Text Mining on Consumer Complaints in Financial Products

Team 6: Xinlian Huang, Lulu Zhu, Jiahui Bi, Bowen Lu

I. Introduction

i. Background

Beginning in 2007, the United States faced the most severe financial crisis since the Great Depression. Millions of Americans saw their home values drop, their savings shrink, their jobs eliminated. Credit dried up, and countless consumer loans—many improperly made to begin with—went into default. Today, we're still in the process of recovering. After that, a U.S. government agency, Consumer Protection Financial Bureau, was created to collect customers complaints regarding financial products to promote fairness and transparency for financial products and services. By submitting a complaint, consumers can be heard by financial companies, get help with their own issues, and help others avoid similar ones.

ii. Motivation

Consumer can choose product, specific issue and submit complaints in the system. This process is really tedious. First, there are lots of product and issue classification. Also, these classifications are really confused for consumers who don't have lots of knowledge in the financial market. So, in order to improve consumer experience and provide accurate classification result for cfpb, we want to build a text auto classification system.

iii. Objectives

Build text auto-classification system, complaints can be accurately assigned to specific department or team manually.

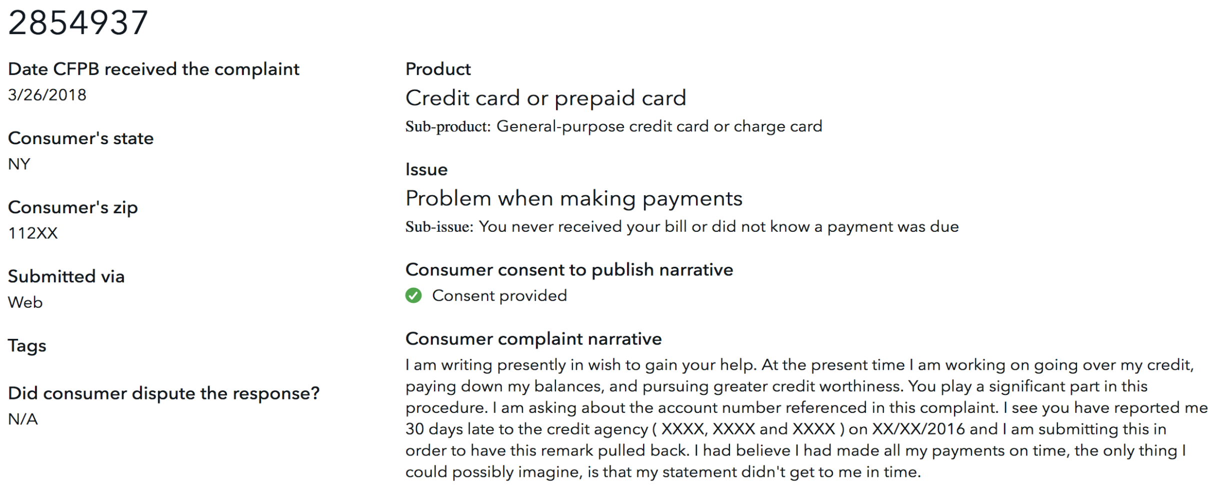
II. Data Scope

i. Data source

The dataset is from Consumer Financial Protection Bureau(CFPB), an official government website of United States. (Link: https://www.consumerfinance.gov/)

Consumers could submit their complaints on certain financial products, issues, or companies, we will make use of consumers' narratives to do the text mining.

ii. Data sample



2854937

Date CFPB received the complaint
3/26/2018

Consumer's state
NY

Consumer's zip
112XX

Submitted via
Web

Tags

Did consumer dispute the response?
N/A

Product
Credit card or prepaid card
Sub-product: General-purpose credit card or charge card

