

INSTITUTE FOR ADVANCED COMPUTING AND

SOFTWARE

DEVELOPMENT AKURDI, PUNE

Documentation On

**“Home Credit Default Risk Prediction”**

PG-DBDA MAR 2023

*Submitted By:*

**Group No: 24**

**Deepak Kumar**

**(230341225011)**

**Sunny Kumar (230341225046)**

**Mr. Abhijit Nagargoje Mr Rohit Puranik**

**Project guide Centre Coordinator**

**Contents**

1. Introduction ..................................................................................................................................... 1

**1.1 PROBLEM STATEMENT** ................................................................................................................................. 1

**1.2 Abstract** ..................................................................................................................................................... 1

**1.3 Product Scope**............................................................................................................................................. 1

**1.4 Aims & Objectives**....................................................................................................................................... 2

2. Overall Description......................................................................................................................... 3

**2.1 Workflow of Project:** .................................................................................................................................... 3

**2.2 Data Preprocessing and Cleaning:**........................................................................................................... 3

**2.2.1 Data Cleaning:**........................................................................................................................................ 3

**2.2.2 Label encoding:**...................................................................................................................................... 4

**2.3 Exploratory Data Analysis:**............................................................................................................................ 4

**2.4 Model Building:** .......................................................................................................................................... 10

**1.Train/Test split:**..................................................................................................................................... 10

**2. Logistic Regression:**................................................................................................................................... 11

**3. Random Forest**: ............................................................................................................................................. 13

**4. XGBOOST:** .................................................................................................................................. 15

3. Conclusion: ....................................................................................................................................... 32

4. Future Scope .................................................................................................................................... 33

5. References........................................................................................................................................ 34

**List of Figures**

Figure 1 Workflow Diagram .................................................................................................3

Figure 2 Bar plot of Distribution of target..................................5

Figure 3 Histogram showing Distribution of Clients Age ..................................6

Figure 4 Countplot showing count of gender..................................................7

Figure 5 Countplot showing count of gender after removing XNA........................................8

Figure 6 Countplot showing count of gender based on Target...........................................9

Figure 7 Countplot showing count of gender having own car...............................10

Figure 8 Countplot showing number of children. ..............................................................12

Figure 9 Countplot based on occupation type..................................................12

Figure 10 Displot based on Day of birth............................................13

Figure 11 Displot based on Day of employed. ...............................................................14

Figure 12 Countplot based on anomalies based on Target...................................................14

Figure 13 Histogram showing Distribution of Days employed...................................15

**Introduction**

An existential problem for any Loan providers today is to find out the Loan applicants who are very likely to repay the loan. This way companies can avoid losses and incur huge profits.

Home Credit offers easy, simple and fast loans for a range of Home Appliances, Mobile Phones, Laptops, Two Wheeler's , and varied personal needs. Home Credit comes up with a Kaggle challenge to find out the loan applicants who is capable of repaying a loan, given the applicant data, all credits data from Credit Bureau, previous applications data from Home Credit and some more data.

**1.1** **PROBLEM STATEMENT**

**1.2 Abstract**

Stackoverflow.com is a popular Q/A platform for programmers to solve their programming queries. Here the community can rate a question as high/low quality. This enables stackoverflow to maintain high standard of questions. But a user posting a new question has no clue whether the question will be accepted or rejected. Therefore the aim of this project is to predict the quality of the question using ML models so that the user can make a calculated guess before posting the question.

1.3 Product Scope

The main use of this classification models is to check the quality of the question posted by the user in our web interface. The user will input the question’s title and body and press the Predict Question Button after this the model will predict the quality of the

question and send that quality back to the user.



IACSD-PG-DBDA-FEB-20

Page | 1

Stackoverflow Question Quality Analysis and Prediction

1.4 Aims & Objectives

The primary goal of this project is to extract quality patterns from the stackoverflow question dataset and use the trained models to predict any question given by the user on stackoverflow. Before giving the question the training of the models will be done using a bunch of different ML models and after the training is done the ML models will be compared based on their accuracy score and f1-score and the best model will be selected which will then be used to make prediction’s for the question given by the user. The prediction which is given by our system will give the user the quite insight of

the question which he/she wants to post on stackoverflow.com.



IACSD-PG-DBDA-FEB-20

Page | 2

Stackoverflow Question Quality Analysis and Prediction

**2. Overall Description**

2.1 Workflow of Project:

The diagram below shows the workflow of this project.

*Figure 1Workflow Diagram*

**2.2 Data Preprocessing and Cleaning:**

2.2.1 Data Cleaning:

The data can have many irrelevant, missing parts, HTML tags, links. To handle this part, data cleaning is done.

