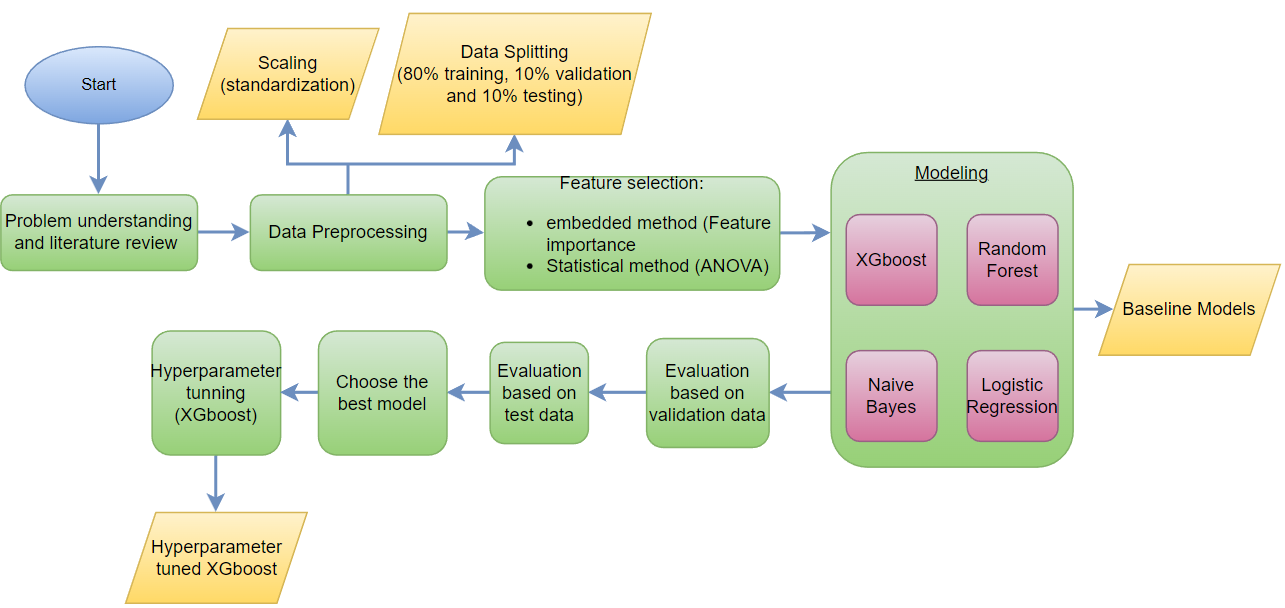
**Multiclass classification:**

**2.2 Detailed design:**

Because of the imbalanced data, the binary class classification was not enough for network failure classification, so applying multiclass classification based on failure data would help to classify failure types.

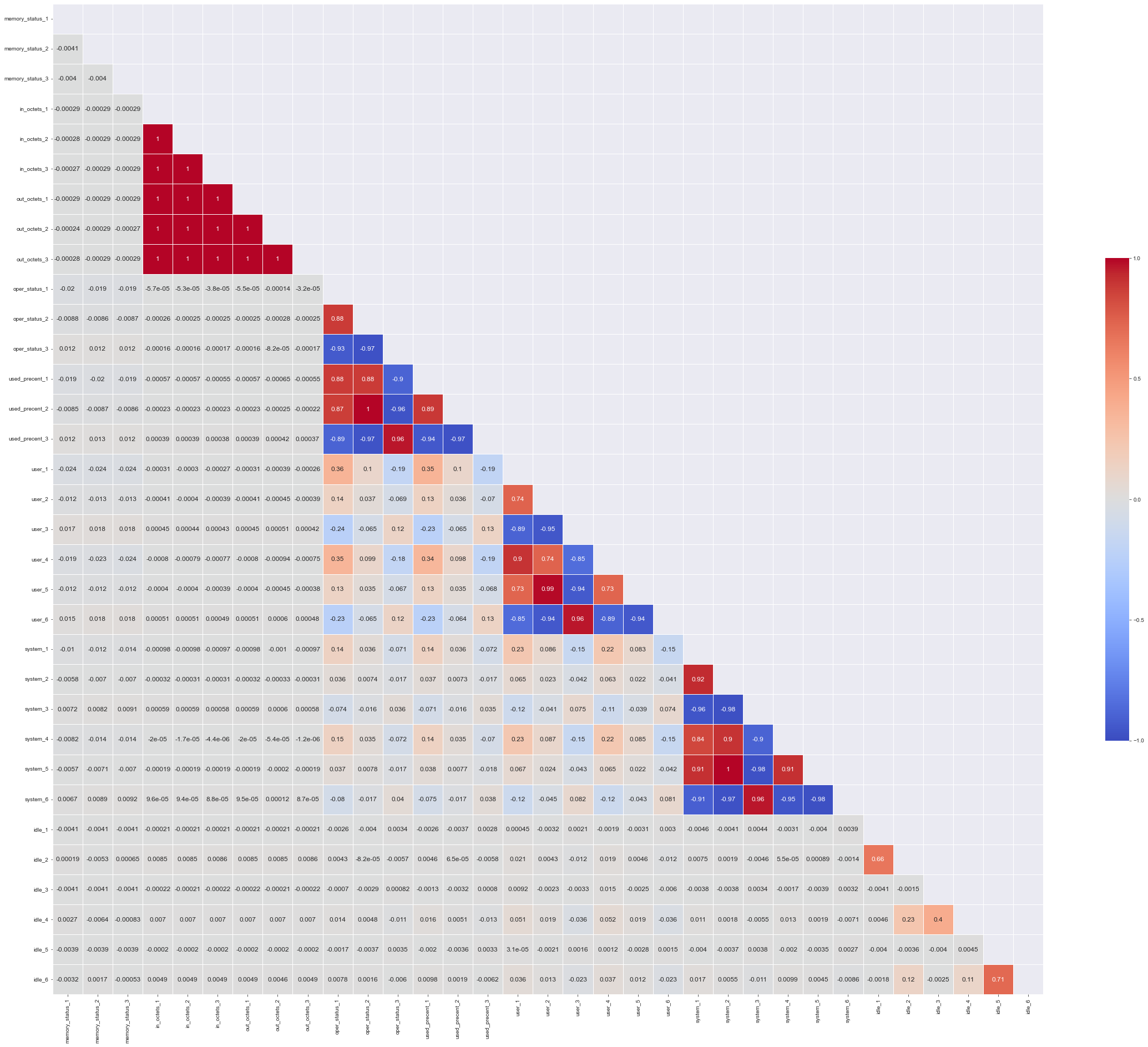
The goal of multiclass classification is to classify the network into normal and different failure types which are:

