**Abstract**

Our research team is dedicated to solving two practical business issues: the development of an alarm classifier and the establishment of an alarm number prediction mechanism. Team member Rahul is responsible for implementing the alarm classifier based on TFIDF and K-means, identifying alarm rules through unsupervised learning methods, reducing the number of alarm events, and providing viable options for collaboration with subject matter experts and product managers. Meanwhile, Yiwei is responsible for predicting the number of alarms in a certain period in the U.S. region by establishing a time series model, which aids in the prediction and management of future alarm numbers, and optimization of system behavior. Furthermore, the prediction model helps reveal trends and seasonal patterns in alarm numbers, which is critical in addressing potential issues. In summary, our team's work will assist in better understanding, predicting, and managing the number of alarms within our systems.

**Introduction**

For our team, our goal was to address two practical business problems: the development of an alarm classifier and the implementation of an alarm count prediction mechanism.

**1. Alarm classifier**

Regarding the development of the alarm classifier, Rahul took the lead in this project. He was responsible for establishing the text preprocessing functions, utilizing the TF-IDF text feature extractor, using the elbow method to determine the optimal number of clusters, applying the K-means algorithm for clustering the alarm messages, and visualizing the clusters using PCA. Additionally, Yiwei also contributed to the project by defining the category names and transforming technical and model content into actionable recommendations for clients.

Benefits of Alarm Classifier:

* By identifying potential rules and patterns, we can group alarms and significantly reduce the number of alarm events.
* Utilizing unsupervised ML methods, we divide the alarms into several groups, enabling collaboration with subject matter experts and product managers to develop alarm management workflows and intelligent digital alerts.
* This NLP based alarms classifier could be used over all message.

**2. Alarm count prediction mechanism**

Regarding the alarm count prediction mechanism, Yiwei took the lead in this project. He focused on forecasting the number of alarms for the first half of July in the US region using time series modeling. Throughout the process, he was responsible for creating the time series alarm dashboard, constructing the exponential smoothing model, performing hyperparameter tuning, and making the final predictions.

How it benefits the organization:

* The time series prediction model is highly valuable for forecasting and managing future alarm counts. By predicting future alarm numbers, we can better understand and anticipate system behavior, allowing us to be prepared in advance, such as scheduling maintenance personnel or procuring necessary equipment.
* Furthermore, the model helps us identify trends and seasonal patterns in alarm counts, providing valuable insights into system behavior and performance. For example, if we observe a significant increase in alarm counts during a specific season or time period, we can further investigate and address any underlying issues. Overall, the time series prediction model is a powerful tool that enhances our understanding, prediction, and management of alarm counts in our system.

**EDA section (M1 – M3)**

For each of the monthly:

A screenshot of a computer

Description automatically generated

**A screen shot of a computer code

Description automatically generated with low confidenceA screenshot of a computer code

Description automatically generated with low confidence**

For 1st, 2nd& 3rd datasets we have null values and we cleared them off because they are not having any greater impact on this larger dataset.

**A screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generated with medium confidence**

**A screenshot of a computer

Description automatically generated with medium confidenceA screenshot of a computer

Description automatically generated with medium confidence**

**A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidence**

We plotted the bar graph which shows the severity of the alarm label for three months. The count value denotes the frequency of occurance.

**A picture containing text, screenshot, diagram, display

Description automatically generatedA picture containing text, screenshot, diagram, plot

Description automatically generated**

**A picture containing text, screenshot, diagram, plot

Description automatically generated**

We then started to evaluate the time stamps and has concluded as follows:

We have created a new column called the event duration which is basically the difference between the activated and cleared time stamps. Later on we converted the date and time format to the seconds format which is useful for us to evaluate.

Here goes with the description of the event duration column(only 1st month data) which is difference between the activated and cleared time stamps.

A screenshot of a computer program

Description automatically generated with low confidence

Now going with the description of the difference between the two consecutive activated timestamps:

A screenshot of a computer program

Description automatically generated with low confidence

Intrepeting:

* Count indicates the number of events in our dataset
* Mean represents the average duration of the events
* Standard deviation is variability in the event duration.
* Minimum represents the min duration of the event duration
* 25%,50%,75% represents the 1st, 2nd, 3rd quartile of the event duration.
* Max represent the max duration of the event duration.

Visualising the event duration distribution:

A picture containing text, screenshot, display, number

Description automatically generated

Exploring the time stamps(event dur) by event count by hour for the day:

A picture containing text, screenshot, diagram, line

Description automatically generated

Exploring the timestamp (event duration) by event count by day for the week:

**A picture containing text, screenshot, font, number

Description automatically generated**

And on further analysis, as the data is very huge, I am splitting the data in to 100 parts and then applying the nlp algorithms to clean the text .

**EDA section (M4 – M6)**

**1.Data cleaning**

**A screenshot of a computer

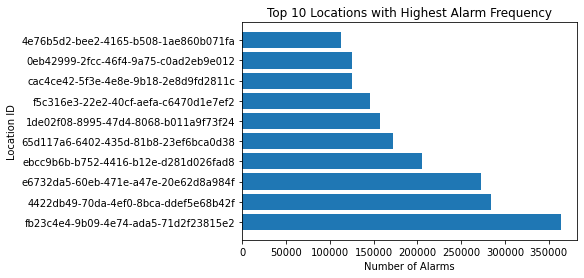
Description automatically generatedA screenshot of a computer

Description automatically generated**

* Checking the data structure - Converting timestamps to datetime format: Convert the ActivatedTimestamp and ClearedTimestamp columns to datetime format so they can be used for time series analysis.
* Missing value handling - Over 63% of the locationid columns are missing, and several options for handling these missing values may need to be considered.

**2.DataAnalysis**

**Figure 1: Top 10 locations with the highest number of alarms**



The graph shows the top ten locations with the highest number of alarms. analyzing the frequency of alarms can reveal which locations have frequent failures or problems with their devices. This can help us determine which locations may need more attention and maintenance to reduce the risk of failure and improve equipment reliability.

**Figure 2: Top 10 Asset Types by Alarm Count**

**A picture containing screenshot, colorfulness, rectangle, square

Description automatically generated**

As you can see from the chart, the Top 10 Asset Types by Alarm Count, the device type with the highest alarm frequency is UPS, with a count of over 6 million. The second one is RPDU, with more than 4.5 million alarm frequencies. The third to fifth are POD, CAMERA, and EMS, with alarm frequency of over 2.5 million, 1.8 million, and 0.7 million, respectively. By comparing the reliability of devices, the difference in alarm frequency of different device types can reveal the reliability difference between device types. A device type with a high alarm frequency may mean that the type is more prone to failures or problems, while a device type with a low alarm frequency may indicate that the type is relatively reliable. Such comparisons can help us evaluate the performance and reliability of different equipment types to guide purchasing and equipment selection decisions.

**Figure 3: The distribution of alarm severities**

A picture containing text, circle, screenshot, colorfulness

Description automatically generated

By analyzing the percentage of alarm severities, the most frequent alarm category is CRITICAL, accounting for 43.1%, followed by WARNING, accounting for 32.4%, followed by INFO, ERROR, and FAIL, accounting for 10.4%, 9.0%, and 5.2%, respectively. According to the sponsor's introduction, we understand that the alarms are listed in descending order of urgency as FAIL, CRITICAL, WARNING, ERROR, and INFORMATION. A high number of alarms with high severity may indicate a greater risk and potential for serious failures or security problems. Conversely, a low number of high-severity alarms may indicate a stable system or device performance and low risk.

**Figure 4: Top 5 Alarm Labels by Count**

A picture containing screenshot, rectangle, colorfulness, square

Description automatically generated

The top 5 alarm labels by count are Netbotz Appliance Alarm, Battery disconnected, communication status threshold, phase near overload, device The analysis will help us to better target these types of alarms and improve the efficiency of alarm classification by targeting these high frequency alarms.

**Question 5: Top 5 Organization Country Codes by Count**

A picture containing screenshot, rectangle, colorfulness, square

Description automatically generated

The chart shows that the top 5 countries with the highest alarm frequency are US, IN, IT, GB, FR. This will help us understand the distribution of Schneider Electric in the global market, which countries have a high alarm frequency, and develop specific strategies to improve the efficiency of alarm classification and reduce the number of alarms.

**Figure 6: Average Alarm Duration for Top 10 AlarmLabel**

A picture containing colorfulness, screenshot, rectangle, lilac

Description automatically generated

By analyzing the average alarm duration for the top 10 alarm labels, we can obtain the following valuable information:

1. Identify the severity of the problem: The average alarm duration can be used as an indicator to assess the severity of the problem. If the average duration of an alarm tag is longer, it may indicate that the problem takes longer to resolve or requires more resources. This can help us determine which problems are particularly complex or time-consuming so that appropriate measures can be taken in resolving them.