*1. Remove HTML tags*

The attributes present in our dataset like title and body have HTML tags like

[p, a, strong etc.]



IACSD-PG-DBDA-FEB-20

Page | 3

Stackoverflow Question Quality Analysis and Prediction

*2. Removing StopWords*

The stop words present in our dataset don’t have any significance when we train our model. We don’t lose any information while training the data. But removing stopwords will make the size of our dataset small and this reduces training time.

*3. Lemmatizing our dataset*

The data contains many words that can be reduced down to its basic form. This step will reduce training time but the dataset will still retain its original meaning.

2.2.2 Label encoding:

To make the data understandable or in human readable form, the training data is often labeled in words. Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated.

2.3 Exploratory Data Analysis:

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.

Following are some plots we used to extract some useful information



IACSD-PG-DBDA-FEB-20

Page | 4

Stackoverflow Question Quality Analysis and Prediction

*Figure 2 Pie chart showing the distribution of the question quality’s*

Exp: Here we can see that our stackoverflow dataset is uniformly distributed across all the three types of qualities namely LQ\_EDIT, LQ\_CLOSE, HQ



IACSD-PG-DBDA-FEB-20

Page | 5

Stackoverflow Question Quality Analysis and Prediction

*Figure 3 Bar plot of the most common words in the Body attribute.*

Exp: Here we can observe all the common words in the Body of the question’s in the stackoverflow dataset. This is represented using bar plot graph. The most common word is gt and least common word is var



IACSD-PG-DBDA-FEB-20

Page | 6

Stackoverflow Question Quality Analysis and Prediction

*Figure 4 Table the most common words in Body attribute*.

Exp: Here the most common words in the question’s body for our dataset. This is represented in the

color table graph. The most common word is gt



IACSD-PG-DBDA-FEB-20

Page | 7

Stackoverflow Question Quality Analysis and Prediction

*Figure 5 Table for the most common words in the title attribute.*

Exp: Here the most common words in the question’s title for our dataset. This is represented in the

color table graph. The most common word is file.



IACSD-PG-DBDA-FEB-20

Page | 8

Stackoverflow Question Quality Analysis and Prediction

*Figure 6 Bar plot for the most common*

Exp: Here we can observe all the common words in the Title of the question’s in the stackoverflow dataset. This is represented using bar plot graph. The most common word is file and least common word is object.



IACSD-PG-DBDA-FEB-20

Page | 9

Stackoverflow Question Quality Analysis and Prediction

*Figure 7 Tree plot of the most common words in the Body attribute.*

Exp: Here we can observe all the common words in the Body of the question’s in the stackoverflow dataset. This is represented using tree graph. The most common word is gt and least common word is var. The most common words is represented by the size of box.

**2.4 Model Building:**

1. Train/Test split:

One important aspect of all machine learning models is to determine their accuracy. Now, in order to determine their accuracy, one can train the model using the given dataset and then predict the response values for the same dataset using that model

and hence, find the accuracy of the model. A better option is to split our data into



IACSD-PG-DBDA-FEB-20

Page | 10

Stackoverflow Question Quality Analysis and Prediction

two parts: first one for training our machine learning model, and second one for

testing our model.

• Split the dataset into two pieces: a training set and a testing set.

• Train the model on the training set.

• Test the model on the testing set, and evaluate how well our model did.

Advantages of train/test split:

• Model can be trained and tested on different data than the one used for training.

• Response values are known for the test dataset, hence predictions can be evaluated

• Testing accuracy is a better estimate than training accuracy of out-of-sample performance.

Machine learning consists of algorithms that can automate analytical model building. Using algorithms that iteratively learn from data, machine learning models facilitate computers to find hidden insights from Big Data without being explicitly programmed where to look.

We have used the following three algorithms to build predictive model.

2. Logistic Regression:

Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.



IACSD-PG-DBDA-FEB-20

Page | 11

Stackoverflow Question Quality Analysis and Prediction

*Figure 8 Example of logistic regression function.*

*Figure 9 The accuracy score of logistic regression model*



IACSD-PG-DBDA-FEB-20

Page | 12

Stackoverflow Question Quality Analysis and Prediction

*Figure 10 The confusion matrix of logistic regression model*

3. Naïve Bayes:

Naive Bayes is a kind of classifier which uses the Bayes Theorem. It predicts membership probabilities for each class such as the probability that given record or data point belongs to a particular class. The class with the highest probability is

considered as the most likely class.