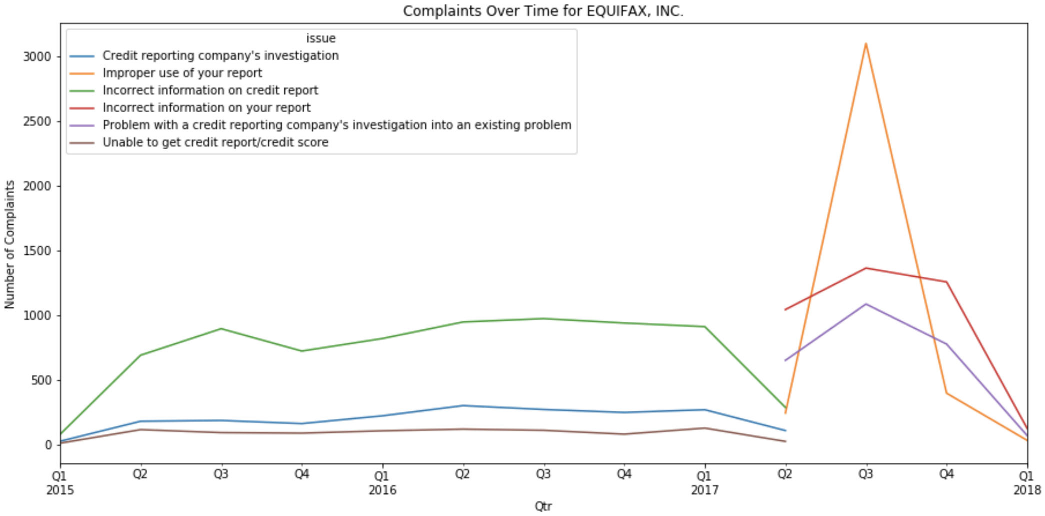
Issue
Problem when making payments
Sub-issue: You never received your bill or did not know a payment was due

Consumer consent to publish narrative
✓ Consent provided

Consumer complaint narrative
I am writing presently in wish to gain your help. At the present time I am working on going over my credit, paying down my balances, and pursuing greater credit worthiness. You play a significant part in this procedure. I am asking about the account number referenced in this complaint. I see you have reported me 30 days late to the credit agency (XXXX, XXXX and XXXX) on XX/XX/2016 and I am submitting this in order to have this remark pulled back. I had believe I had made all my payments on time, the only thing I could possibly imagine, is that my statement didn't get to me in time.

iii. Exploratory Data Analysis

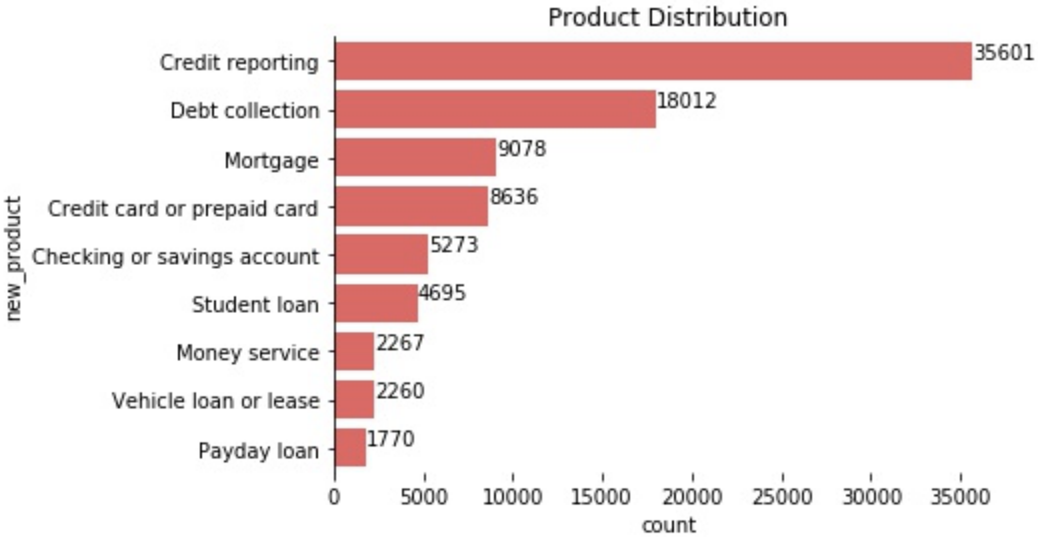
1. Timeline



Take Equifax for example, Equifax is one of the three major consumer credit reporting agencies, we can see the major issue is about credit reporting. Two interesting things in here, the first one is there is a huge increase of the improper use of report during 2017 third quarter. As we all know that there is a data breach scandal for Equifax in September 2017. Another thing is there is a gap at second quarter last year, that's because CFPB modified their classification system on Apr. 24th, 2017. So, we decide to cut off the dataset and only use complaints after this date. In the end, our dataset contains over 80,000 complaints with their product and issue.

