using six different unsupervised ML algorithms for time series. The algorithms used were: K-Means, DBSCAN, Gaussian mixture model, k-NN, PCA and Autoencoders. These algorithms offer diverse approaches to clustering and anomaly detection in multivariate time series data, each with its own strengths and weaknesses. The study applied these algorithms to meteocean data in hurricane season and monitoring data from dynamic industrial machinery. According to the experimental results, clustering techniques like C-AMDATS demonstrated a greater ability to identify and separate anomalous regions, demonstrating their potential to help experts efficiently label unsupervised machine learning algorithms on raw data.

Additionally, past literature [4] proposed a solution for occupancy detection with a limited number of non-intrusive environment sensors. The study utilized human activity detection models to convert raw environment data into general activities (e.g., door handle touch events and water usage events) and used machine learning-based classification algorithms to predict occupancy information from these activities. The results indicated that the Random Forest algorithm provided valid estimations of occupancy information with high accuracy and F1-score, even when the number of sensors was reduced to three. This highlights the potential of incorporating human activity models to improve the performance of machine learning-based classifiers and reduce the required number of sensors significantly.

A very similar project by Mikita Volakh[5] provides a classification of room occupancy status based on the same dataset used in this project. To find the most accurate prediction model, a comparative analysis between Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Convolutional Neural Network (CNN) architectures is conducted. To provide users the ability to monitor occupancy a KMeans classifier was used to create three clusters for each metric, establishing thresholds for occupancy statuses (green, orange, red). On the results, it was found that the CNN model does not perform as effectively as recurrent neural network models on the given dataset. Also, results indicate that, on average, tuning the models gave an improvement as seen for the GRU model which increased its accuracy by 5.93%.

3 Dataset Description and Preparation

The Vrije Universiteit's New University building in Amsterdam is the source of the data set used in this study. Specifically, the data was obtained from the upper floors of the building which includes different types of rooms for study. This dataset includes three types of rooms: offices, meeting rooms and labs. Also, rooms are fitted with different types of sensors but, for this project, only light, sound and CO2 sensors were used. From January 12, 2022, to March 15, 2023, the data was recorded within a ten-minute interval.

The data loading and preprocessing stage involves several key steps to guarantee the data's suitability for further analysis. First, files containing the raw sensor data are obtained from the specified storage location. These files include readings from a variety of sensors that have been placed throughout the building's various rooms. Therefore, a room has to be selected to later gather and analyze the sensor readings of this concrete location. To facilitate higher-level analysis, the data is resampled to hourly intervals with the idea that this might give more interesting "events" due to a larger timespan. To achieve this, a minute column was added to represent the 10-minute intervals within each hour (look at Figure 1). Therefore, the new dimension of the dataset is set to 36 as there are six sensors and six 10-minute interval columns. After resampling, the sensor readings are scaled to a standardized range to ensure uniformity and comparability across different sensors and sensor types. This scaling process uses the Min-Max scaling technique to transform the sensor readings to a predefined range, in this case, 0 to 100. This helps to handle outliers and especially to reduce the impact of individual sensor readings with the objective of normalizing the different types of sensor values.

		0	10	20	30	40	50
thingy052_eCO2	2023-03-15 15:00:00	5.9	5.7	6.0	5.2	5.7	5.2
thingy052_sound	2023-03-15 15:00:00	7.9	5.0	6.9	11.1	7.9	7.4
thingy052_color_g	2023-03-15 15:00:00	29.3	29.3	29.3	29.1	28.9	29.5
thingy052_color_b	2023-03-15 15:00:00	29.9	29.5	29.5	29.2	29.2	29.5
thingy052_color_c	2023-03-15 15:00:00	31.6	31.6	29.8	29.8	31.6	31.6
thingy052_color_r	2023-03-15 15:00:00	37.6	37.6	37.2	37.2	37.2	37.2

Figure 1: Extracted rows after data processing for all five sensor columns at a certain timestamp

4 Unsupervised ML algorithms

This section will review the concept and application of two well-known unsupervised ML algorithms for anomaly/pattern detection applied in this research. The analysis compares K-means and HDBSCAN(Hierarchical Density-Based Spatial Clustering of Applications with Noise) to determine which method best works for the anomaly detection task.

