Consumer complaints classification

Name:	**Anuradha Mahato**
Registration No./Roll No.:	19347
Institute/University Name:	IISER Bhopal
Program/Stream:	DSE
Problem Release date:	August 15, 2022
Date of Submission:	September 29, 2022

1 Introduction

The objective is to develop supervised machine learning frameworks to classify each complaint of the given corpus into one of five categories: credit reporting, debt collection, mortgages and loans, credit cards and retail banking using text feature extraction and text classification techniques.
The data is collected from a federal U.S. agency (Consumer Financial Protection Bureau) that acts as a mediator when disputes arise between financial institutions and consumers. The data included customer's complaints of one year from March 2020 to March 2021.

2 Methods

In the beginning of the project, visualization of distribution of the complaints data into each category was done. The training data consists of 158350 complaints and the test data consists of 4060 complaints. We observed that the data is skewed towards credit reporting with almost half of the text in this category.
Ten null complaints were removed since they did not give any information about their categorization. Text cleaning was done to remove the punctuation marks and make the data case-insensitive. The following techniques were employed to pre-process the cleaned text:

- **Word-Tokenization:** To split the complaints into smaller units to get meaningful words
- **Lemmatization**: To shorten the modified words to their word stem, base, or root form [1].
- **Stop-Words Elimination**: Removing words that do not add any valuable information and does not change the semantics of corpus. Stopwords are meaningless words that have low dis-

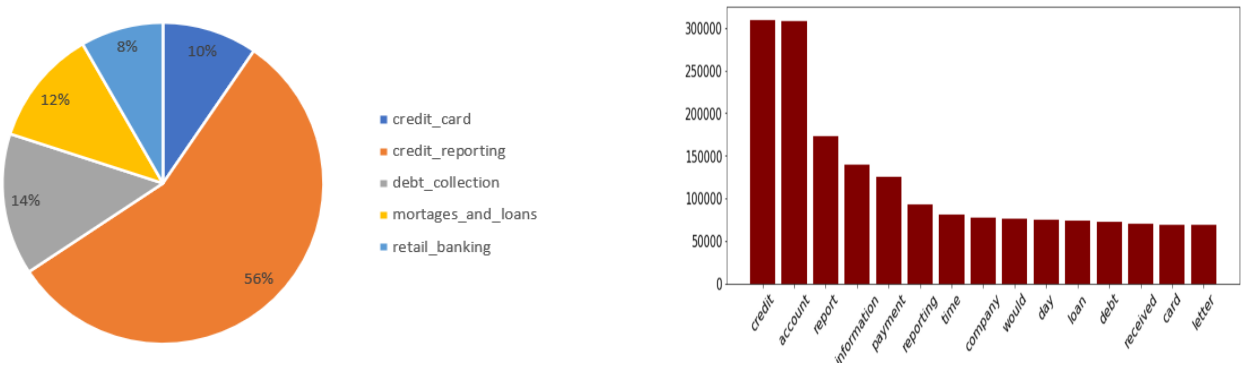

Figure 1: Distribution of categories of products and most frequent words

crimination power [2].
The mentioned techniques had been implemented using the **nltk** library of Python.

The feature extraction was done using **tf-idf, bag of words, and bag of words with ngrams of range (1,2), (2,3) and (1,3)**. They were used to transform the words into vectors for further processing of corpus of complaints.
To evaluate the performance of classifiers, 80% of complaints are used for training purpose and rest 20% complaints are used for cross validation with stratified class labels. A pipeline has been created using 5 models that are Logistic Regression, Decision Tree, Random Forest, Multinomial Naïve Bayes and k-Nearest Neighbor Classifier along with Grid Search parameter tuning technique. These models have been implemented using the **scikit-learn** library of Python.
https://github.com/shraddhaagarwal10/Consumer-complaints-classification

3 Evaluation Criteria

The pipeline prints the confusion matrix and classification report of the predicted class labels of validation set for each model. As evaluation metrics, macro-averaged f1-score and accuracy are used. As it's essential to get less number of false positives and false negatives (good precision and recall scores) for better predictions in each class, f1-scores are helpful.