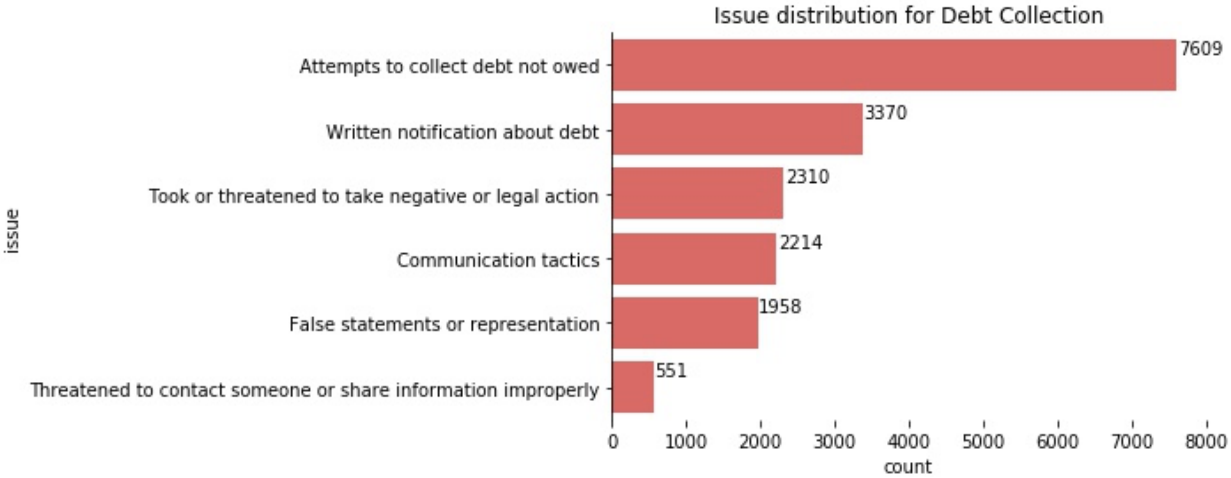
2. Product distribution

For financial companies, financial products or services are usually the important sources of total income. Therefore, we research on the relationships between those products and their corresponding issues, trying to give financial companies some hints about what products received the most amount of complaints, what's the most frequent reported issue for a certain product.



3. Issue distribution

After checking the issue distribution and relationship between product and issue. It's obviously that specific issues are related to specific product. Take Debt Collection for instance, it has six issues within this product.

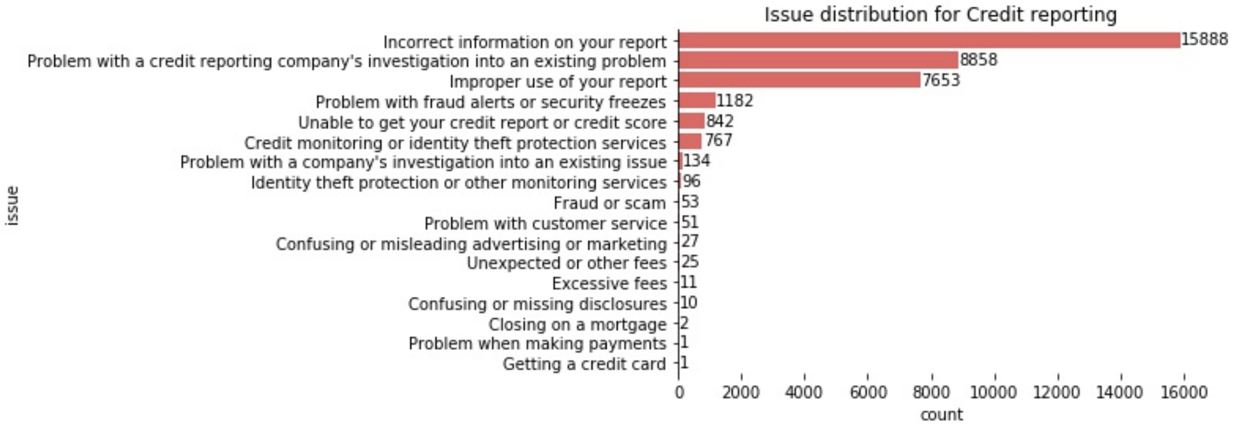


iv. Relabel

1. Product

As shown in the above figure, we have 9 products, and we combine three of them, “student loans”, “vehicle loans” and “payday loans” as “loans” since they are quite similar.

