Project Summary And Report Writing

About Project:

Project Summary - Machine Learning for IT Service Management (ITSM) Enhancement

Project Title: DataMites™ Project Mentoring PR-0012

Client: ABC Tech

Category: ITSM - Machine Learning

Project Reference: PM-PR-0012

Project Overview:

ABC Tech, an established IT-enabled organization, receives a significant volume of IT incidents and tickets daily. These tickets are managed according to ITIL best practices, encompassing incident management, problem management, change management, and configuration management. ABC Tech's ITIL processes have reached a matured level, and an audit confirmed that further improvements may not yield significant returns.

However, ABC Tech is facing a challenge with its incident management process, which was rated poorly in a recent customer survey. To address this issue, the organization's management explored the potential of leveraging machine learning (ML) to enhance IT Service Management (ITSM) processes. Four key areas have been identified where ML can play a crucial role:

1. Predicting High-Priority Tickets: The project aims to predict high-priority (Priority 1 and 2) tickets. This prediction allows ABC Tech to take preventive measures or address problems before they escalate, resulting in improved customer satisfaction.

2. Forecasting Incident Volume: ML will be used to forecast incident volume in different fields on a quarterly and annual basis. This forecasting helps ABC Tech better prepare with the necessary resources and technology planning to handle incoming incidents efficiently.

3. Auto-Tagging Tickets: The project involves auto-tagging tickets with the correct priorities and departments, reducing the need for reassignments and related delays. This results in faster incident resolution and improved operational efficiency.

4. Predicting RFC (Request for Change) and Misconfigurations: The project will also focus on predicting RFCs and identifying potential failures or misconfigurations of ITSM assets. This proactive approach helps prevent service disruptions and inefficiencies.

Data Set:
The project will leverage a dataset containing approximately 46,000 records spanning the years 2012, 2013, and 2014. This data will be queried from a MySQL database with read-only access.

Key Data Fields:
- CI_Name
- CI_Cat
- CI_Subcat
- WBS
- Incident_ID
- Status
- Impact
- Urgency
- Priority
- Category
- KB_number
- Alert_Status
- No_of_Reassignments
- Open_Time
- Reopen_Time
- Resolved_Time
- Close_Time
- Handle_Time_hrs
- Closure_Code
- No_of_Related_Interactions

- **Related_Interaction**
- No_of_Related_Incidents
- No_of_Related_Changes
- Related_Change

Priority Matrix:
The priority matrix, based on urgency and impact, will guide the assignment of priorities to incidents.

Project Goals:
- Improve customer satisfaction by accurately predicting high-priority tickets and addressing them proactively.
- Enhance operational efficiency by forecasting incident volumes and optimizing resource allocation.
- Reduce reassignments and related delays through auto-tagging of tickets with correct priorities and departments.
- Minimize service disruptions by predicting RFCs and ITSM asset misconfigurations.

Project Deliverables:
- Machine learning models for priority prediction, incident volume forecasting, auto-tagging, and RFC prediction.
- Reports and dashboards for monitoring and analyzing ITSM processes.
- Documentation and training for ABC Tech's IT team.

This project aims to harness the power of machine learning to revolutionize ITSM at ABC Tech, ultimately resulting in improved service quality, efficiency, and customer satisfaction. It is set to make significant strides in transforming ABC Tech's IT incident management processes.

—----------

About Dataset:

1. Data Overview:
- The dataset consists of 46,606 rows and 25 columns.
- The columns include various attributes related to incidents and their details.

2. Data Columns:

- The columns in the dataset include 'CI_Name,' 'CI_Cat,' 'CI_Subcat,' 'WBS,' 'Incident_ID,' 'Status,' 'Impact,' 'Urgency,' 'Priority,' 'number_cnt,' 'Category,' 'KB_number,' 'No_of_Reassignments,' 'Open_Time,' 'Reopen_Time,' 'Resolved_Time,' 'Close_Time,' 'Handle_Time_hrs,' 'Closure_Code,' 'No_of_Related_Interactions,' 'No_of_Related_Incidents,' and 'No_of_Related_Changes.'

3. Data Types:
- Most columns are of object data type, and a few are of float64 data type.

4. Missing Values:
- Some columns contain missing values, such as 'CI_Cat,' 'CI_Subcat,' 'Priority,' 'No_of_Reassignments,' 'Reopen_Time,' 'Resolved_Time,' 'Closure_Code,' 'No_of_Related_Interactions,' 'No_of_Related_Incidents,' and 'No_of_Related_Changes.'

5. Categorical Columns:
- The categorical columns in the dataset are: 'CI_Cat,' 'CI_Subcat,' 'WBS,' 'Incident_ID,' 'Status,' 'Impact,' 'Urgency,' 'Category,' 'Open_Time,' 'Reopen_Time,' 'Resolved_Time,' 'Close_Time,' 'Handle_Time_hrs,' and 'Closure_Code.'