2. Fault Repair Optimization: By analyzing the average alarm duration, we can identify the types of alarms that take longer to repair on average. This helps to optimize the fault repair process and improve the efficiency of fault resolution. For alarm types with longer average duration, we can consider improving the troubleshooting process, enhancing training, providing better tools and resources, etc. to reduce the troubleshooting time.

As you can see from the graph, the average alarm duration for the top 10 alarm tags is more than 150,000 minutes, which is a very long-lasting alarm duration, so we should do some in-depth investigation and update for this point.

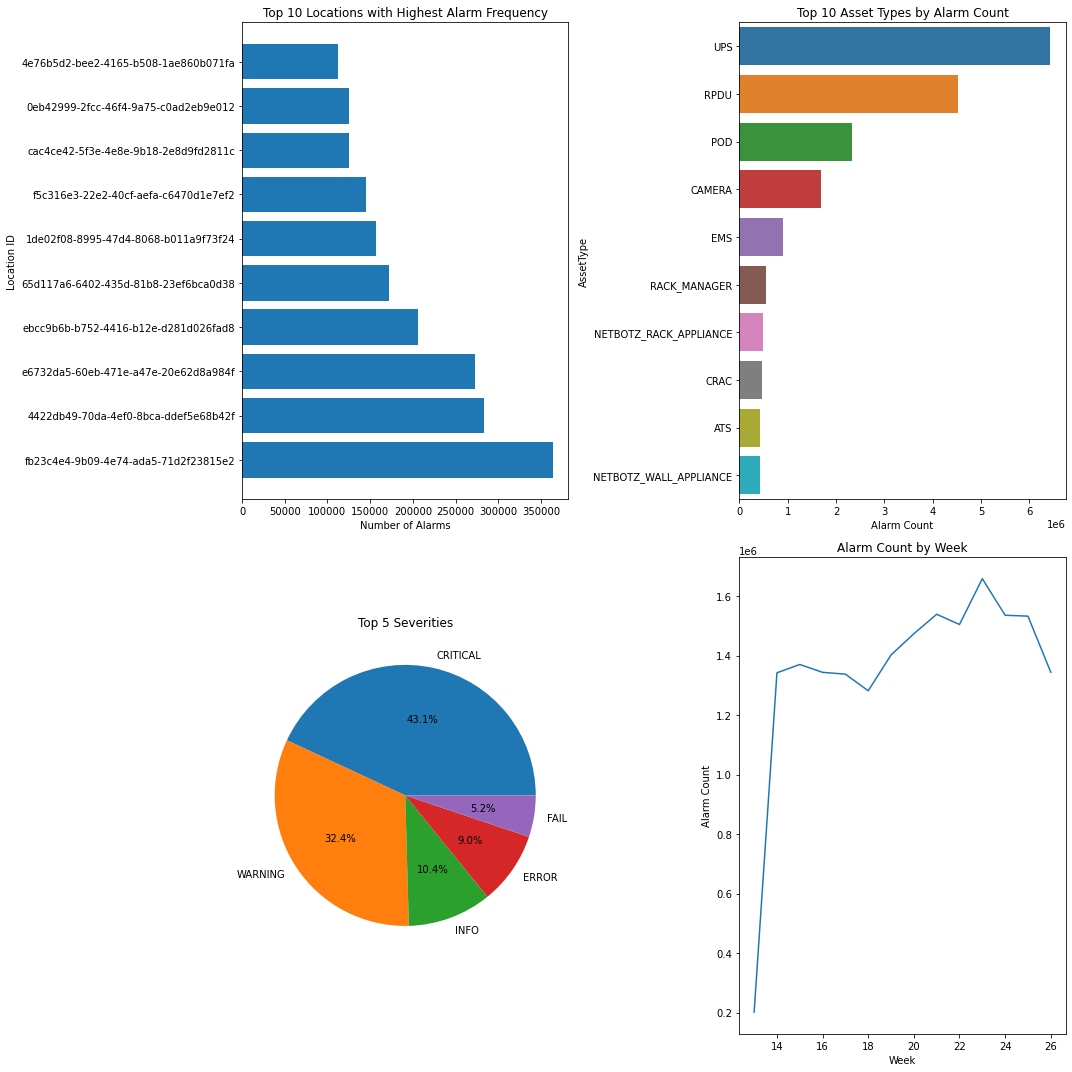
**Figure 7: The distribution of the frequency of alarms per week**

**A line graph on a white background

Description automatically generated with low confidence**

By understanding the distribution of the frequency of alarms per week, we can better know which week has the highest and lowest frequency of alarms, and explore the specific causes. As you can see from the graph, there is a sharp increase in the number of alarms from week 13 to week 14, from 200,000 to about 1.3 million. After 18 weeks from week 14, the number of alarms remained almost unchanged, with a little downward trend, but between weeks 18 and 23, the number of alarms increased further, peaking at 1.7 million in week 23. After that, there was a downward trend until week 26, and the frequency of alarms dropped to 1.4 million in week 26.

**Dashboard analysis**



**Interpretation**

The dashboard we made is about answering four questions about the top 10 locations with the highest number of alarms, the severity of the top 5 alarms, the type of the top 10 alarms, and the distribution of the frequency of alarms per week . The tool used is Python, and the two libraries called are matplotlib and seaborn to achieve a graphical presentation of all our questions in an easy-to-understand way for the audience to understand the information I want to convey. If you have ideas for enhancing our dashboard, we're always open to them, thanks.

**Machine Learning Section**

**Part 1 – Clustering model base on NLP techniques**

Here is the cleaned text and which is free of double spaces, special characters, and fullstops. And, we are further working on the merging of the datasets, clustering analysis and then finding the patterns.

A screenshot of a computer

Description automatically generated with low confidenceA white background with black text

Description automatically generated with low confidence

Next, we merged our **6 months data** and considered **only the US state** location and extracted the unique alarm message data from each month and finally merged all the unique data.



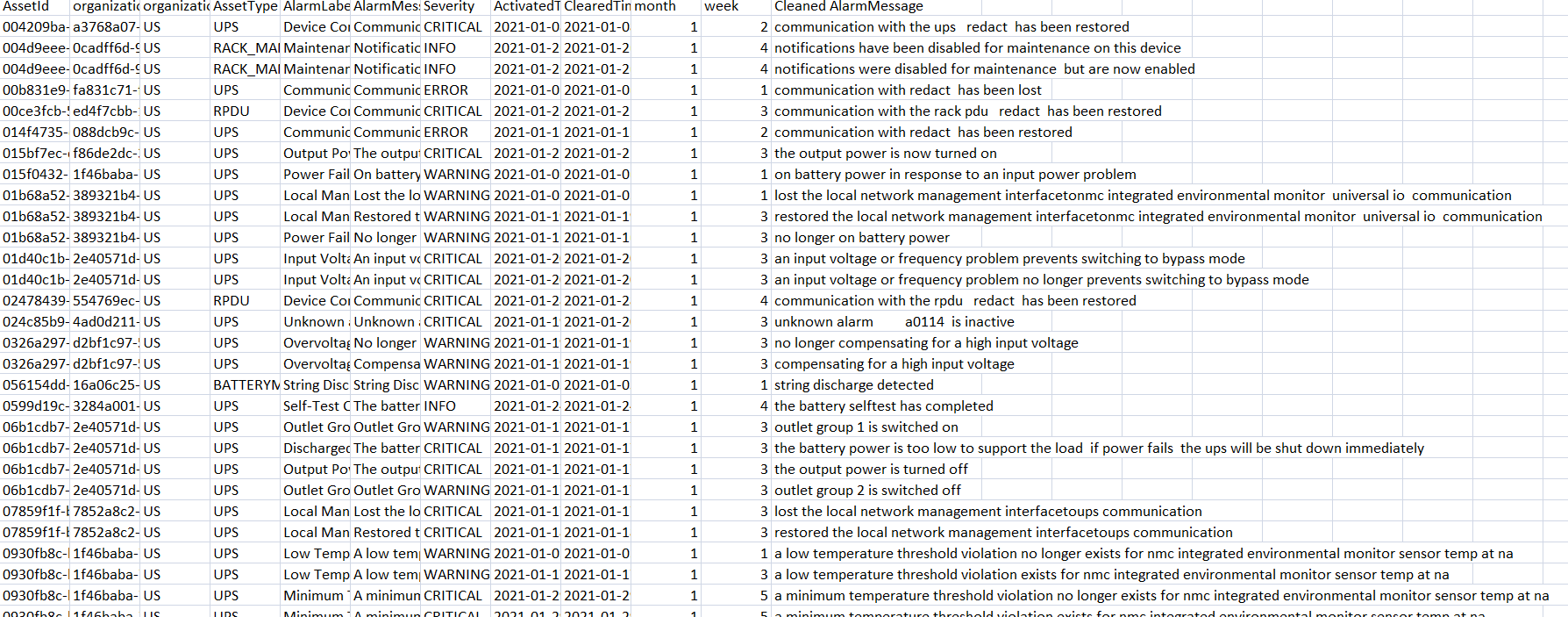
**Text Preprocessing**



**Interpretation**

This provided code contains two text preprocessing functions: clean\_sentence and clean\_column. these functions are used to clean and process the text data. the clean\_sentence function removes unwanted characters by regular expressions, removes extra spaces, and converts the text to lowercase. the clean\_column function Iterates over the specified columns in the given DataFrame, divides each text and applies the clean\_sentence function to clean it, and finally adds the cleaned text as a new column. In addition, the code downloads the required NLTK resources to support the word splitting function. These functions can help extract and process the text data to provide cleaner and more standardized text data for subsequent analysis and modeling.