2. Issue



Take credit reporting as instance, several issues have very low amount with chaotic information, we combine all the issues whose amount is below 200 to “others”. Similar treatment for other products.

3. Product and issue combination

Since specific issues are related to specific product, we combine product add issue as our label to make our classification label more meaningful.

III. Modeling

i. Naive Bayes

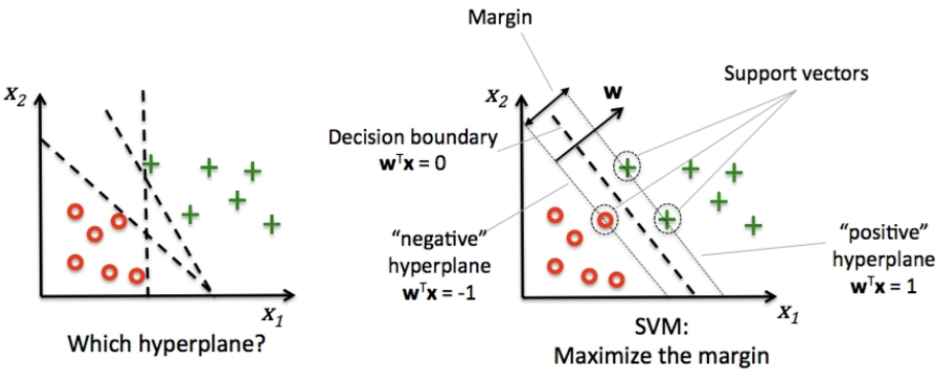
Naive Bayes classifiers are a family of simple "probabilistic classifiers "based on applying Bayes' theorem with strong (naive) independence assumptions between the features [1]. In our model, what we will get is the probability of a topic given a set of words.

$$\underbrace{P(c \mid x)}_{\text{Posterior Probability}} = \frac{\overbrace{P(x \mid c)}^{\text{Likelihood}}\,\overbrace{P(c)}^{\text{Class Prior Probability}}}{\underbrace{P(x)}_{\text{Predictor Prior Probability}}}$$

$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

ii. Support Vector Machine

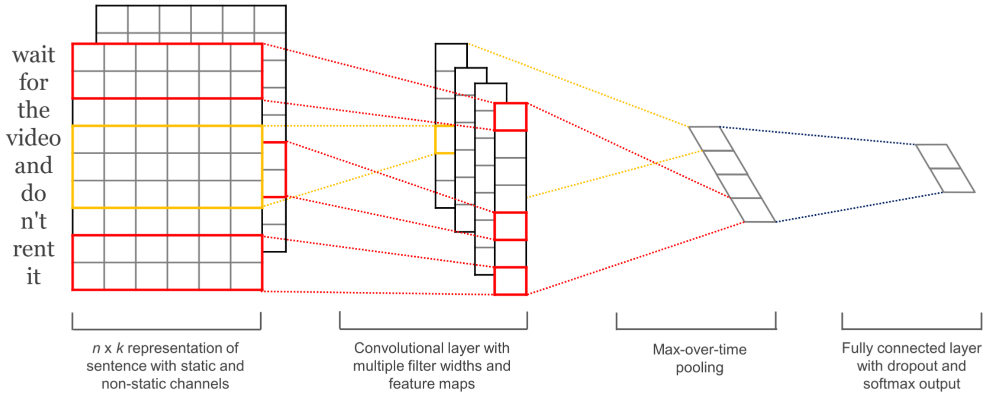
A support vector machine (SVM) constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like outlier detection. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier [2].



iii. Convolutional Neural Network

In machine learning, a convolutional neural network (CNN) is a class of deep, feed-forward artificial neural networks that has successfully been applied to analyzing visual imagery [3]. In the text mining, we can also use similar technique by adding word embedding layer. Different embedding methods are used and compared. Specific performance comparison will be discussed in detail in the section of Evaluation.