4.1 K-means Clustering

K-means clustering[6] is a popular unsupervised machine learning algorithm used for partitioning data into clusters based on similarity. By iteratively updating the centroids to minimize the within-cluster sum of squares (WCSS), the algorithm effectively assigns each data point to the closest centroid. Determining the optimal number of clusters is a crucial step in K-means clustering. For this purpose, two methods were employed, namely, the Elbow Method and Silhouette Score algorithms:

- Elbow Method: Evaluates the sum of squared distances within clusters (WCSS) for different values of K and identifies the point where the decrease in WCSS becomes less significant, indicating the optimal number of clusters.
- Silhouette Score: Measures how similar each data point is to its assigned cluster compared to other clusters. The silhouette score ranges from -1 to 1, with higher scores indicating better clustering.

To determine the desired number of clusters, the combined score is calculated as the average of the silhouette score and the change in WCSS (deltas) obtained from the Elbow Method. The optimal number of clusters is then selected based on the maximum combined score ensuring a balanced determination of K.

4.2 HDBSCAN Clustering

The HDBSCAN clustering algorithm[7] is a density-based algorithm. Unlike K-Means, it does not require that every data point is assigned to a cluster, since it identifies dense clusters. Points not assigned to a cluster are considered outliers, or noise. An algorithm that can effectively find distinct groups in a dataset and identify outliers is a valuable technique. In addition, HDBSCAN does not require specifying the number of clusters in advance, as it automatically determines the number of clusters based on the data's density distribution. To implement HDBSCAN clustering effectively, the following steps are undertaken:

1. Grid Search for Optimal Parameters: A grid search is performed to find the best parameters to find out the minimum cluster size and minimum number of samples for each cluster. This is done by iterating through different combinations of parameters and selecting the configuration with the highest silhouette score as the optimal values. This integration of grid search for parameter tuning increases the effectiveness of the model.

2. Construction of HDBSCAN Model: Using the optimal parameters obtained from the grid search the HDBSCAN model is constructed. The model is fitted to the data, and cluster labels are assigned to each data point.

3. Extraction of cluster centers: Extracting cluster centers is vital to further analyze and detect anomalies. Unfortunately, Python does not include a function to extract the cluster centers for HDBSCAN in comparison to K-means. The most effective process to get the cluster centers is by finding the representative points for each cluster. These representative points serve as centroids or prototypes of the clusters, providing insights into the characteristics of each cluster. This is done by iterating through all clusters and taking the mean of each feature within the cluster.

4.3 Model Comparison

To evaluate and compare the performance of the K-means and HDBSCAN clustering models for further usage in anomaly detection, two metrics are utilized, silhouette score used earlier for evaluating both models and the Davies-Bouldin Index(DBI). The DBI[8] is a popular measure to assess clustering performance by dividing clusters. DBI works by calculating the average value of each item in the data set. The value of each point is calculated as the sum of the compactness values divided by the distance between the two center points of the group as separation. The smaller DBI value indicates the best number of clusters. The model selection process compares the silhouette scores and Davies-Bouldin indices of both models and chooses the model with a higher silhouette score and lower Davies-Bouldin index, indicating a better clustering performance. The selected clustering model, whether K-means or HDBSCAN, will be utilized for subsequent analysis and interpretation of the clustered data.