IACSD-PG-DBDA-FEB-20

Page | 13

Stackoverflow Question Quality Analysis and Prediction

*Figure 11 Naïve Bayes formula.*

*Figure 12 The accuracy score of Naïve bayes model*



IACSD-PG-DBDA-FEB-20

Page | 14

Stackoverflow Question Quality Analysis and Prediction

*Figure 13 The confusion matrix of naïve bayes model*

**4. K nearest neighbor:**

K Nearest Neighbor Algorithm. K nearest neighbor algorithm is very simple. It works based on minimum distance from the query instance to the training samples to determine the K-nearest neighbors. The data for KNN algorithm consist of several

multivariate attributes name that will be used to classify.



IACSD-PG-DBDA-FEB-20

Page | 15

Stackoverflow Question Quality Analysis and Prediction

k' in KNN is a parameter that refers to the number of nearest neighbors to include in

the majority of the voting process.

*Figure 14 Example of KNN.*

*Figure 15 The accuracy score of KNN model*



IACSD-PG-DBDA-FEB-20

Page | 16

Stackoverflow Question Quality Analysis and Prediction

*Figure 16 The confusion matrix of KNN model*

**5. Decision Tree:**

The general motive of using Decision Tree is to create a training model which can use to predict class or value of target variables by learning decision rules inferred from prior data (training data). Each internal node of the tree corresponds to an

attribute, and each leaf node corresponds to a class label.



IACSD-PG-DBDA-FEB-20

Page | 17

Stackoverflow Question Quality Analysis and Prediction

*Figure 17 Example of decision tree.*



IACSD-PG-DBDA-FEB-20

Page | 18

Stackoverflow Question Quality Analysis and Prediction

*Figure 18 The accuracy score of decision tree model*

*Figure 19 The confusion matrix of decision tree model*



IACSD-PG-DBDA-FEB-20

Page | 19

Stackoverflow Question Quality Analysis and Prediction

**6. Linear SVC**

The objective of a Linear SVC (Support Vector Classifier) is to fit to the data you provide, returning a "best fit" hyperplane that divides, or categorizes, your data. From there, after getting the hyperplane, you can then feed some features to your classifier to see what the "predicted" class is. This makes this specific algorithm rather suitable for our uses, though you can use this for many situations.

*Figure 20 The data points for present in the dataset*



IACSD-PG-DBDA-FEB-20

Page | 20

Stackoverflow Question Quality Analysis and Prediction

*Figure 21 SVC gives us the decision boundary between the distinct data points*

Here is the output of training and testing the SVC model

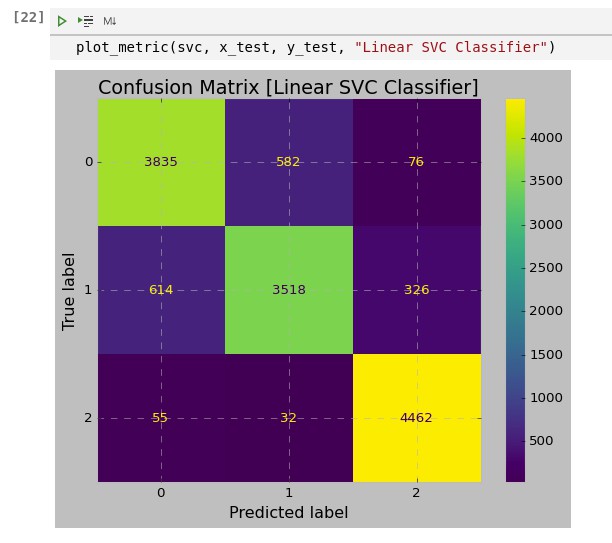
*Figure 22 The accuracy score of LinearSVC model*



IACSD-PG-DBDA-FEB-20

Page | 21

Stackoverflow Question Quality Analysis and Prediction



*Figure 23 Shows the confusion matrix of Linear SVC model*

**7. Random Forest Model:**

Random forest is a [supervised learning algorithm.](https://builtin.com/data-science/supervised-learning-python) The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models

increases the overall result.



IACSD-PG-DBDA-FEB-20

Page | 22

Stackoverflow Question Quality Analysis and Prediction

Put simply: random forest builds multiple decision trees and merges them

together to get a more accurate and stable prediction.

*Figure 24 Shows the working of the random forest model*

The output of the trained Random Forest Model

*Figure 25 The accuracy score of the random forest model*



IACSD-PG-DBDA-FEB-20

Page | 23

Stackoverflow Question Quality Analysis and Prediction

*Figure 26 The confusion matrix of the trained random forest model*

**9. XGBoost:**

[XGBoost is](https://xgboost.ai/) a decision-tree-based ensemble Machine Learning algorithm that uses a [gradient boosting framework.](https://en.wikipedia.org/wiki/Gradient_boosting) In prediction problems involving unstructured data (images, text, etc.) artificial neural networks tend to outperform all other algorithms or frameworks. However, when it comes to small-to-medium structured/tabular data, decision tree based algorithms are

considered best-in-class right now.