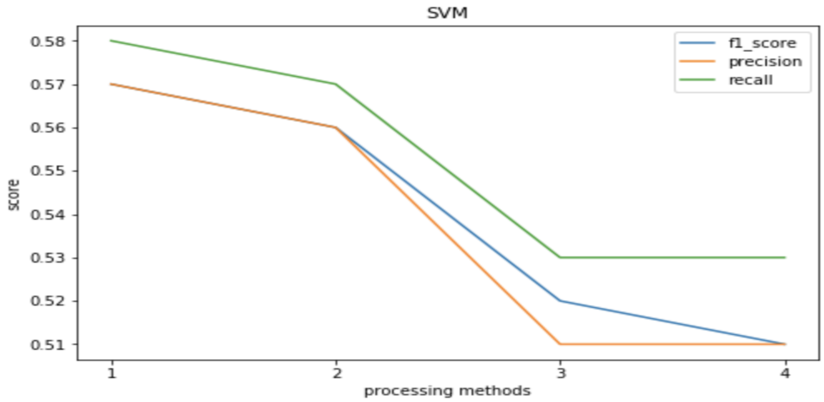
In the CNN model, we used 3 kinds of filters: filter for unigram, for bigram, and for trigram. For each kind, there are 64 of them.

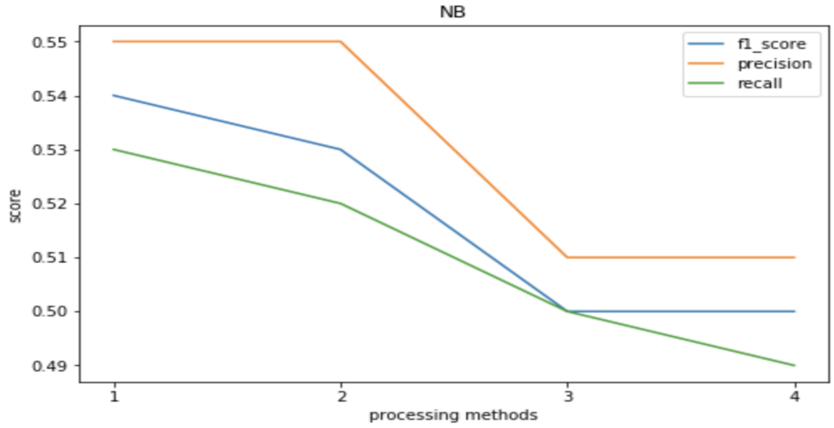

wait
for
the
video
and
do
n't
rent
it
n x *k* representation of sentence with static and non-static channels
Convolutional layer with multiple filter widths and feature maps
Max-over-time pooling
Fully connected layer with dropout and softmax output

IV. Evaluation

i. Comparison within each algorithm

1. Tf_idf modification methods for SVM and NB





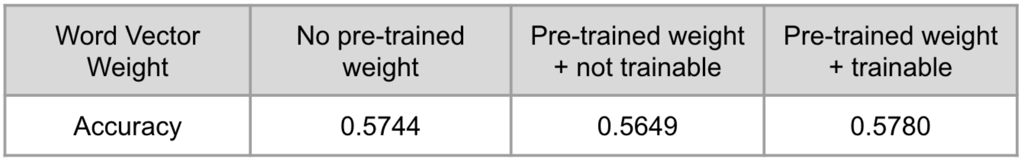
For SVM and NB, an important step is to create the Term Frequency and Inverse Document Frequency (tf-idf) Matrix. In our project, we tried four different methods to do this.

- only tokenize and vectorize
- tokenize and vectorize + remove stopwords
- tokenize and vectorize + remove stopwords + remove non-english words
- tokenize and vectorize + remove stopwords + remove non-english words + stem

The performance of each method and algorithm is show in the graphs. Initially, we expected the fourth method to get the best performance because, stop words or stemming words usually don't contain too much useful information. However, the reality is that the first method performs the best. By analyzing our dataset, we found out the reason: the narratives we had are really short and in oral English style. Therefore, when we removed all stop words or stemming words, not too many words are left.