6. Summary of Categorical Columns:
- The 'CI_Cat' column has 12 unique categories, with 'application' being the most frequent.
- The 'CI_Subcat' column has 64 unique categories, with 'Server Based Application' being the most frequent.
- The 'WBS' column has 274 unique categories, with 'WBS000073' being the most frequent.
- The 'Incident_ID' column is unique for each row.
- The 'Status' column has two unique values, with 'Closed' being the most frequent.
- The 'Impact' and 'Urgency' columns contain numerical values with associated labels.
- The 'Category' column has four unique categories, with 'incident' being the most frequent.
- The 'Closure_Code' column has 14 unique categories, with 'Other' being the most frequent.

7. Numeric Columns:

- The numeric columns include 'Priority,' 'number_cnt,' 'No_of_Reassignments,' 'No_of_Related_Interactions,' 'No_of_Related_Incidents,' and 'No_of_Related_Changes.'

8. Summary Statistics for Numeric Columns:
- The 'Priority' column has a mean of approximately 4.22, with a range between 1 and 5.
- The 'number_cnt' column ranges from 0.000023 to 0.999997, with a mean of approximately 0.4997.
- The 'No_of_Reassignments' column ranges from 0 to 46, averaging approximately 1.13.
- The 'No_of_Related_Interactions' column ranges from 1 to 370, averaging approximately 1.15.
- The 'No_of_Related_Incidents' column ranges from 1 to 63, averaging approximately 1.67.
- The 'No_of_Related_Changes' column ranges from 1 to 9, averaging approximately 1.06.

9. Constant Feature:
- The 'Alert_Status' column is constant with all values being 'closed.' It doesn't provide any meaningful information, so it can be dropped.

10. Sample Data:
- A random sample of one row from the dataset is provided for reference.

In summary, this initial data exploration and analysis provide a basic understanding of the dataset.

—----------

Data Preprocessing and Outlier Handling Report:

In this report, we discuss the steps taken for data preprocessing and outlier handling in our dataset. The dataset contains information related to incidents, and we have focused on numerical columns to address missing values and outliers.

Handling Missing Values:

We began by examining missing values in the dataset. Here are the key steps taken to address missing data:

1. No_of_Related_Changes: We noticed that the 'No_of_Related_Changes' column had some missing values. To handle these missing values, we filled them with the value 2, which was the mode of this column.

2. **No_of_Related_Incidents**: Similarly, we found missing values in the 'No_of_Related_Incidents' column. We filled these missing values with the median value of 1.

3. **Reopen_Time**: The 'Reopen_Time' column had a significant number of missing values, approximately 95% of the data. Due to the high percentage of missing values, we decided to drop this column from the dataset.

Outlier Handling:

To address outliers in numerical columns, we focused on the following columns: 'No_of_Reassignments', 'No_of_Related_Interactions', and 'No_of_Related_Incidents'. Here's how we dealt with each of them:

No_of_Reassignments:

1. First, we created a box plot for the 'No_of_Reassignments' column, which revealed the presence of outliers.
2. We calculated the lower and upper quartiles (Q1 and Q3) and the interquartile range (IQR).
3. We defined the lower and upper limits for identifying outliers as values below Q1 - 1.5 * IQR and above Q3 + 1.5 * IQR, respectively.
4. We found that approximately 4.23% of the data had outliers in the 'No_of_Reassignments' column.
5. To handle these outliers, we replaced them with the median value.

No_of_Related_Interactions:

1. We created a box plot for the 'No_of_Related_Interactions' column, indicating the outliers' presence.
2. We calculated the lower and upper quartiles (Q1 and Q3) and the interquartile range (IQR).
3. We defined the lower and upper limits for identifying outliers as values below Q1 - 1.5 * IQR and above Q3 + 1.5 * IQR, respectively.
4. We found that approximately 7.37% of the data had outliers in the 'No_of_Related_Interactions' column.
5. To handle these outliers, we replaced them with the median value.

No_of_Related_Incidents:
- The most common number of related incidents is 1.
- The distribution of the number of related incidents is skewed to the right, indicating that there are more incidents with fewer related incidents than those with more related incidents.
- The maximum number of related incidents observed is 53.

To handle outliers:
- We calculated the first quartile (Q1) and the third quartile (Q3) and found that both were 1.0.
- The interquartile range (IQR) is 0.0.
- Since there were no outliers above or below the upper and lower limits (both equal to 1.0), we did not make any changes to this column.

No_of_Related_Changes:
- The most common number of related changes is 1.
- The distribution of the number of related changes is skewed to the right, indicating that there are more changes with fewer related changes than those with more related changes.
- The maximum number of related changes observed is 53.
- The median number of related changes is 2.
- The 75th percentile of the number of related changes is 4.
- The 95th percentile of the number of related changes is 10.

To handle outliers:
- We calculated the first quartile (Q1) and the third quartile (Q3) and found that both were 2.0.
- The interquartile range (IQR) is 0.0.
- Since there were no outliers above or below the upper and lower limits (both equal to 2.0), we did not make any changes to this column.

In summary, we have effectively handled missing values and outliers in your dataset. By filling in missing values and adjusting outliers, we have prepared the data for further analysis or modelling. These preprocessing steps ensure that the data is cleaner and more reliable for any downstream tasks.