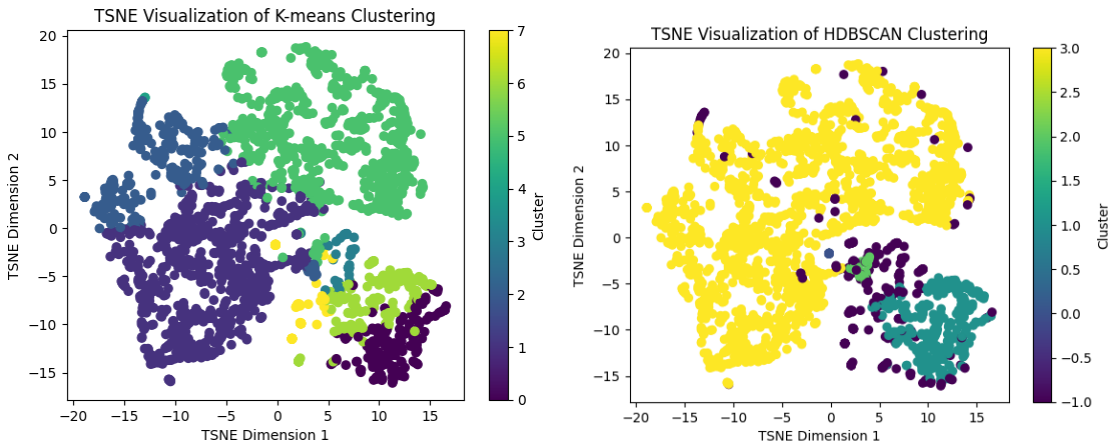


Figure 2: Comparisson of K-means and HDBSCAN using TSNE visualization for the same room.

For Figure 2 the following model comparison took place:
Silhouette Score (KMeans): 0.318
Silhouette Score (HDBSCAN): 0.332
Davies-Bouldin Index (KMeans): 1.187
Davies-Bouldin Index (HDBSCAN): 1.760
HDBSCAN was chosen as the clustering model.

A t-SNE(t-distributed Stochastic Neighbor Embedding)[9] visualization technique is used to visualize the high-dimensional sensor data (36 dimensions in this case) in a two-dimensional space. As for alternative techniques to t-SNE, two notable methods are Uniform Manifold Approximation and Projection (UMAP) [10] and Shapley Additive Explanations (SHAP)[11]. These alternative methods offer different approaches and may be preferable depending on the specific characteristics of the dataset and analysis goals.

4.4 Cluster Centroids and Feature Importance

To start understanding the clusters and their differences, it is vital to first have a look at the cluster centroids. Centroids represent the average location of data points within a cluster, so in other words, they represent the cluster center. Each of these centroids gives different sensor readings depending on its cluster. For example, a cluster that represents an unused room will show low values for each of the sensor readings.

Also, the feature importance will highlight the most influential features in cluster formation. The feature importance is calculated using the absolute difference across clusters. This process involves comparing the mean values of each feature within clusters and computing the absolute difference between these values. Features with larger absolute differences across clusters are considered more important for distinguishing between different cluster groups. The results obtained in this experiment showed a very similar pattern for all rooms. The most important feature was light, followed by CO2 and sound readings. We can interpret these results as logical as most of the changes in a room are due to the change in light conditions. On the other hand, for two of the rooms marked as labs, the order of the features was inverted. This will be further analyzed in the next section.

5 Data Clustering and Anomaly Detection

In this section, the clustered data will be visualized and analyzed to reveal underlying patterns and structures within the dataset. This will help with well-informed decision-making and anomaly detection techniques.

The process of identifying outliers involves measuring their distances from the assigned cluster centers. Outliers that surpass a predetermined threshold are classified as anomalous. This threshold must be adjusted manually before the execution of the program. The threshold value is adjusted based on specific data characteristics and the desired number of anomalies, but this project aimed to identify approximately 100 to 200 anomalies for each room out of the 2516 data points in total.

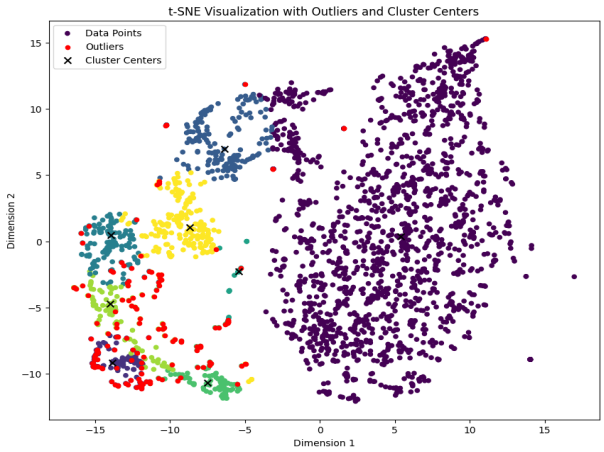


Figure 3: Result of combining t-SNE visualization with anomaly detection for a one-person office.