**Figure: The snippet of the output of the cleaned alarm message using nlp cleaning/text normalization.**



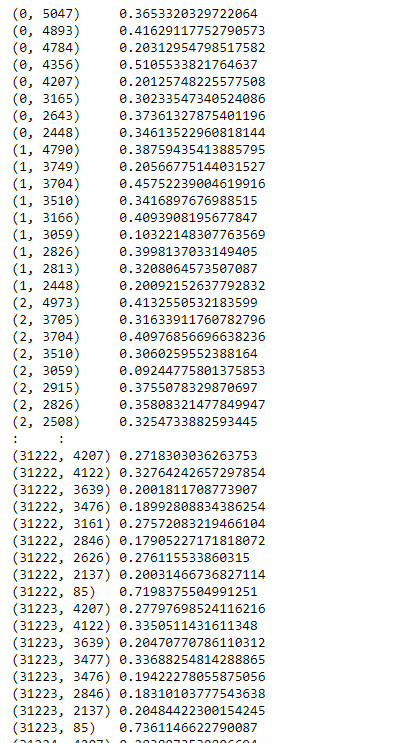
**TF-IDF text vectorization processing**



**Interpretation**

After cleaning the sentences using the NLP text normalization, we had performed the word vectorisation using the Term Frequency-Inverse Document Frequency(TF-IDF) method and the Bag of words Method(BOW).

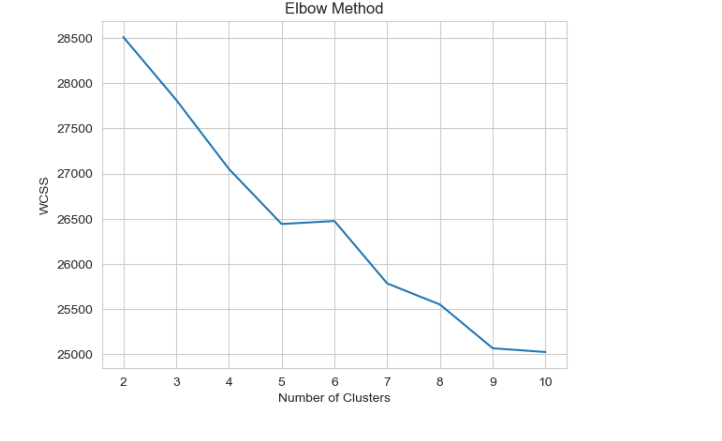
The text is processed using the TF-IDF vectorization method in the scikit-learn library. First, the initialization of TF-IDF vectorization is performed by importing the TfidfVectorizer class. Then, the vectorizer is adapted to the text columns in order to build vocabularies and calculate document frequencies. Finally, the text columns are transformed into TF-IDF matrices by calling the transform method, where each text is represented as its lexical item weight in the vocabulary. This processing method converts the text data into a numerical representation for subsequent machine learning model training and analysis.



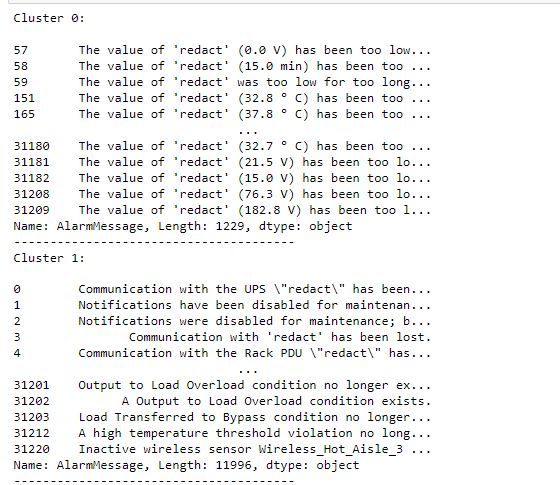
**Find the best cluster numbers**

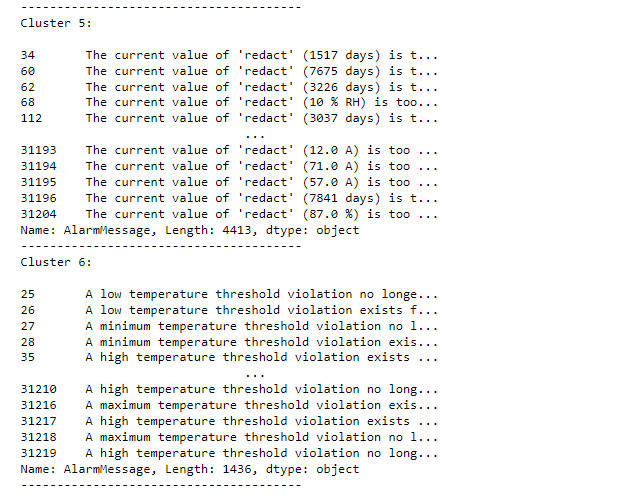
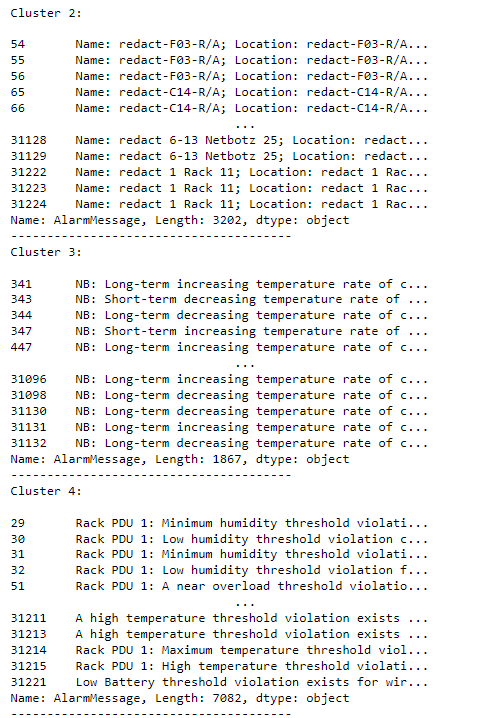
WCSS (Within-Cluster Sum of Squares) is one of the commonly used metrics in clustering algorithms. It measures the sum of squares of the distances between each data point in the clustering result and the center of the cluster to which it belongs. Specifically, for each data point, WCSS calculates the square of the Euclidean distance between it and the center of the cluster to which it belongs, and the sum of the squared distances of all data points is summed up to obtain the WCSS value of the entire clustering result.

When analyzing the results, we usually observe the curve of WCSS versus the number of clusters and look for the inflection point. The WCSS decreases faster before the inflection point and slows down after the inflection point. We will choose the number of clusters corresponding to the inflection point as the best number of clusters. **So we choose 7 as our number of clusters.**



As per the above optimized number of clusters, we used the k means clustering unsupervised algorithms for the clustering of the dataset.Below are the seven number of clusters and their patterns.





**Interpretation**

Based on the form and content of each category, I have defined a name for each category and provided the corresponding explanation:

* **Cluster 0: Distribution Module Breaker Open**

The alarm messages in this category relate to the distribution module's circuit breaker opening. It is possible that a specific circuit breaker for a distribution module was opened due to a fault or other reason, causing a power interruption or other problem.

* **Cluster 1: Communication and System Status**

Alarms in this category are related to communication and system status. Alarms may be due to an interruption or loss of communication with the equipment or system, or alarms about the status and functionality of the equipment.

* **Cluster 2: Redact Name and Location**

Alarm messages in this category contain redact (redact) information for the name and location. The specific name and location have been removed, possibly because of privacy or sensitive information.

* **Cluster 3: Long-term and Short-term Temperature Rate**

Alarms in this category relate to both long-term and short-term temperature rates of change. Alarms may be information about the rate of temperature change and can be used to monitor temperature trends and changes in a system or environment.

* **Cluster 4: Temperature and Humidity Threshold Violation**

Alarms in this category are related to temperature and humidity threshold violations. Alarms may be generated because the temperature or humidity has exceeded the set threshold range, indicating that there may be a problem with the server room or equipment environment.