—----------

Feature Engineering Report:

1. **Data Overview:**
   - We performed an initial examination of the dataset, displaying the first 10 rows.
   - We investigated unique values and value counts for the" No _ of _ Related Interactions " column.

2. Data Preprocessing:
   - We created a copy of the dataset, referred to as `data1`.
   - We removed certain columns from `data1`, including "WBS," "Incident_ID," "Urgency," "No_of_Reassignments," "No_of_Related_Incidents," "No_of_Related_Changes," "Open_Time," "Resolved_Time," "Close_Time," and "Handle_Time_hrs."

3. Correlation Analysis:
   - We calculated the correlation matrix for the remaining columns within `data1`.
   - A heatmap was generated to visualize the correlations, facilitating our understanding of the relationships between different features.

4. Highly Correlated Features:
   - We designed a function to identify highly correlated features, with a specified correlation threshold of 0.7.
   - We examined the dataset to detect highly correlated features; however, no such features were found based on the chosen threshold.

It's important to note that the feature engineering process can be iterative. We may explore additional feature engineering techniques, such as creating new features, managing missing data, and encoding categorical variables, to enhance the model's performance or prepare the data for machine learning tasks. Additionally, feature selection may involve experimenting with various

correlation thresholds or alternative feature selection methods to optimize the feature set.

—----------

Model Building Report

1. Predicting High Priority Tickets

Feature Selection and Data Preprocessing:
In the case of predicting high-priority tickets, we selected a subset of features that are available when a ticket arrives. These features include 'CI_Cat' (Configuration Item Category), 'CI_Subcat' (Configuration Item Subcategory), 'WBS' (Work Breakdown Structure), and 'Category.'

We encoded categorical variables using Label Encoding to convert them into a numerical format. We split the data into training and test sets, with 70% of the data used for training. Standardization was applied to the features to ensure that all the numerical variables had the same scale.

Model Selection and Training:
We trained several machine learning models, including Logistic Regression, Decision Tree Classifier, Random Forest Classifier, CatBoosting Classifier, AdaBoost Classifier, and Support Vector Classifier, to predict high-priority tickets.

Model Evaluation:
For each model, we evaluated its performance on both the training and test sets. The key evaluation metrics used include accuracy, precision, recall, and F1-score. Here are the results for the Decision Tree model:

- Training Set:
  - Accuracy: 0.8202
  - Precision: 0.8237
  - Recall: 0.8202
  - F1 Score: 0.8198

- Test Set:
  - Accuracy: 0.8194
  - Precision: 0.8223

- Recall: 0.8194
- F1 Score: 0.8183

Similar evaluations were performed for other models.

Summary:
Among the models tested, the Decision Tree Classifier and Random Forest Classifier showed the highest accuracy and F1 scores for predicting high-priority tickets. The Decision Tree model yielded an accuracy of approximately 82%, while the Random Forest model achieved similar results. Further model tuning and hyperparameter optimization may be necessary to improve the model's performance.

2. Forecasting the Incident Volume

Data Preparation:
In this task, we aimed to forecast the volume of incidents over time. The incident data was first prepared by parsing the 'Open_Time' column to ensure consistent date formats. Duplicate values were removed, and the date column was set as the index.

Exploratory Data Analysis:
We conducted exploratory data analysis (EDA) to understand the incident volume over time. The incident volume was visualized as a time series, revealing an increase in incidents after October 2013.

Time Series Forecasting:
We selected the ARIMA (AutoRegressive Integrated Moving Average) model for time series forecasting. We used the SARIMAX model from the Statsmodels library and identified the model with the lowest AIC (Akaike Information Criterion). The selected ARIMA model (1, 1, 1) was used for forecasting.

Model Evaluation:
We predicted future incident volumes for a specific period and compared the predictions with observed data. The model's performance was assessed based on the visual comparison between predicted and observed values.

Summary:

The ARIMA model (1, 1, 1) was chosen for forecasting incident volumes, and it provided predictions for the specified period. The model's performance can be further evaluated using additional metrics or fine-tuned to improve accuracy.

3. Predicting Request For Change (RFC)

Feature Selection and Data Preprocessing:
For predicting Request For Change (RFC), we selected predictors that included 'CI_Subcat,' 'WBS,' 'Priority,' 'Category,' 'No_of_Related_Interactions,' and 'No_of_Related_Incidents.' These features were subjected to label encoding and standardization.

Model Selection and Training:
We trained machine learning models, including the Decision Tree Classifier and Random Forest Classifier, for the RFC prediction task.

Model Evaluation:
The models were evaluated based on accuracy, precision, recall, and F1-score on both the training and test sets. The Decision Tree Classifier yielded an accuracy of approximately 98.92%, and the Random Forest Classifier showed similar results.

Summary:
Both the Decision Tree and Random Forest models achieved high accuracy in predicting Request For Change (RFC). The Decision Tree model provided an accuracy of nearly 98.92%.

—----------