Figure 3 shows the general pattern for most of the rooms. First, there is a cluster with most of the data points and very few outliers. This cluster always represents when the room is unused, so most of its points occur during the nighttime. Subsequently, we can find different clusters each representing different room conditions. Anomalies within these clusters are due to room unusual activity, offering insights into varying environmental and occupancy states. A study on these anomalies and what they could indicate will be done further in this section

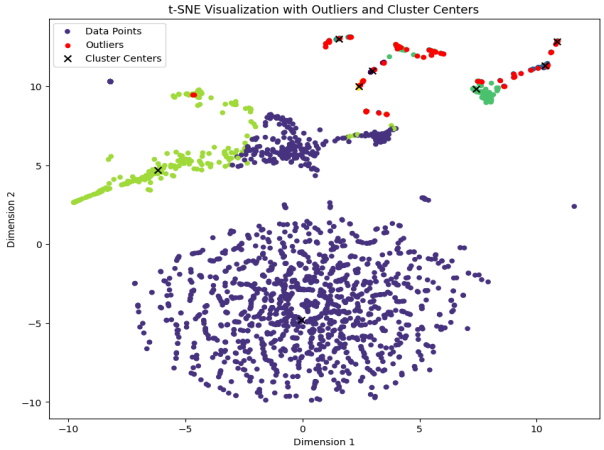


Figure 4: Result of combining t-SNE visualization with anomaly detection for a small meeting room.

While Figure 4 and Figure 3 share some similarities, such as the appearance of a very dense cluster, they also differ in a few ways that can be explained by the type of room. The primary distinction is the relative lack of density and dominance of the secondary clusters. This is most likely the result of fewer clusters or room usage overall, which is caused by a decreased use of the space.

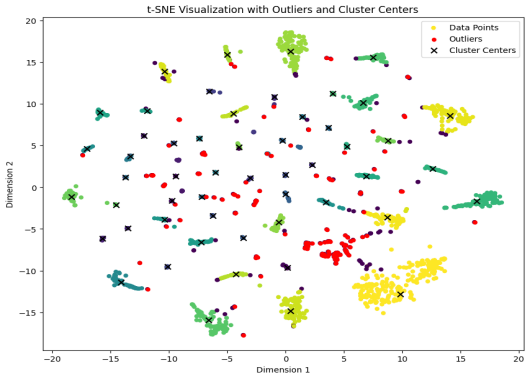


Figure 5: Result of combining t-SNE visualization with anomaly detection for a laboratory.

Figure 5 is very different from Figures 3 and 4, its pattern can be related back to the type of room (lab in this case). This room type presents a very high number of clusters suggesting very different types of room usage. As seen in section 4.4, labs have CO2 as their most important feature which can be interpreted as the room being used by different numbers of people, generating an abnormal amount of clusters.

After identifying anomalies, the next step involves analyzing sensor trends surrounding these anomalous events. This is done by setting a time window centered around each anomaly timestamp, allowing the examination of sensor readings before and after the detected anomaly. By visualizing sensor data in proximity to anomalies we can analyze what is happening during this time window, giving a further indication of why they were categorized as anomalous.

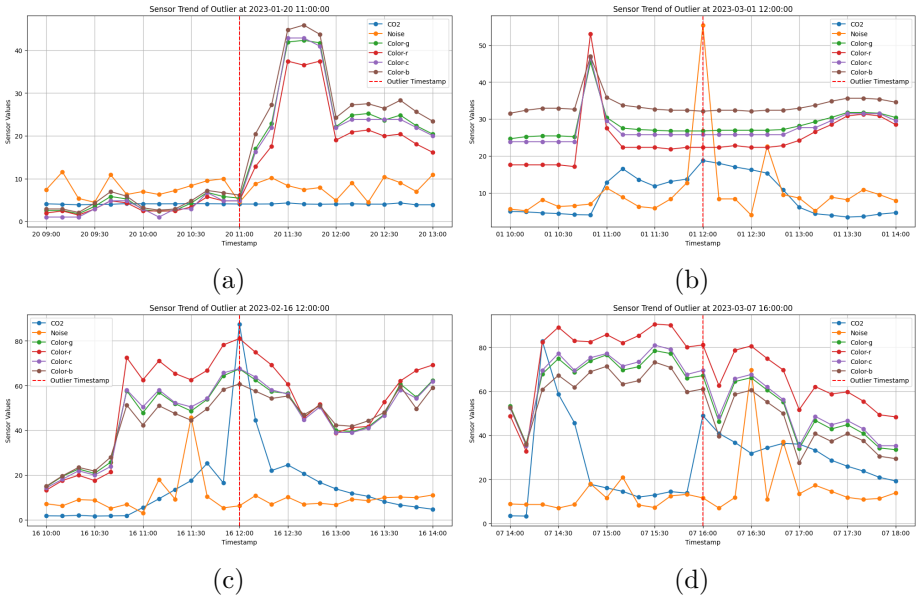


Figure 6: Different detected sensor trends surrounding anomalous events for a one-person office