* bridge-delif
* memory-stress-start
* interface-loss-start
* interface-down
* vcpu-overload-start

By performing data preprocessing, feature selection on failure data, then apply XGBOOST, RandomForest Naïve Bayes, and logistic regression as baseline models. Finally, tuning and evaluating the results of the champion model.

**2.3 Implementation:**

**After applying binary classification, studying the reasons of network failure, and based on the domain expert the most important 33 features are shown in the following figure:**

****

Standardization also applied to center the values around the mean using standard deviation.

1. **Apply logistic regression model:**

Logistic regression model is one of supervised techniques of machine learning that provides the probability of predicting the class label.

Parameters used to train logistic regression as a baseline model:

|  |  |
| --- | --- |
| multi\_class | random\_state |
| auto | 42 |

1. **Apply naïve bayes model:**

Naïve bayes model is one of the supervised techniques in machine learning which is based on the Bayes theorem that has an effective probabilistic of class labels.

The kernel used here is GaussianNB that can predict continuous values.

1. **Apply RandomForest model:**

RandomForest model is one of the supervised learning techniques that is based on tree.

|  |  |
| --- | --- |
| Random\_state | 42 |

1. **Apply XGBOOST model:**

XGBOOST model is one of the gradient boosting techniques for supervised learning in machine learning.

|  |  |
| --- | --- |
| Random\_state | 42 |

1. **Apply hyperparameter tuning on XGBOOST model:**

As the results of XGBOOST model as a baseline, it has been chosen for tuning.

Parameters of XGBOOST model:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| eta | max\_depth | min\_child\_weight | subsample | colsample\_bytree | seed | eval\_metric |
| 0.3 | 15 | 6 | 0.9 | 0.7 | 0 | mlogloss |

**2.4 Testing:**

**2.4.2 Validation and Verification:**

Confusion matrix and F1-Score have been used to evaluate the results of multi-class classification problem.

To meet the design requirements, the model can classify the network into normal and failure types of abnormal with high scores.

1. **Overall results and analysis:**
2. **Results of logistic regression model:**

**Classification report of testing logistic regression model:**

Table

Description automatically generated

According to the F1-Score, the model can’t predict class 1, class4 and class 5 and misclassify some points in class2 and class3.

Also according to the confusion matrix, the model can’t predict class1, class4 and class5.

A screenshot of a computer

Description automatically generated with medium confidence

1. **Results of Naïve Bayes model:**

Classification report on testing Naïve Bayes model:

Table

Description automatically generated

According to the F1-Score in the classification report of Naïve Bayes model can’t predict all of the classes well and it shown in the following confusion matrix:

A screenshot of a computer

Description automatically generated with medium confidence

This results are an obvious evidence that the problem is a non linear problem, so boosting techniques is the best choice to deal with this problem. Random Forest and XGBOOST will be applied.

1. **Results of Random Forest model:**

Classification report on testing Random Forest model:

Table

Description automatically generated

According to the F1-Score in the classification report of Random Forest model, the model can predict class0, class4 and class5 well, but the model misclassified class1, class2 and class3 and the confusion matrix shows that:

A screenshot of a computer

Description automatically generated with medium confidence

1. **Results of XGBOOST model:**

Classification report on testing XGBOOST model:

Table

Description automatically generated

According to the F1-Score in the classification report of XGBOOST model, the model can predict class0, class4 and class5 well, but the model misclassified some values of class1, class2 and class3 and the confusion matrix shows that:

A screenshot of a computer

Description automatically generated with medium confidence

After comparing the results of the 4 baseline models, the champion model is XGBOOST.

TSNE plot of XGBOOST as a baseline model:

Chart

Description automatically generated

According to the results of Random Forest and XGBOOST model, they are nearby from each other, the time taken to train each model will be an important factor in comparison

|  |  |  |
| --- | --- | --- |
|  | RandomForest | XGBOOST |
| Wall time | 1min 7s | 43.2 s |

XGBOOST model take less time than Random Forest model, so it’s the champion model.

1. Results of XGBOOST model after tuning:

Classification report on testing tuned XGBOOST model:

Table

Description automatically generated

According to the F1-Score in the classification report of tuned XGBOOST model, the model can predict class0, class1, class4 and class5 well, but the model misclassified some values of class2 and class3 and the confusion matrix shows that:

Diagram

Description automatically generated

TSNE plot of tuned XGBOOST model:

Chart

Description automatically generated

According to the TSNE plot of tuned XGBOOST model, the majority class is class0 which has the red color and the minority classes are class1, class2, class3, class4, and class5. Some of the classes are overlapped, so the solution to overcome overlapping is applying Cascaded model to apply a binary class classification model then apply multiclass classification model and concatenate between the results to make the model able to detect failure types if the network status is abnormal.