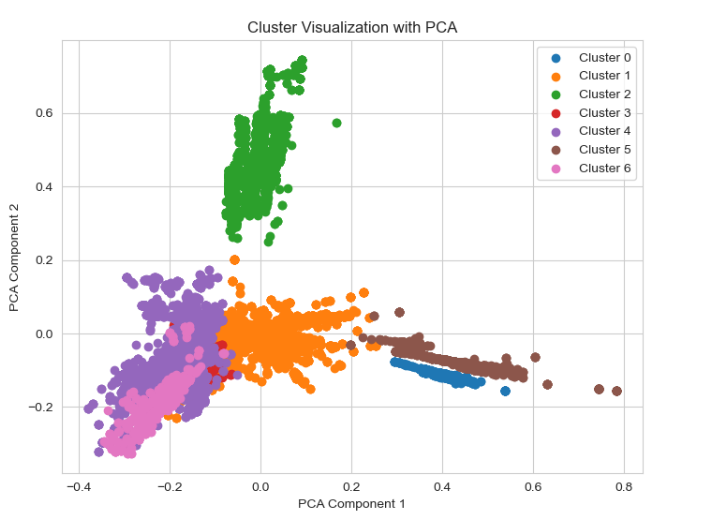
* **Cluster 5: Power and Load Conditions**

Alarms in this category are related to power and load conditions. Alarms may be information about power supply, load conditions and transitions, and can be used to monitor the operating status and performance of power systems and equipment.

* **Cluster 6: Value Threshold Violation**

Alarms in this category involve value threshold violations. The alarm may be because a specific value (e.g., temperature, time, etc.) is outside of the set threshold range, indicating a possible problem with the system or device.

**Cluster Visualization with PCA**



**Interpretation**

The above plot provides a visualization of the clusters obtained from the K-means clustering algorithm using PCA for dimensionality reduction. The scatter plot represents the data points in a two-dimensional space, where each data point is represented by its corresponding PCA components.

The plot shows the distribution of the data points based on their assigned clusters. Each cluster is assigned a unique color, and the legend provides a visual guide to identify each cluster.

The position of each data point on the scatter plot represents its projection onto the two principal components obtained from PCA. The x-axis corresponds to the first principal component (PCA Component 1), and the y-axis corresponds to the second principal component (PCA Component 2). The coordinates of each data point on the plot are determined by the values of its PCA components.

The plot helps visualize the separation or grouping of data points in the lower-dimensional space defined by the two principal components. Data points that are closer together in the plot are more similar to each other based on their PCA components, suggesting that they share common characteristics or patterns.

The visualization allows you to observe the cluster structure and identify any patterns or relationships between the data points within and between clusters. It provides an overview of how the clustering algorithm has grouped the data points based on their similarities in the original high-dimensional space, projected onto a two-dimensional space for visualization purposes.

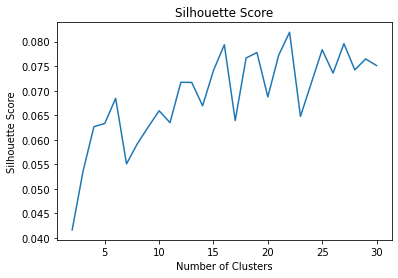
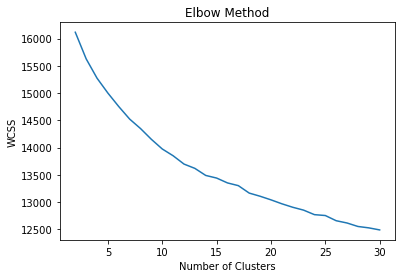
**Conclusion and recommendations**

By building alarm classifiers and clustering and classifying alarm messages, several specific benefits can be realized for the sponsoring organization as follows:

1. Fault prevention and rapid response: By identifying and grouping similar alarm events, potential failures or problems can be identified earlier and appropriate preventive and response measures can be taken to reduce the risk of failures and to respond quickly when they do occur, reducing downtime and lost production.
2. Resource optimization and efficiency improvement: Through clustering and classification, alarm data can be better understood and analyzed to identify the main causes and patterns of alarms, and from them, opportunities for optimization of resource utilization can be identified. This can help optimize maintenance plans for equipment, optimize resource allocation and scheduling, and improve the utilization and efficiency of equipment and systems.
3. Alarm management and workflow optimization: By grouping alarm events and defining the corresponding handling processes and policies, more effective alarm management and workflow can be established. This helps reduce redundant alarms, reduce the rate of false alarms, and improve the accuracy and credibility of alarms, thereby improving the quality of alarm processing and decision making.
4. Data-driven decision support: By analyzing and mining alarm data, deeper insights can be gained into the operational status, trends, and changes in equipment and systems. This provides data-driven decision support to sponsoring organizations, helping to develop more accurate maintenance plans, improve equipment performance and reliability, and enhance overall operational effectiveness.
5. Intelligent alarm management and predictive maintenance: Through in-depth analysis and modeling of alarm data, intelligent alarm management systems can be built and predictive maintenance techniques can be applied. This enables sponsoring organizations to better plan and schedule maintenance activities, predict equipment failures and problems in advance, minimize downtime and losses, and achieve sustainable operation and maintenance of equipment.

**Part 1 – Clustering model base on NLP techniquesUPDATE (JUL 3, 2023)**

**1. Find the best cluster number**



Since WCSS increases with the number of clusters and there is no inflection point, I chose to use the Silhouette score versus the number of clusters to determine the optimal number of clusters. As seen in the figure, the Silhouette score first reaches its highest at the number of clusters = 16, which is about 0.07. Therefore, to avoid the extra computational stress and overfitting caused by too many clusters, I choose 16 as our number of clusters.

**2.Case of each type of clustering**

A picture containing text, screenshot, font, document

Description automatically generatedA picture containing text, font, screenshot, white

Description automatically generatedA picture containing text, screenshot, font, document

Description automatically generatedA picture containing text, screenshot, font, document

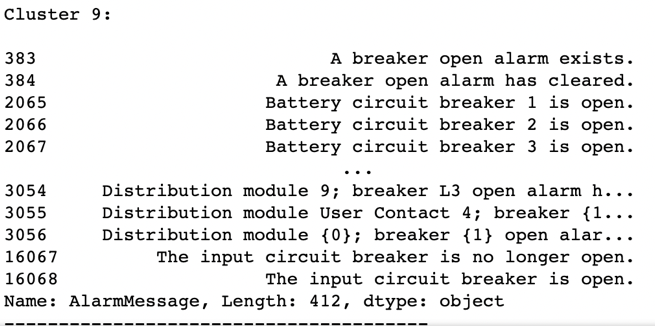
Description automatically generatedA black text on a white background

Description automatically generated with low confidenceA picture containing text, screenshot, font, document

Description automatically generatedA black text on a white background

Description automatically generated with low confidenceA picture containing text, screenshot, font, document

Description automatically generatedA black text on a white background

Description automatically generated with low confidenceA picture containing text, screenshot, font, number

Description automatically generatedA picture containing text, screenshot, font, number

Description automatically generatedA picture containing text, screenshot, font, document

Description automatically generatedA picture containing text, screenshot, font, document

Description automatically generatedA screen shot of a computer code

Description automatically generated with low confidenceA picture containing text, screenshot, font, document

Description automatically generated

**Cluster name defination**

Topic 0: "Electrical Current"

Background knowledge: alarm messages related to current.

Suggested Topic Name: Current Abnormal

Topic 1: "Temperature Threshold Violation"

Background knowledge: Alarm messages related to temperature threshold violation.

Suggested Topic Name: Temperature Violation

Topic 2: "Value Abnormality"

Background knowledge: Alarm messages related to value abnormality.

Suggested topic name: Value Abnormality

Topic 3: "Condition Change"

Background knowledge: Alarm messages related to condition change.

Suggested topic name: Condition Change

Topic 4: "Normal State"

Background knowledge: Alarm messages related to normal state.

Suggested Topic Name: Normal State

Topic 5: "Battery Voltage"

Background knowledge: Alarm messages related to battery voltage.

Suggested Topic Name: Battery Voltage Abnormal

Topic 6: "Battery Enclosure"

Background Knowledge: Alarm messages related to the battery enclosure.

Suggested Topic Name: Battery Enclosure Issues

Topic 7: "Door Access"

Background knowledge: Alarm messages related to door access.

Suggested Topic Name: Door Access Status

Topic 8: "Alarm Inactive"

Background Knowledge: Alarm messages related to alarm status being inactive.

Suggested Topic Name: Alarm Inactive

Topic 9: "Rate and Term"

Background knowledge: Alarm messages related to rate and term.