Figure 6 gives a first insight into common anomalies and their trends. It is important to note that the outlier timestamp takes place within an hour. So for example, for plot (a), the anomalous event happens between 11:00 to 12:00 .The first plot illustrates a sudden increase in light intensity captured by the light sensors. The spike in light intensity may indicate an abrupt change in lighting conditions within the room. Possible scenarios could include the turning on of lights or the opening of curtains. An opposite scenario of light decreasing drastically is also a common anomaly seen in these plots. In plot (b) a sharp peak in sound intensity is observed, suggesting a significant noise event within the room. This could correspond to activities such as conversations, movement of furniture, or even equipment malfunction. Plot (c) depicts a CO2 peak highlighting a sudden rise in carbon dioxide levels within the room. This is the most alogical type of anomaly as CO2 levels tend to grow gradually. Possible explanations could be poor ventilation and inadequate air circulation, having an abnormal number of occupants in this room, or a sensor malfunction. Plot (d) showcases sensor readings indicating the active use of the room. While room usage is an expected behavior, it's worth noting that maybe for this plot, values of CO2 and sound exceeded normal usage boundaries. However, it's important to recognize that room usage itself should not be considered anomalous, as it represents one of the normal patterns of activity within the building.

To delve deeper into the context surrounding these events, it is also important to look into the cluster center time windows. This will help gain further insights into the context surrounding normal room usage and appreciate differences with anomalous plots.

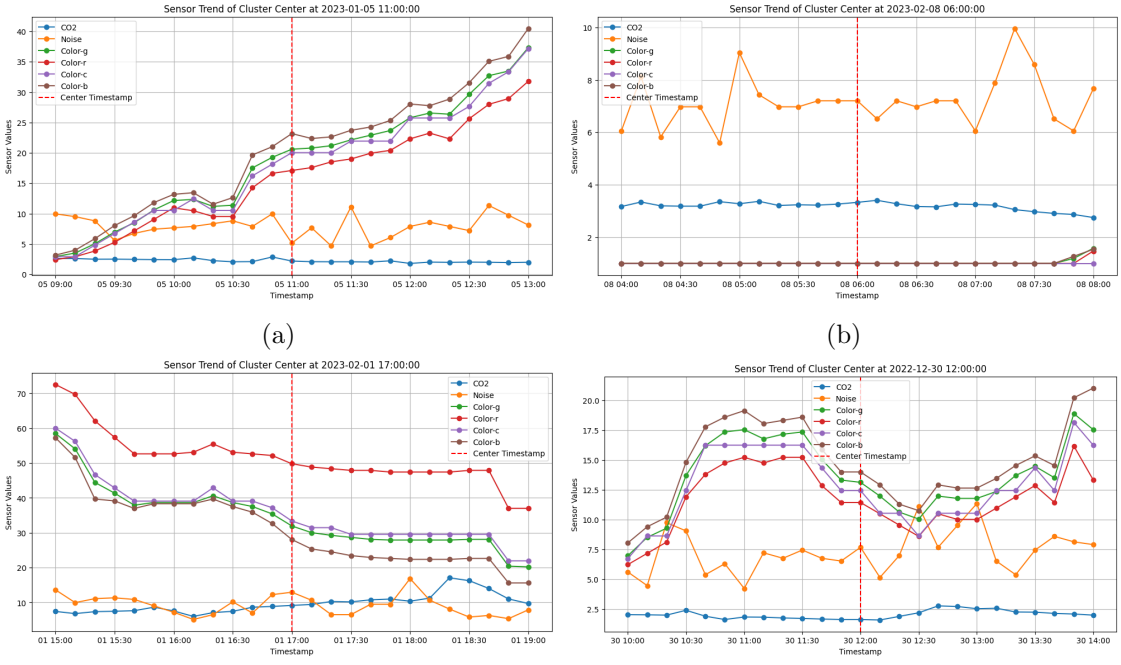


Figure 7: Sensor trends surrounding cluster centers for the same room that was used in Figure 6

