By utilizing text classification algorithms, it is possible to considerably streamline the process of classifying consumer complaints into distinct specified categories, including debt collection, credit reporting, mortgage, and more. This strategy seeks to improve customer service operations' efficiency by utilizing databases of client complaints. The need for manual sorting and handling may decrease as the model gets more developed and turns into a completely automated system. Machine learning and AI carry the potential of improving customer satisfaction levels and eventually increasing revenue through the accurate classification of new complaints and their subsequent routing to the right resolution teams.

In order to speed up the procedure and prevent repetitive dataset uploads, I trained a text classifier on a sizable dataset of 4,055,791 consumer complaints reported to the CFPB by US financial institutions using Jupyter Notebook (Anaconda3).

Labeling complaints to a certain product is a project-managed problem statement. To find the optimum machine learning algorithm, several will be tried. With the use of text pre-processing and natural language processing, the model will categorize fresh complaints and evaluate its effectiveness.

I detail the necessary steps in the following step-by-step manual for text classification work on the Consumer Complaint Database:-

Everything we'll utilize to execute our project must be imported from the very beginning.

**Explanatory Data Analysis (EDA) and feature engineering are the first steps.**

After loading the dataset, browse it to learn about its distribution, columns, and structure.

Look at the distribution of categories in the target variable "Product."

Deal with missing data.

Perform appropriate feature engineering, including the selection of key features and TF-IDF features.To decrease the category of product complaints, we also employ the replace function.

**Text pre-processing, step two:**

Split the text into words or tokens by using tokenization.

Eliminating Stop Words: Get rid of everyday words that don't add much.

Label encoding: change the product's label to a numeric format.

For feature engineering, we employ TF-IDF and chi-Square in this case.

**The third step is choosing a multi-classification model.**

Create training and testing sets from the dataset. (Data from 25% test and 75% train)Select a text categorization algorithm that works well, like:

• Support vector machines (SVM) in linear form

• Classifier Random Forest

**Comparing model performance at step four:**

Using the text data that has been previously processed, create a number of classification models, and then test them using metrics like mean accuracy.

Cross-validate your model to determine how robust it is.

**Model Evaluation at step five:**

The optimal model should be chosen based on assessment metrics, and additional matrices like accuracy, precision, recall, and F1-score should also be evaluated. Confusion matrices can be used to visualize the outcomes.

**Prediction—Step 6:** Make the prediction. Revert the numerical forecasts to the category names that correspond to them.