Another interesting point here is that SVM has slightly better performance than NB in general. The reason is that NB treats every word independently while SVM takes the interactions between words into consideration. Those interactions matter because they conform with the language habits: the word do have influence on the other words in a sentence.

2. Embedding methods for CNN



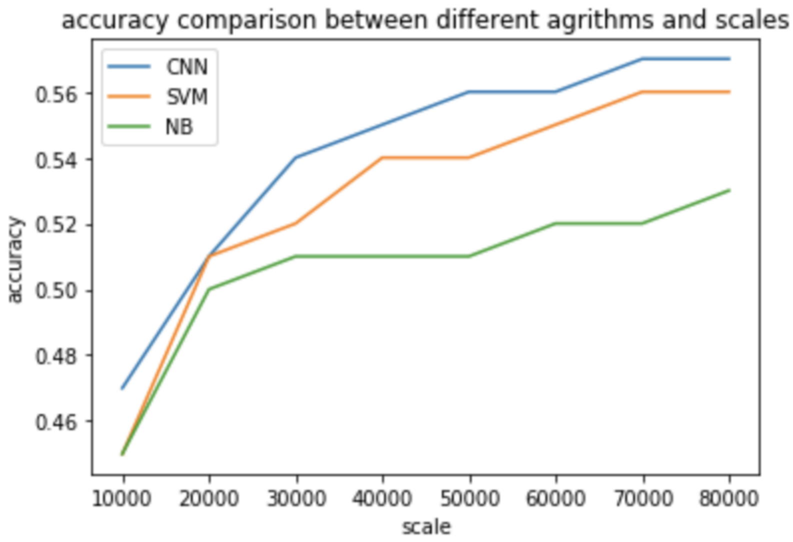
Word Vector Weight	No pre-trained weight	Pre-trained weight + not trainable	Pre-trained weight + trainable
Accuracy	0.5744	0.5649	0.5780

For CNN, we tried three embedding methods for word vectors:

- No pre-trained weight: train the word vector from all 0.
- Pre-trained weight + not trainable: use pre-trained word vector from GloVe, not changeable in the training process.
- Pre-trained weight + trainable: use word vector from Glove, but the weight can change during the training process.

The accuracy of each method is shown in the table above. Even though those accuracies are really close to each other, we can still learn something from them: the accuracy of the third method is the highest. The reason may be that the third method combines the advantages of former two methods. First, it uses less computing resource compared to the first method because it doesn't have to train the word vector from scratch. Second, it is more suitable for our own dataset compared to the second method because pre-trained weight is trained from more general text rather than text of financial complaints.

ii. Comparison between algorithms



Besides those comparison between different models, we still want to find out whether data scale has any impact on our results. We split the dataset into 8 subsets, which contain data range from 10,000 to 80,000, and train the 8 datasets again in our three models, the accuracies are shown as in the following graph.

For our three models, the accuracy of three models all increase along with the increasing data scale, and the accuracy reaches over 0.52. But by comparing the three models, CNN still has the highest accuracy. When the data scale is 80,000 the accuracy can reach around 0.57, but Naive Bayes still ranks the last for its relative low accuracy.

iii. Results Analysis

Overall, we have accuracy around 0.56 among three models, but there are still some problematic predictions. Among all labels, “('Money transfer, virtual currency, or money service', 'service')” is the only one that is predicted totally wrong among three models.

1. Pattern Recognition

But for single specific model, situation differs from each other. For example, in SVM, “Checking or savings account, Opening an account” with 15 support records. However, the true labels for these 15 records do not belong to this category. Also, in Naives Bayes, most of incorrect classifications are labeled with “others”.

2. Possible Underlying Reasons

For the label “('Money transfer, virtual currency, or money service', 'service')”, we extract all narratives which are predicted in this category. By looking through these narratives, we find out that, these narratives containing a lot of words, such as “bank account”, “checking account”.

It makes sense because money transfer issues must be related with checking or saving account. If the user takes a lot of efforts mentioning “checking account” or “bank account”, the classifier will easily classify the record into “Checking or saving account”.