Number of data points sharing cluster with the plots from Figure 7:
Cluster 1: 199 data points
Cluster 2: 1651 data points
Cluster 3: 122 data points
Cluster 4: 187 data points

Figure 7 shows the most common patterns captured by the sensors for this room. Plot (a) represents a progressive increase in light sensor readings, coinciding with morning hours (11:00 in this case). This pattern is likely attributed to the natural sunrise phenomenon, indicating an expected occurrence within the building environment. Plot (b) represents the most common state of the room as seen when printing the number of data points within the clusters. Therefore, it can be linked to the common pattern observed during nighttime hours and periods of room inactivity. Unexpectedly, the plot shows high levels of sound, but interestingly, this pattern is observed across multiple rooms, suggesting a systematic occurrence. The cause for this phenomenon is uncertain but a possibility could be attributed to the operation of machinery or electrical appliances within the building. Plot (c) could perfectly indicate the room not being used during daytime, with constant light values. Lastly, plot (d) could be indicative of active room usage, characterized by fluctuations across multiple parameters such as light, sound, and potentially CO2 levels.

Figures 6 and 7 show expected differences between cluster centers and anomalies. By comparing these plots, anomalies in the dataset can be more easily identified and interpreted by identifying anomalous behaviors or events within the larger context of regular sensor data patterns.

6 Feature Engineering and Supervised Learning

6.1 Feature Engineering

To use supervised learning algorithms for time series classification, it is necessary to include characteristics of the rooms since these features will be needed as training data. For each room type, different common anomalous patterns can be found. Each room type must be carefully studied by looking at different room clusters to extract their most characteristic features. To later study the effectivity in detecting anomalies of the newly added features it is important to add thresholds. Thresholds are applied to distinguish significant deviations from normal sensor behavior. By setting thresholds, anomalies exceeding predefined limits can be flagged.

- Mean Change Rate of Light and Sound Sensors: The most common pattern found in open spaces and offices is due to sudden spikes in light intensity or sharp increases in sound levels. The mean change rate is obtained by comparing the values for light and sound from the previous row in the dataset.
- CO2 Levels (Open Spaces): A large number of anomalies for the Open-space room type are because of high levels of CO2. This new feature is obtained by marking as true, values that exceed a manually set threshold.

- Mean Sensor Values (Laboratories): Most anomalies in the lab room type are due to a conglomeration of people, so all values seem to change at the same time. Consequently, comparing all sensor values from the previous row provides an optimal approach for capturing these irregularities.
- Nighttime Periods: As seen previously, nighttime periods across all room types consistently exhibit prominent clusters with the highest density of values. The threshold is set by filtering timestamps lower than 7:00.

The incorporation of room-specific features facilitated the identification of anomalies exceeding predefined thresholds, enabling more accurate anomaly detection across various room environments. Here are some results:

Number of anomalies for office 008: 183
Number of anomalies found after adding new features to the dataset:
Anomalies: 96
Non-Anomalies: 187
Correctly Classified Anomalies by Feature:
Light change mean: 88
Sound change mean: 8

Number of anomalies for open-space 002: 192
Number of anomalies found after adding new features to the dataset:
Anomalies: 102
Non-Anomalies: 265
Correctly Classified Anomalies by Feature:
Sound change mean: 13
Light change mean: 68
High CO2: 21

Number of anomalies for lab 071: 171
Number of anomalies found after adding new features to the dataset:
Anomalies: 92
Non-Anomalies: 23
Correctly Classified Anomalies by Feature:
Mean across columns: 92

Overall, the results demonstrate that the incorporation of room-specific features has enhanced the accuracy of anomaly detection, particularly by adding the light change mean feature, being responsible for most correctly detected anomalies.