The f1 score is the harmonic mean of recall and precision, with a higher score as a better model.
$f1 - score = \frac{2 \cdot precision \cdot recall}{precision + recall}$

Accuracy is fraction predicted correctly.
$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$

4 Analysis of Results

Logistic Regression with bag of words using n-gram of range (1,3) has given the best results in terms of both accuracy and F1-score. Word order is insignificant in tf-idf and bag of words because it uses unigrams by default, and only individual words are counted. Word order becomes significant when we change the range of n-grams to unigram, bigram, and trigram. When classifying the complaints, n-gram is used to take words into account in their actual order so that the context of the word is retained. The results infer that unigrams + trigrams (1,3) are more useful for finding find context than unigrams + bigrams (1,2) and bigrams + trigrams (2,3) when Used with LR, MNB or DT which are faster than KNN and RF.
We observe that Multinomial Naive Bayes produces the fastest results among all the algorithms as it has lower computational complexity but still produces reasonably good results in comparison to other algorithms that take much longer time. Decision tress perform poorest among all classification algorithms. For decision tree, random forest, k-nearest neighbor classifiers, tf-idf and bow unigram models outperformed bi/tri-grams which may be attributed to the individual word importance in these algorithms.
A comparative study has been conducted to look out the performance of different techniques. The comprehensive graph analysis indicates macro-averaged f1-scores of all five classifiers and all five methods of feature extraction Fig2.
Table1 summarizes macro averaged precision, recall and f1-score and accuracy for 5 models (i.e., LR: Logistic Regression, DT: Decision Tree Classifier, RF: Random Forest Classifier, MNB: Multinomial Naïve Bayes Classifier, KNN: k-Nearest Neighbor) using feature-extraction techniques (tf-idf, bow, bow with n-grams of various ranges) with grid search parameter tuning technique. The confusion matrix for the best performing classifier using best feature selection bow with unigrams + trigrams (1,3) is shown in Fig3.

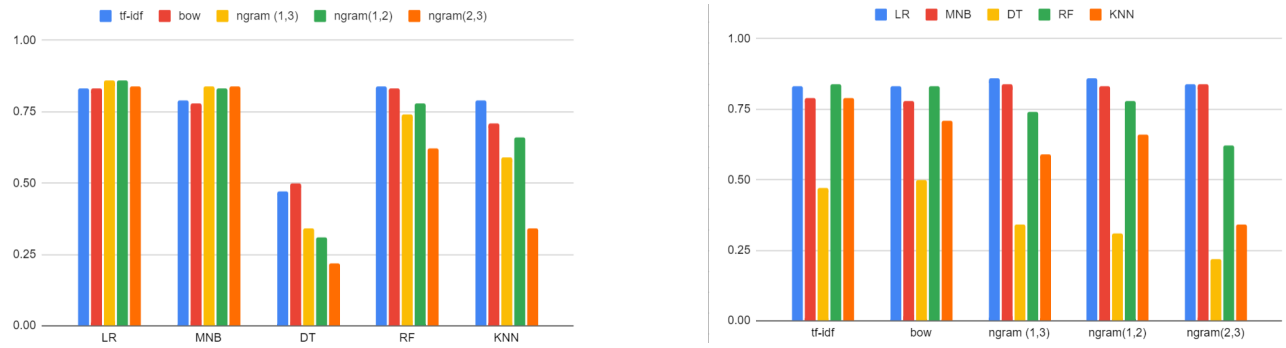

Figure 2: Comparison of Macro averaged F-1 scores using the different classifiers and feature extraction techniques

Table 1: Performance Metrics Of Different Classifiers Using Various Feature Extraction