Suggested Topic Name: Rate and Term Exceptions

Topic 10: "Rack PDU Threshold Violation"

Background Knowledge: Alarm messages related to Rack Power Distribution Unit Threshold Violation.

Suggested Topic Name: Rack PDU Threshold Violation

Topic 11: "Sensor Door Condition"

Background Knowledge: Alarm messages related to sensor and door conditions.

Suggested Topic Name: Sensor Door Condition

Topic 12: "Integrated Monitor"

Background Knowledge: Alarm messages related to integrated monitoring.

Suggested Topic Name: Integrated Monitor Exceptions

Topic 13: "Power Bypass"

Background knowledge: Alarm messages related to power bypass.

Suggested Topic Name: Power Bypass Issues

Topic 14: "Threshold Cleared"

Background Knowledge: Alarm messages related to Threshold Cleared.

Suggested Topic Name: Threshold Cleared

Topic 15: "Front Sensor"

Background Knowledge: Alarm messages related to the front sensor.

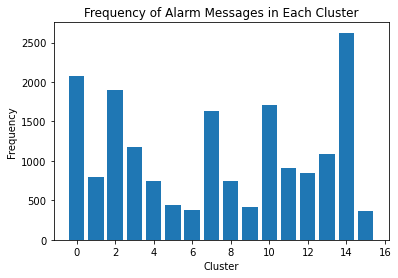
Suggested topic name: Front Sensor Exception

**3. Topic extraction**

Next, I use Latent Dirichlet Allocation (LDA) for the process of topic modeling analysis. The process includes: creating a document-word matrix, applying the LDA model, and obtaining the most likely words in each topic. With the above steps, the code can be used to perform LDA topic modeling analysis, identify the most likely topics from the alert messages, and print out the keywords for each topic.

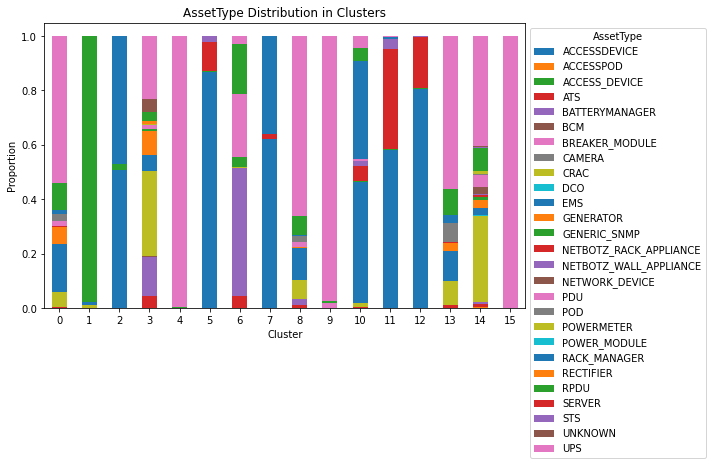
|  |  |
| --- | --- |
| **Cluster** | **Keywords** |
| Cluster 0 | of, the, redact, is, current |
| Cluster 1 | temperature, violation, threshold, for, at |
| Cluster 2 | too, redact, the, been, value |
| Cluster 3 | exists, condition, no, longer, on |
| Cluster 4 | normal, state, to, returned, has |
| Cluster 5 | the, voltage, at, is, battery |
| Cluster 6 | battery, on, the, is, enclosure |
| Cluster 7 | redact, door, location, name, action |
| Cluster 8 | is, alarm, inactive, unknown, 15 |
| Cluster 9 | at, for, of, rate, term |
| Cluster 10 | rack, violation, for, threshold, pdu |
| Cluster 11 | sensor, at, for, door, nbrk0750 |
| Cluster 12 | exists, violation, threshold, monitor, integrated |
| Cluster 13 | the, is, power, to, bypass |
| Cluster 14 | has, the, threshold, cleared, detected |
| Cluster 15 | nbrk0750, for, at, front, sensor |

**Figure: Frequency of Alarm Messages in Each Cluster**



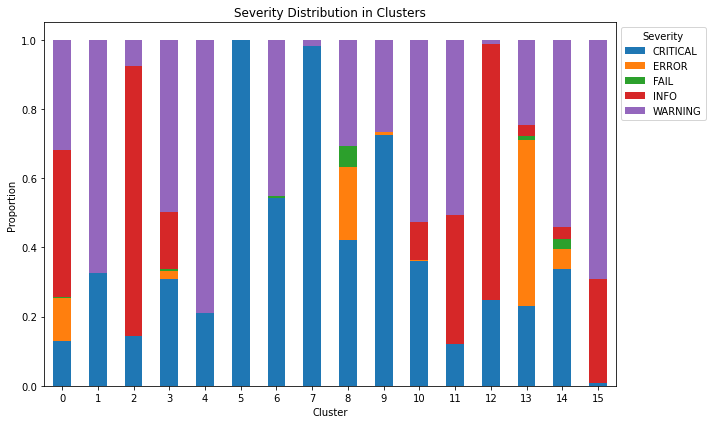
As seen in the graph, category 14 has the highest frequency of alarms, around 2500, followed by 0, 2, 10, and 7.

**Figure: AssetType Distribution in Clusters**



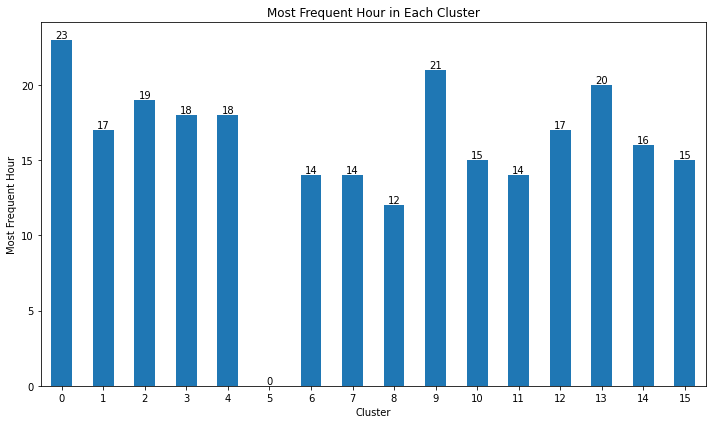
As seen in the figure, the percentage of AssetType in each clustering, for example, for Type 0 alarms, the highest percentage of UPS is about 50%, and also for Type 1 alarms, the percentage of RPDU is almost 100%. By presenting these results, we can understand more clearly the most common alarm types for each clustering, and then help us to better subsidize and aid the number of different alarm types for different clustering models.

**Figure: Severity Distribution in Clusters**



As seen in the figure, the percentage of Severity in each cluster, for example, for Class 0 alarms, INFO has the highest percentage, about 1/3, and also for Class 1 alarms, WARNING has the highest percentage, about 60%. For category 5 alarms, CRITICAL is almost 100%. By presenting these results, we can understand which clusters are the most threatening and need to be addressed with the highest importance.

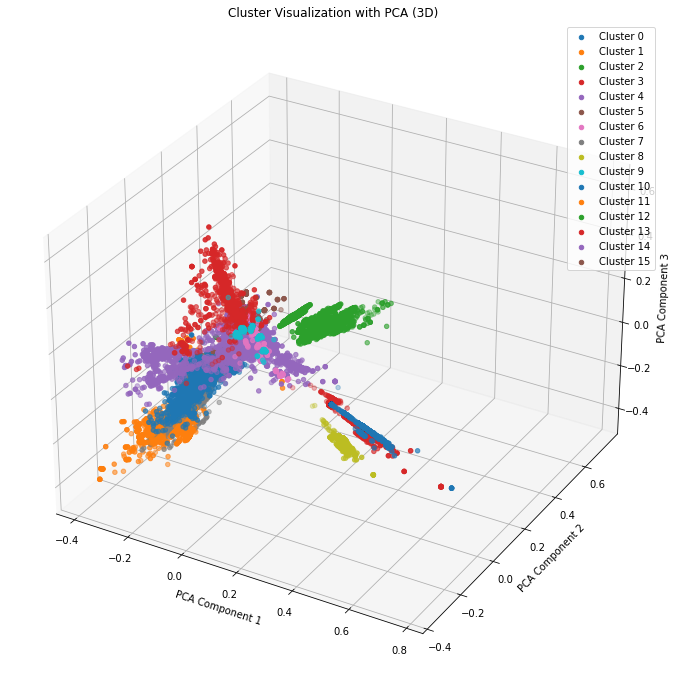
**Figure: Most Frequent Hour in Each Cluster**



The graph shows which times of the day each cluster has the highest alarm frequency in the time of day. For example, for category 0 alarms, 23:00 is the highly frequent period, and for category 1 alarms, 17:00 is the highly frequent period. Through the presentation of these results, the