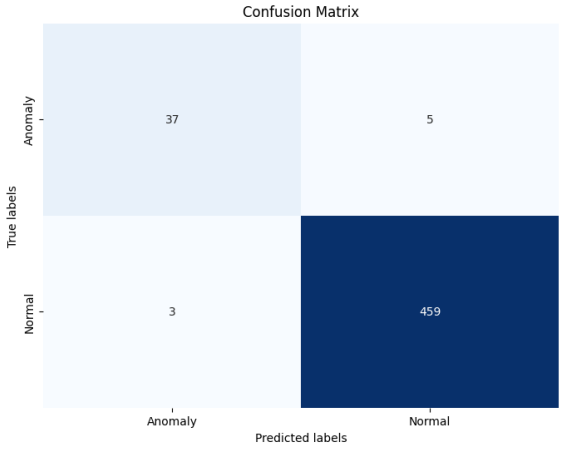
6.2 Supervised learning

Supervised learning is used to build predictive models based on labeled data, where the algorithm learns to map input features to corresponding target labels. In this project's context, the aim is to train a model capable of classifying anomalies in sensor data using the features extracted in the previous section. The first step involves splitting the dataset into training and testing sets. For this project, the size of the test set was set to 20%. The objective is to train a classifier on the training data and evaluate its performance on the unseen test data. Subsequently, the chosen model to train the data is the random forest classification algorithm[12]. Random forest is a powerful machine-learning algorithm for classification problems. The idea behind this model is to use many decision trees to find the best-performing combination.

Table 1: Classification Report for office 008

	Precision	Recall	F1-Score	Support
Anomaly	0.92	0.86	0.89	42
Normal	0.99	0.99	0.99	462

Accuracy for Table 1: 0.982



The percentage of correctly classified instances among all the instances in the test set is indicated by the accuracy score. The assessment of true positives, true negatives, false positives, and false negatives is made possible by the confusion matrix, which offers a tabular summary of predictions versus actual class labels. The classification report also includes metrics for each class, including precision, recall, and F1-score, which provide information about how well the model performs in various classes. Overall, as seen in the results, the model demonstrates strong performance in accurately distinguishing between normal instances and anomalies in the sensor data.

7 Conclusion

This project aimed to identify anomalies in multivariate real-time series data gathered from sensors positioned in different rooms of the NU building of Vrije Universiteit. With the use of data visualization tools and unsupervised machine learning techniques like K-means clustering and HDBSCAN, the project aims to not only identify anomalies but also categorize them based on room-specific patterns.

The used model for anomaly detection for each room is determined by comparing clustering models using assessment metrics such as the Davies-Bouldin index and silhouette score. Furthermore, the project employs t-SNE visualization to visualize high-dimensional sensor data in a two-dimensional space. An analysis of cluster centroids and feature importance reveals insights into room-specific patterns and important factors influencing cluster formation. Anomalies were identified not only through individual sensor readings but also by considering contextual information surrounding the anomalies, such as time window analysis and comparison with cluster center patterns. Lastly, the project delved into supervised learning techniques, particularly random forest classification, to build predictive models for anomaly detection. To ensure the high performance and accuracy of the model, new features were added after carefully studying the conditions of each type of room. The results showed great effectiveness in accurately identifying anomalies from normal instances for the sensor data.

Expected results included logical patterns in sensor data associated with the various room types. For instance, anomalies in offices often correlated with sudden changes in light intensity or sound levels, while anomalies in open spaces were often associated with elevated CO2 levels. Furthermore, feature engineering efforts significantly improved the accuracy of anomaly detection ensuring a correct previous study of the different room clusters and patterns. However, unexpected findings emerged during the analysis of sensor data patterns and anomalies. For instance, anomalies detected in various rooms exhibited distinct patterns from what could be expected as normal room usage. In addition, nighttime readings for most of the rooms showed higher levels of CO2 and sound than expected leading to uncertainty for most of the readings. On the other hand, very high accuracy and general performance of supervised modeling were expected because of the significant number of features and low number of anomalies for the test set, due to a limited amount of data.

In conclusion, the project successfully demonstrated the effectiveness of unsupervised machine learning algorithms in anomaly detection within IoT sensor networks. The project also unveiled unexpected insights, showing the importance of thorough analysis and feature engineering in anomaly detection tasks. Finally, these findings can help with decision-making processes and facilitate proactive maintenance and management of building environments.