IACSD-PG-DBDA-FEB-20

Page | 24

Stackoverflow Question Quality Analysis and Prediction

*Figure 27 The Accuracy score of the XGBoost model*

*Figure 28 The confusion matrix of XGBoost*



IACSD-PG-DBDA-FEB-20

Page | 25

Stackoverflow Question Quality Analysis and Prediction

**10. Stacking Classifier:**

Stacking or Stacked Generalization is an ensemble machine learning algorithm.

It uses a meta-learning algorithm to learn how to best combine the predictions from two or more base machine learning algorithms.

The benefit of stacking is that it can harness the capabilities of a range of well-performing models on a classification or regression task and make predictions that have better performance than any single model in the ensemble.

The output of the trained Stacking Classifier model

*Figure 29 Models used in the stacking classifier*



IACSD-PG-DBDA-FEB-20

Page | 26

Stackoverflow Question Quality Analysis and Prediction

*Figure 30 Training the Stacking Classifier model*

*Figure 31 The Accuracy score of the Stacking Classifier*



IACSD-PG-DBDA-FEB-20

Page | 27

Stackoverflow Question Quality Analysis and Prediction

*Figure 32 The confusion matrix of the stacking classifier model*



IACSD-PG-DBDA-FEB-20

Page | 28

Stackoverflow Question Quality Analysis and Prediction

**3. User Interface**

After Training the models and finding out the best model to be Stacking Classifier we can go ahead with building the user interface for our models. This will give us a clean and simple way of accessing our models and make the predictions on our questions.

For building the user interface we adopted the **Flask Web Framework** which is very easy to use and robust.

Here are a few screenshots of the web app delivering the stackoverflow question quality prediction models.

*Figure 33 . The web application where the user can enter his/her questions and press the “Predict*

*Question Quality” Button*

*Figure 34 The web application with a sample question with it’s title and body*

IACSD-PG-DBDA-FEB-20

Page | 29

Stackoverflow Question Quality Analysis and Prediction

*Figure 35 This gives us the question’s quality and a comparison of the different models*

*Figure 36 For a different question the question’s quality changes, here it’s high quality*



IACSD-PG-DBDA-FEB-20

Page | 30

Stackoverflow Question Quality Analysis and Prediction

**4. Requirements Specification**

4.1 Hardware Requirement:

• 500 GB hard drive (Minimum requirement)

• 8 GB RAM (Minimum requirement)

• PC x64-bit CPU

4.2 Software Requirement:

• Windows/Mac/Linux

• Python-3.9.1

• VS Code/Anaconda/Spyder

• Python Extension for VS Code

• Libraries:

➢ Numpy 1.18.2

➢ Pandas 1.2.1

➢ Matplotlib 3.3.3

➢ Scikit-learn 0.24.1

➢ Flask 1.1.2

• Any Modern Web Browser like Google Chrome

➢ To access the web application written in Flask



IACSD-PG-DBDA-FEB-20

Page | 31

Stackoverflow Question Quality Analysis and Prediction

**5. Conclusion:**

• In this project we have built a system for classifying a question into

High Quality or Low Quality Question.

• For Model building we gathered the data from Kaggle.com

• We cleaned the data of any HTML tags. Performed Label Encoding on the response variable

• Built a dozen different models on the cleaned dataset. Found out

Stacking Classifier to be the best model.

• Built a webapp using the framework Flask to easily use the model to classify different questions.

• The app gives a comparison of the different trained models as well.

• Using the feedback obtained for the question classification the user can improve the question to get a High Quality rating.



IACSD-PG-DBDA-FEB-20

Page | 32

Stackoverflow Question Quality Analysis and Prediction

**6. Future Scope**

• The project currently doesn’t have a tag prediction feature which can

be implemented by taking the question’s title and body as input

• We can generate the question’s title from the if the body is large

enough.

• We can predict the amount of time the question will take to get answered.



IACSD-PG-DBDA-FEB-20

Page | 33

Stackoverflow Question Quality Analysis and Prediction

**7. References**

• Predicting the Quality of Questions on Stackoverflow, Antoaneta Baltadzhieva Grzegorz Chrupała (2015) <https://www.aclweb.org/anthology/R15-1005.pdf>

• Stackoverflow Question Quality Dataset - [https://www.kaggle.com/imoore/60k-stack-overflow-questions-](https://www.kaggle.com/imoore/60k-stack-overflow-questions-with-quality-rate)

[with-quality-rate](https://www.kaggle.com/imoore/60k-stack-overflow-questions-with-quality-rate)



IACSD-PG-DBDA-FEB-20

Page | 34