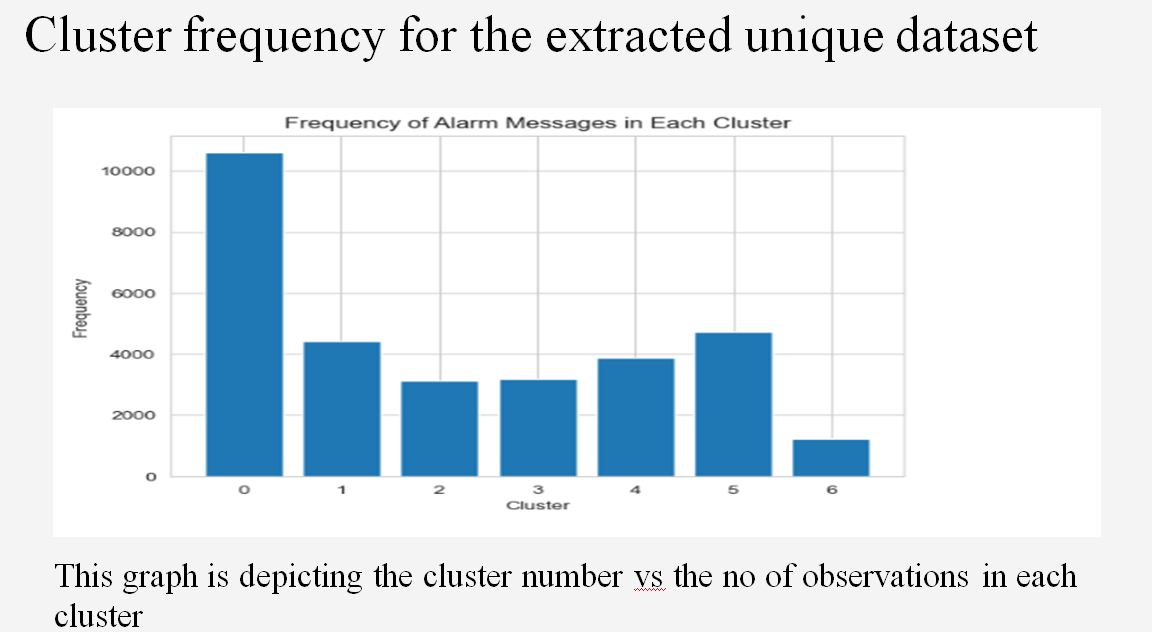
* Resource allocation and optimization: By observing the time periods with the highest alarm frequency in each cluster, it can help companies to allocate resources and optimize operations rationally. For category 0 alarms, 23:00 is a highly frequent period, which may mean that the monitoring and handling capacity of resources and personnel need to be enhanced during that time period to cope with peak hour alarms. Companies can use these insights to rationalize staff scheduling, provide faster response and resolution to ensure alarm events are handled in a timely manner.
* Early warning and prediction: By understanding the time periods with the highest frequency of alarms in each cluster, it can help companies with early warning and prediction. For category 1 alarms, 17:00 is a highly frequent period, which may mean that monitoring and maintenance of related equipment and systems need to be enhanced during that time period to reduce the occurrence of potential failures and alarm events. Companies can prevent and reduce the frequency of alarm events by increasing inspections and performing maintenance and overhauls in advance during these times.

**4. Two and three dimensional PCA clustering visualization**

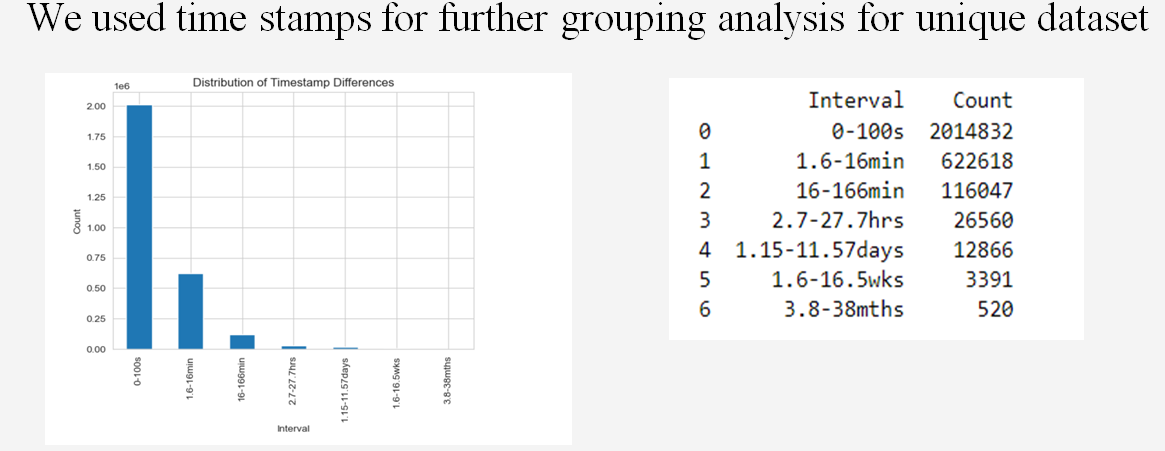


We used PCA downscaling and K-means clustering algorithms for data visualization. First, the data are projected from high-dimensional space to three-dimensional space by PCA downscaling. Then, the data are clustered using the K-means algorithm and each data point is represented as a scatter plot in the three-dimensional space. The position of each scatter point is determined by the three principal components after dimensionality reduction, and the different clusters are distinguished by different colors. In this way, we can visualize and analyze the relationship and distribution between different clusters through this three-dimensional graph.

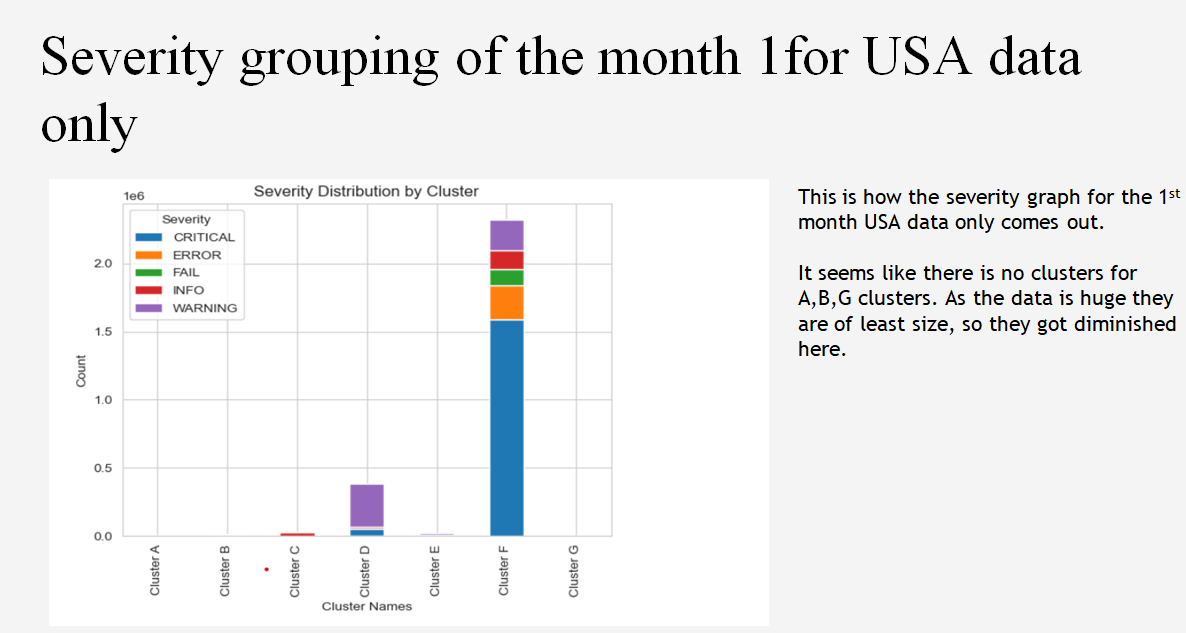
Here is the cluster frequency chart for the number of observations vs frequency for different clusters.We observe cluster 0 has the highest and the cluster 6 with the least.