For Naives Bayes, a lot of records, their issues are classified into “others” even if the product type is predicted correctly. The reason for these narratives might be: contents contain too much information and topics are overlapped with each other.

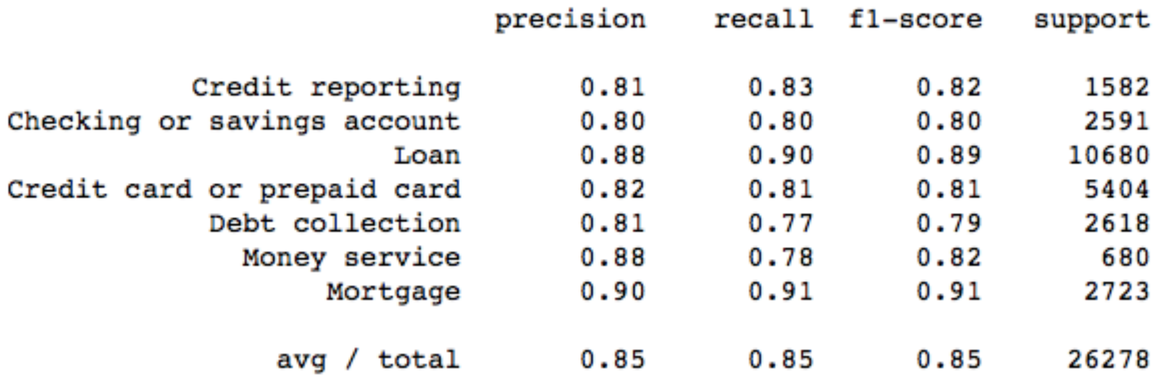
V. Two-Step Prediction

i. Method

In our previous prediction, we use the combination of product with issue as our label, the accuracy approaches 0.58 in CNN. In order to have a better result and explore the possible solutions, we try to do the prediction by two steps. First, use the product as our label to let the classifier recognize the product type. Second, use the predicted product type to shrink our scope. Within the product, predict the issue.

ii. Results

The classification report in step one is shown as below.

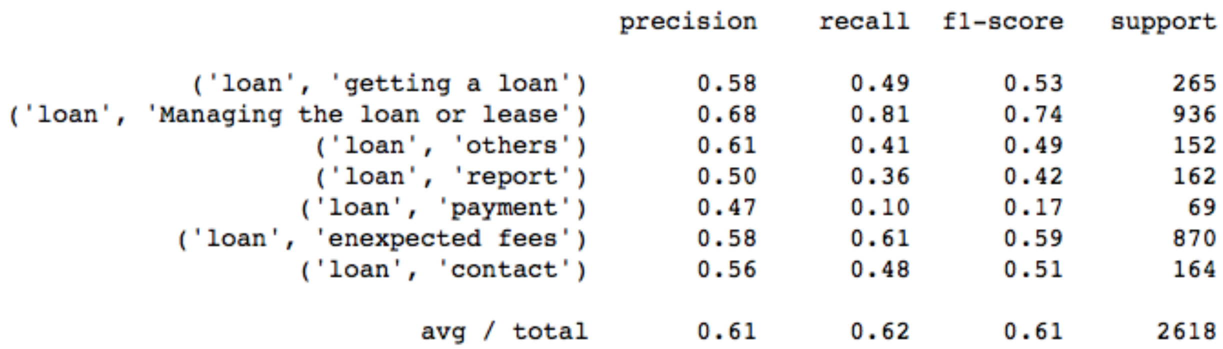


```
                             precision    recall  f1-score   support

           Credit reporting       0.81      0.83      0.82      1582
Checking or savings account       0.80      0.80      0.80      2591
                       Loan       0.88      0.90      0.89     10680
Credit card or prepaid card       0.82      0.81      0.81      5404
            Debt collection       0.81      0.77      0.79      2618
              Money service       0.88      0.78      0.82       680
                   Mortgage       0.90      0.91      0.91      2723

                avg / total       0.85      0.85      0.85     26278
```