8 Future Work

- Automated Room Type Detection: A possible direction for future research involves creating a system that can automatically identify the kind of room by analyzing the attributes of every cluster. Through the analysis of the cluster centers and the corresponding sensor readings, it might be possible to identify unique patterns linked to various types of rooms. For example, clusters representing occupied rooms may exhibit similar sensor trends across various rooms, allowing for the automated identification of room usage. Putting such a system to work would allow for customized anomaly detection tactics for particular room kinds

- Real-Time Anomaly Detection: Many real-life business anomalies require immediate action. Implementing a system that continuously monitors sensor data streams and identifies anomalies as they occur would facilitate timely intervention and proactive maintenance efforts. These types of projects already exist and are in continuous development. For example, the paper by Siya Chen, G. Jin, and Xinyu Ma[13] proposes a spacecraft on-orbit real-time anomaly detection method based on CF-LSTM.

- Alternative Unsupervised Anomaly Detection Techniques: Future research should consider investigating alternative unsupervised anomaly detection methods other than clustering algorithms. Techniques such as autoencoders or isolation forests offer alternative approaches without the need for labeled training data. Examining how well these methods work with the project's dataset could reveal important information when compared to the current results.

- Building Management Systems: Further usage of this project by building management systems[14] based on the results could improve overall operational efficiency. Such systems could automatically adjust heating, ventilation, lighting levels and other environmental parameters to maintain optimal conditions.

9 References

[1] Belay, M.A.; Blakseth, S.S.; Rasheed, A.; Salvo Rossi, P. Unsupervised Anomaly Detection for IoT-Based Multivariate Time Series: Existing Solutions, Performance Analysis and Future Directions. Sensors 2023, 23, 2844. https://doi.org/10.3390/s23052844

[2] Julien Audibert. Unsupervised anomaly detection in time-series. Neural and Evolutionary Computing [cs.NE] Sorbonne Université, 2021. English. https://theses.hal.science/tel-03681871

[3] Figueiredo, Ilan, Guarieiro, Lilian, Sperandio Nascimento, Erick Giovani. (2020). Multivariate Real Time Series Data Using Six Unsupervised Machine Learning Algorithms. 10.5772/intechopen.94944.

[4] Chenli Wang, Jun Jiang, Thomas Roth, Cuong Nguyen, Yuhong Liu, Hohyun Lee, Integrated sensor data processing for occupancy detection in residential buildings, 2021, https://doi.org/10.1016/j.enbuild.2021.110810.

[5] Mikita Volakh, Real-time Monitoring and Predictive Modeling for Space Occupancy, 2023, https://cs.vu.nl/ versto/IoT-lab.html

[6] Abiodun M. Ikotun, Absalom E. Ezugwu, Laith Abualigah, Belal Abuhaija, Jia Heming, K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data, 2023, https://doi.org/10.1016/j.ins.2022.11.139.

[7] Stewart, G.; Al-Khassaweneh, M. An Implementation of the HDBSCAN* Clustering Algorithm. Appl. Sci. 2022, 12, 2405. https://doi.org/10.3390/app12052405.

[8] Yudhistira Arie Wijaya, Dedy Achmad Kurniady, Eddy Setyanto,Wahdan Sanur Tarihoran,Dadan Rusmana, Robbi Rahim. Davies Bouldin Index Algorithm for Optimizing Clustering Case Studies Mapping School Facilities. 2021. https://doi.org/10.18421/TEM103-13.

[9] Luuk Derksen, Using T-SNE in Python to Visualize High-Dimensional Data Sets, 2022, https://builtin.com/data-science/tsne-python

[10] McInnes, L., Healy, J., Melville, J. (2018). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. https://doi.org/10.48550/arXiv.1802.03426

[11]Lundberg, S., Lee, S. (2017). A Unified Approach to Interpreting Model Predictions. https://doi.org/10.48550/arXiv.1705.07874

[12] Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[13] Siya Chen, G. Jin, Xinyu Ma, Detection and analysis of real-time anomalies in large-scale complex system, Volume 184, 2021, https://doi.org/10.1016/j.measurement.2021.109929.

[14] D. Minoli, K. Sohraby and B. Occhiogrosso, "IoT Considerations, Requirements, and Architectures for Smart Buildings—Energy Optimization and Next-Generation Building Management Systems,", Feb. 2017, doi: 10.1109/JIOT.2017.2647881.

10 Appendix

The project is available in this GitHub repository.
The dependencies for the correct functioning of the project can be found in this link