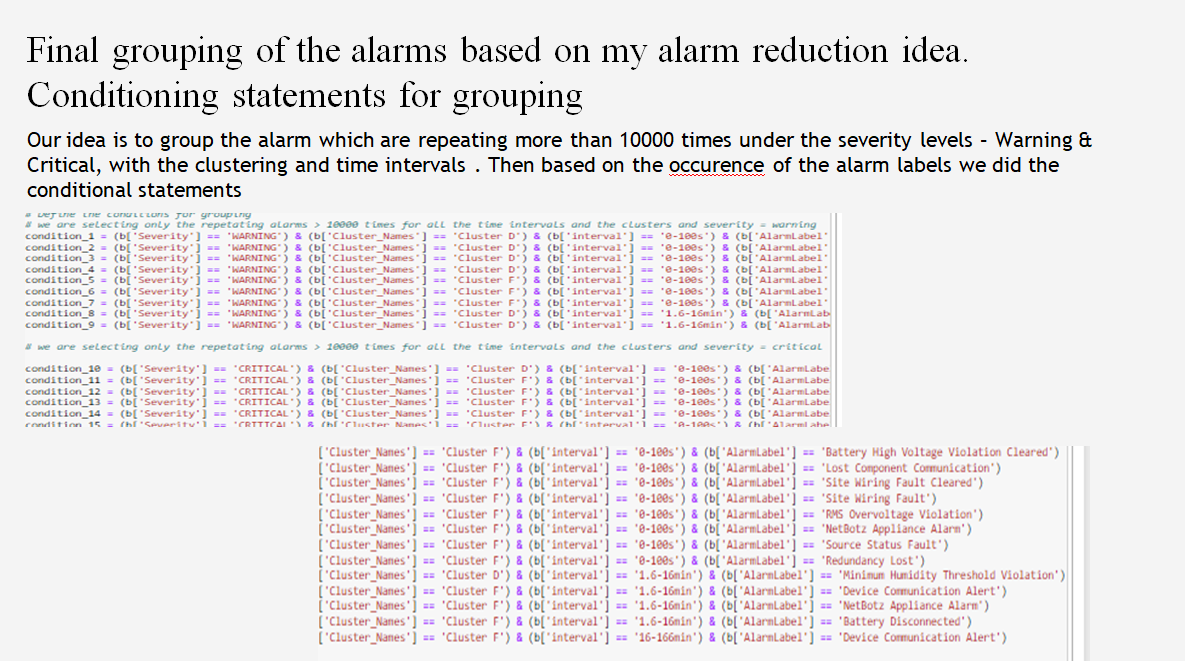


Further we used the timestamps difference to find how grouping can be done. We observe that with in 100s there are huge alarms .

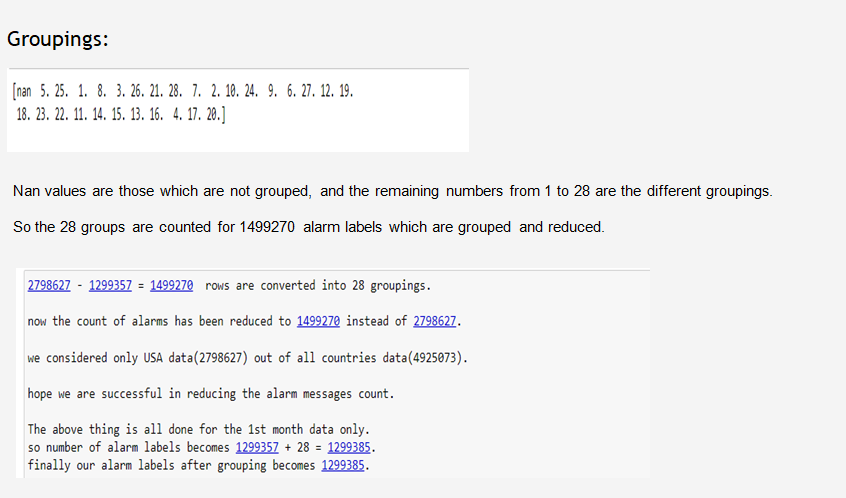


Here is the severity grouping of the 1st month for the USA data only. We observe cluster F is having the highest number of the alarm messages.





Here is how the grouping is done and alarm count is reduced.



**Part 2 – Time series prediction**

**1. Time series dashboard**

A picture containing screenshot, colorfulness, graphic design, graphics

Description automatically generated

**Interpretation**

Each of these four graphs provides insight into a different aspect of the alarm data, helping us to understand the patterns and possible causes of the alarms. The following are some of the analyses that can be derived from these graphs:

1. For each week, a stacked graph of the number of alarms in the TOP5 countries: From this graph, you can see the number of alarms per week in each country. This can help us to understand which countries have the highest number of alarms and may need further investigation into the cause. Among them, we can see that US has been maintaining the highest number of alarms as a percentage, followed by IN, IT, GB, FR. Therefore, we assume that US is the country with the most frequent alarms and needs a better alarm management system and emergency response mechanism. In addition, it can be seen that from week 13 to axis 14, the number of alarms in each country has increased significantly, and we may need to learn more about what happened in that week.

2. For each week, the stacked graph of the number of alarms for TOP5 asset type: This graph shows the number of alarms per asset type per week. It can be seen that there are two asset types with a particularly high number of alarms, namely UPS, RPDU, which may indicate that there is a specific problem of this asset type that needs attention. In addition, the number of alarms for each asset type does not change much each week, so it indicates that the number of alarms for the TOP5 asset type is relatively stable.

3. Stacked graph of the number of each severity for each week: This graph shows the number of alarms for different severity levels. You can see that the CRITICAL level has the most alarms and may need to be prioritized. In addition, from weeks 18-23, the number of alarms at the CRITICAL level stays incremental, and we may need to look at what happened that week.

4. For each week, the bar chart of the average alarm length: This chart shows the average alarm length for each week. As can be seen from the graph, there is a significant increase in the average length of alarms for weeks 17 and 22, which may indicate that some alarms were not handled in a timely manner during those two weeks.

**2. Building Time Series Models**

Objective: To construct a time series prediction model based on historical alarm data to predict the number of alarms per day for the next 15 days in US.

**I. Data Preparation**

The data source is the historical records of the alarm system, containing information such as the timestamp of each alarm, the type of alarm, the severity, and the country where it is located. We first need to clean and pre-process the data, including handling missing values, outliers, and extracting the required fields, such as alarm time and alarm type.

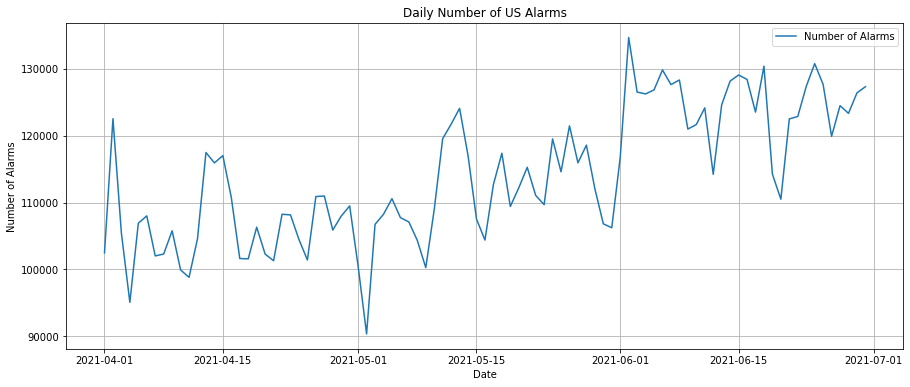
**II. Feature Engineering**

Based on the alarm time, we can generate various time series features, such as weeks, months, quarters, etc. Also, we can count the number of alarms per day as the target value for our model prediction.

**Update Last week feedbacks (Latest time-series version)**

* Based on the feedback from last time, I will make the following adjustments for time series prediction
* Go for time series forecasting of the number of alarms based on a specific country, e.g. US
* Give up using ARIMA model with poor predictive power and use exponential smoothing model instead
* Forecast the next 15 days of data

**I. Raw data time series graph of US**



From this graph, it can be identified that the number of alarms in the US region is seasonal and trending over time.