Feature Extraction	Classifier	Precision	Recall	F-1 score	Accuracy	Execution Time (in sec)
TF-IDF	LR	0.81	0.85	0.83	0.86	343
	DT	0.54	0.45	0.47	0.64	1061
	RF	0.86	0.82	0.84	0.88	21250
	MNB	0.80	0.79	0.79	0.84	11
	KNN	0.82	0.77	0.79	0.84	16433
BOW	LR	0.82	0.83	0.83	0.86	3071
	DT	0.47	0.35	0.38	0.64	426
	RF	0.86	0.81	0.83	0.88	15964
	MNB	0.76	0.82	0.78	0.82	6
	KNN	0.79	0.66	0.71	0.80	21315
BOW N-GRAM (1,3)	LR	0.87	0.86	**0.86**	**0.90**	33196
	DT	0.55	0.32	0.34	0.61	4292
	RF	0.72	0.78	0.74	0.79	84417
	MNB	0.84	0.84	0.84	0.88	38
	KNN	0.76	0.53	0.59	0.74	27918
BOW N-GRAM (1,2)	LR	0.86	0.85	0.86	0.89	17591
	DT	0.54	0.30	0.31	0.59	1112
	RF	0.76	0.81	0.78	0.83	15675
	MNB	0.81	0.85	0.83	0.87	22
	KNN	0.78	0.60	0.66	0.77	29879
BOW N-GRAM (2,3)	LR	0.86	0.83	0.84	0.89	15035
	DT	0.57	0.24	0.22	0.58	2006
	RF	0.60	0.76	0.62	0.66	31111
	MNB	0.84	0.84	0.84	0.88	38
	KNN	0.69	0.32	0.34	0.62	15231

Predicted Class

Actual Class	credit_reporting	credit_card	retail_banking	debt_collection	mortages_and_loans
credit_reporting	2168	449	43	91	284
credit_card	130	17118	324	163	44
retail_banking	94	926	3303	143	48
debt_collection	63	416	106	3035	83
mortages_and_loans	180	105	11	51	2292

Figure 3: Confusion Matrix of LR using BOW with ngram of range (1,3)

5 Discussions and Conclusion

LR with all feature-selection techniques gives better results than other models, and, among the feature-extraction techniques, n-grams give the best results when combined with Logistic Regression. This is the best fit model on the validation set; hence, it is considered the high-performing model to predict the class labels of the test set. Hyperparameter tuning employs grid search to find the best parameters while trying every possible combination. However, the worst classification was performed by Decision Tree as it has no randomization or good split criteria and thus achieved the lowest accuracy and F1-score. Additionally, DT achieves the highest error rate with a high number of false positive and false negative cases compared to other classifiers.

We can conclude that the company can effectively classify complaints of its consumers into the products by determining the context of the given text. For selecting the right context, they can use Logistic Regression Classification algorithm with n-grams which fits the data set correctly. This technique would assist the marketing department to identify the category of complaints. After having an inspection over these complaints, the organization can improve the particular product according to its customer's needs.

This work could be further enhanced by developing a new algorithm which would classify the data with higher accuracy and F1-score. None of the classifiers achieved all the parameters to the satisfactory level. So, deep learning approaches can be applied to improve the classification performance.

A team of two students did this project. Shraddha implemented all the classifiers using tf-idf and bag of words with ngram of range (1,3) and MNB and LR classifiers of ngram of range (2,3) and Anuradha implemented the bow unigram model, bow with ngram(1,2) for all classification algorithms and ngram(1,3) for DT, KNN and RF. The codes for data analysis, building pipeline, visualizing results, and generating test class label file, were written together.

References

[1] Tonmoy Hasan and Abdul Matin. Extract sentiment from customer reviews: A better approach of tf-idf and bow-based text classification using n-gram technique. In *Proceedings of International Joint Conference on Advances in Computational Intelligence*, pages 231–244. Springer, 2021.

[2] Hassan Saif, Miriam Fernández, Yulan He, and Harith Alani. On stopwords, filtering and data sparsity for sentiment analysis of twitter. 2014.