Among the test set, average precision reaches as high as 0.85, single product prediction also has a pretty high f1 score. Then based on the certain product, we use the classifier to predict the issue within the product. Here we use “Loan” as an example in the trial. Following shows the classification report for “Loan”.



```
                                   precision    recall  f1-score   support

          ('loan', 'getting a loan')       0.58      0.49      0.53       265
('loan', 'Managing the loan or lease')       0.68      0.81      0.74       936
                  ('loan', 'others')       0.61      0.41      0.49       152
                  ('loan', 'report')       0.50      0.36      0.42       162
                 ('loan', 'payment')       0.47      0.10      0.17        69
         ('loan', 'enexpected fees')       0.58      0.61      0.59       870
                 ('loan', 'contact')       0.56      0.48      0.51       164

                         avg / total       0.61      0.62      0.61      2618
```

Within a specific product, the issues are predicted more accurately. The average precision is 0.61 and f1 score is 0.61. But within each specific category, some are predicted still bad, such as ('loan', 'payment').

VI. Future work

The "narrative" column is made up of large volumes of unstructured data, so our models need to deal with inputs of high dimensions and low quality. We learned from other resources about the solutions of enhancing the accuracy and time efficiency. The following two ideas may be helpful for further improvement.

i. Dimension reduction: Truncated SVD

The number of tokens in all narratives is more than 53,000, making the tf-idf as a large sparse matrix generated for NB or SVM. Besides, many of the words in narratives are irrelevant to the meaning conveyed. Dimension reduction helps to select a small group of more important words for the classification.

Truncated SVD is a variant of Singular Values Decomposition which only computes K largest singular values. Different with traditional PCA, this estimator does not center the data and directly use sample matrices instead of sample covariance matrices. So it works with scipy.sparse matrices efficiently and returns low dimensional matrices.

For instance, in Latent Semantic Analysis, truncated SVD works on tf-idf matrices as returned by the vectorizers in sklearn.feature_extraction.text.

ii. Gradient Boosting Decision Trees

Gradient Boosting Decision Trees (also called Gradient Boosting Machine) is a series of machine learning algorithms for regression and classification, which is efficient in high dimension analysis.

As known, traditional decision tree is a relatively weak prediction model and easy to result in overfitting. Boosting techniques is introduced for ensembling a bunch of decision tree models and iteratively adding new models to predict the residuals of preceding models until no further improvement made. Gradient Boosting means that the gradient descent algorithm is applied in the iteration to minimize the error.

There are three popular models of Gradient Boosting Decision Trees: XGBoost, LightGBM, and CatBoost. XGboost is an advanced implementation of gradient boosting algorithm, including features of stochastic, regularization, parallel processing, continuous training, etc. LightGBM is a fast, distributed, lower memory required gradient boosting algorithm. Tree splitting method is one of the main difference between XGboost (depth-wise) and LightGBM (leaf-wise). CatBoost works well with categorical variables like text, audio, image, etc.

VII. Business Values

i. For consumers

Large amount of consumers without sufficient knowledge on finance may be hard to understand the distinction among issues. If a complaint narrative is automatically classified based on the text mining, the risk of throwing complaints into wrong issues will be decreased. It is a good news for consumers that they will no longer need to manually select issues.

ii. For financial institutes

Our work classified the complaints as specific products+issues, and part of them are predicted differently compared with their original labels. This is nothing strange for misclassification, but some may provide insights for financial institutes which receiving complaints.

For instance, if large amount of complaints of Issue A frequently classified as another seemingly unrelated Issue B for a consistent period, the company being complained may investigate the reason and discover that the crux of Issue A is actually in Issue B. This can also be applied to forecast the possibly troublesome products or issues in the future.

iii. For CFPB

We are willing to give them several advice of improving their complaint submission system according to our experiences. They should adjust the category of products and issues since there are some overlapping among the issues of different products. Multilabel should also be adopted for enhancing classification performance and further analysis.

Most important thing is, that if automatic classification model applied, the issue of complaints will be immediately predicted by algorithms when consumers just submitting the narratives.

VIII. Reference

1. Naive Bayes https://en.wikipedia.org/wiki/Naive_Bayes_classifier
2. Support Vector Machine https://en.wikipedia.org/wiki/Support_vector_machine#Definition
3. Convolutional neural network https://en.wikipedia.org/wiki/Convolutional_neural_network