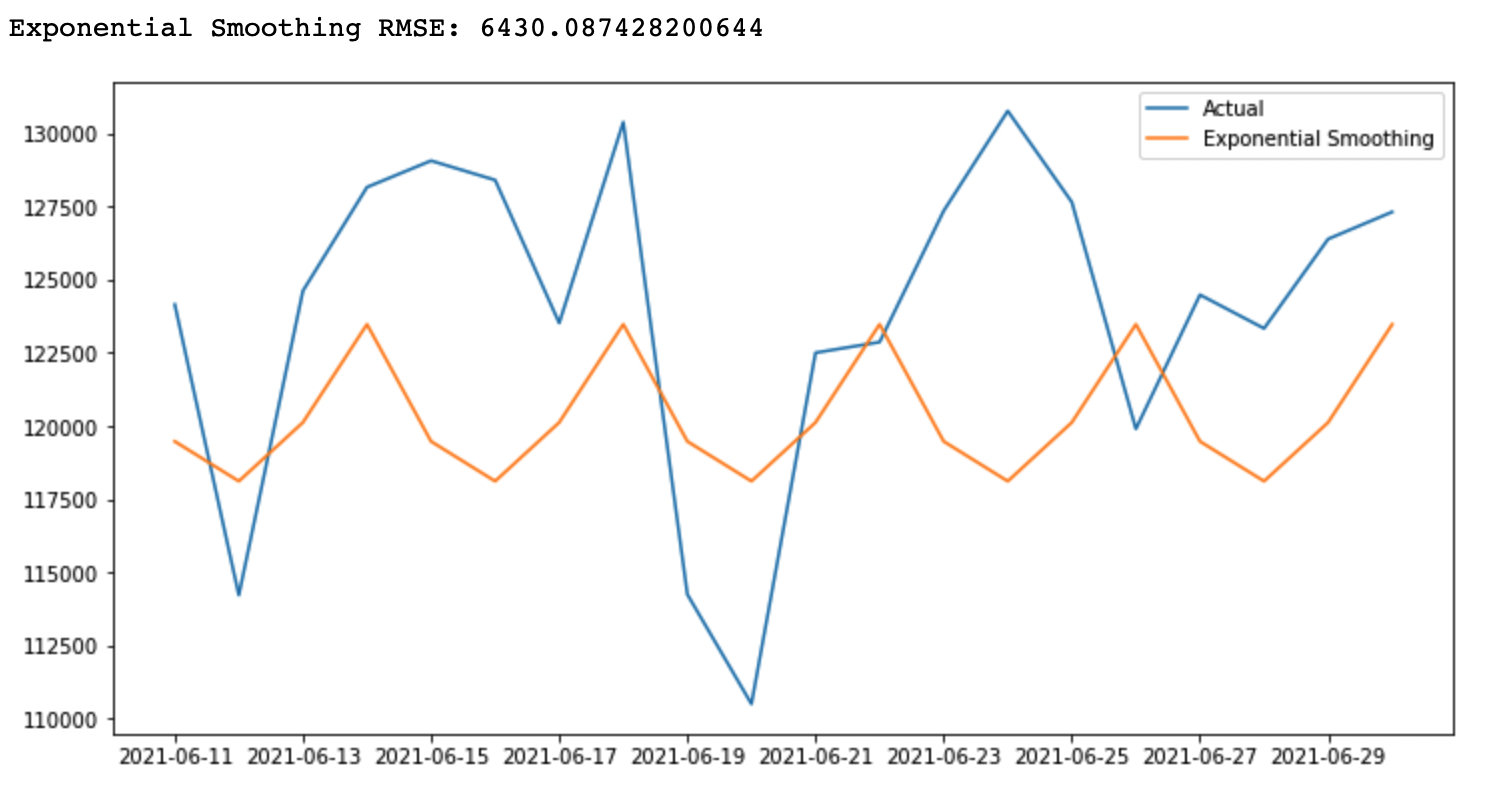
**II. Model selection**

We choose to use an exponential smoothing model, which is a common time series prediction model that captures the seasonality and trend of the data.

**III. Model training and validation**

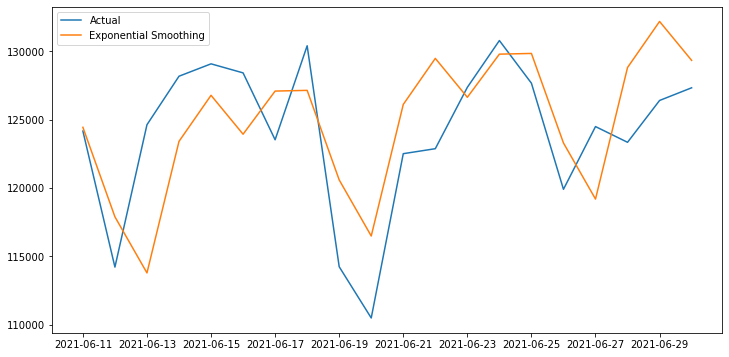
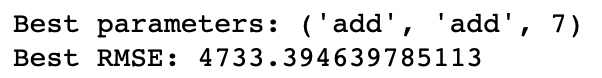
Based on the date, we divide the dataset into a training set and a test set. The test set is selected as the last 20 days of the whole time interval, and the rest is the test set. The test set is used to train the model, and the test set is used to validate the prediction of the model. When training the model, we need to adjust the parameters of the model to get the best prediction.

**IV. Model evaluation**



We use metrics such as root mean square error (RMSE) to evaluate the prediction performance of the model. From the figure, we can see that there is some difference between the trend predicted by the model and the actual trend, especially for some specific time periods, such as the period from 2021-6-23 to 6-25, the predicted trend is completely opposite to the actual trend. In addition the RMSE is as high as 6430, which means the average difference between the predicted and actual values of this model is about 6430 number of alarms, respectively. Therefore, we believe that the model is not good at prediction, so we need to further optimize the model, such as using hyperparameter adjustment.

**V. Model optimization**



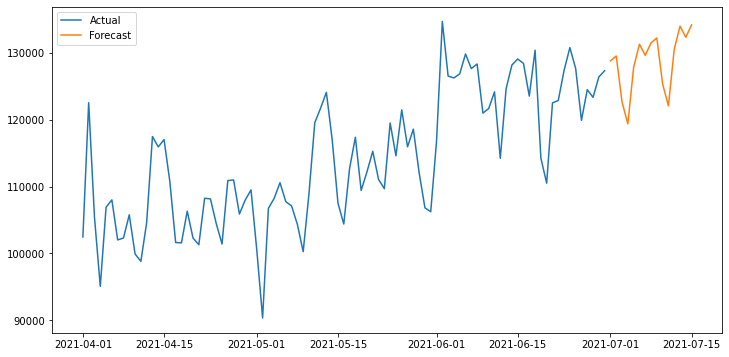
First in the tuning process, we first define the parameter options for trend and seasonality in the model, such as trend\_params and seasonal\_params. two trend models ('add' and 'mul') and two seasonal models ('add' and 'mul') are used here. Also to determine the range of seasonal periods, from 2 to 9. This means that we will try different values of seasonal periods. This allows iterating through different combinations of parameters, training and evaluating the model, and selecting the combination of parameters with the best performance.

From the results it appears that:

1. The best trend parameter (trend) is 'add', which means that a linear trend is added to the model.
2. The best seasonal parameter (seasonal) is 'add', which means that additive seasonality is added to the model.
3. The optimal number of seasonal periods (seasonal\_periods) is 7, indicating that the seasonal pattern in the model has a periodicity of 7 time steps.

Taken together, this result indicates that the time series model has relatively low error, fits the data well and makes accurate forecasts when using 'add' trend, 'add' seasonality and 7 seasonal periods. After the parameter optimization, the RMSE of the exponential smoothing model was reduced to 4733.39. This indicates that the average difference between the predicted and actual values is about 4733 number of alarms, which is lower than the model before optimization. This result indicates that the parameter optimization significantly improves the prediction performance of the model. Specifically, the prediction error of the optimized exponential smoothing model was about **26%** lower than that of the pre-optimized exponential smoothing model ((6430.09-4733.39)/6430.09). Then, we used this optimized model to predict the number of alarms in the coming month and visualized the prediction results.

**VI. Model prediction**



After determining the optimal model, we decided to predict the number of alarms in the U.S. region for the first half of July (15 days). The graph shows that not only the slight upward trend at the end of June is continued, but also there is a cyclical change, which can help us to warn in advance and develop corresponding response strategies.

**Conclusion and recommendations**

This time series prediction model is very useful for predicting and managing the number of alarms in the future. By predicting the number of future alarms, we can better understand and predict the behavior of the system, so that we can prepare in advance, such as arranging maintenance personnel in advance, purchasing necessary equipment in advance, etc.

In addition, the model can help us understand trends and seasonal patterns in the number of alarms, which is also very useful for understanding the behavior and performance of the system. For example, if we find a significant increase in the number of alarms during a particular season or time period, we can further investigate and address possible problems.

Overall, this time series prediction model is a powerful tool that can help us better understand, predict, and manage the number of alarms in our system.

**References**

[1] Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2), 1153-1176.

[2] Shi, Y., Eberlein, A., & Lu, J. (2019). A novel alarm clustering model using Affinity Propagation. *Expert Systems with Applications*, 124, 287-302.

[3] Kaur, R., Malhotra, R., & Singh, D. (2020). Machine learning based intrusion detection models: A systematic review. *Computer Science Review*, 34, 100210.

[4] Mchayaa, M., Azizi, A., & Hassouni, M. E. (2020). Machine Learning for Real-Time Intrusion Detection and Classification. *Journal of Information Security and Applications*, 50, 102425.

[5] Wang, Z., Zhang, Z., & Li, Z. (2021). Alarm correlation analysis based on LSTM and attention mechanism. *IEEE Access*, 9, 109567-109577.

[6] Ahmed, M. K., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60, 19-31.

[7] D'Amico, A., Whitley, K., Tesone, D., O'Brien, B., & Roth, E. (2005). Achieving cyber defense situational awareness: A cognitive task analysis of information assurance analysts. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 49(3), 229-233.