CONSUMER COMPLAINTS CLASSIFICATION USING TRADITIONAL MACHINE LEARNING AND DEEP LEARNING MODELS

MS-BANA CAPSTONE PROJECT

SUBMITTED BY: KUMARI, PRIYA(M13410586)

GUIDED BY: DR. PENG WANG AND DR. YAN YU

ABSTRACT

Unstructured text data is everywhere on internet in the form of emails, chats, social media posts, complaint logs, and survey. Extracting texts and classifying them can generate a lot of useful insights, which can be used by businesses to enhance decision-making. Text classification is the process of categorizing text into different predefined classes. By using Natural Language Processing (NLP), text classifiers can automatically analyze text and then assign a set of pre-defined tags or categories based on its content. Lately, deep learning approaches are achieving better results compared to previous machine learning algorithms on tasks like image classification, natural language processing, face recognition, etc. The success of these deep learning algorithms relies on their capacity to model complex and non-linear relationships within the data. This study would cover supervised learning models and deep learning models for multi-class text classification and would investigate which methods are best suited to solve it. The classifier assumes that each new complaint is assigned to one and only one category.

Contents

INTRODUCTION

Text classification is one of the most important tasks in Natural Language Processing. It is the process of classifying text strings or documents into different categories, depending upon the contents of the strings. However, most of text classification articles are binary text classification such as email spam filtering (spam vs. ham), sentiment analysis (positive vs. negative). In most cases, our real-world problems are much more complicated than that. Text classifiers can be used to organize, structure, and categorize pretty much anything. For example, new articles can be organized by topics, support tickets can be organized by urgency, chat conversations can be organized by language, brand mentions can be organized by sentiment, and so on [3].

Most text classification systems can be deconstructed into the following four phases: Feature extraction, dimension reductions, classifier selection, and evaluations. In this study three phases are covered excluding dimension reductions. The text classification can be done using machine learning models such as conventional supervised learning models and deep learning models. Traditional classifier models such as Logistic Regression, Random Forest, Gradient Boosting and SVM can also be used for text classification. Different types of deep learning models can be applied in text classification problems. The two main deep learning architectures used in text classification are Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN).

The rapid increase in the quantity of customer data has promoted the necessity to analyze these data. Recent progress in text mining has enabled analysis of unstructured text data such as customer suggestions, customer complaints and customer feedback. Customer satisfaction is not an absolute scenario, but very much depends on interactions, feedback, praise, and complaints. Complaints must be looked at in a constructive, positive, and professional perspective [1]. Businesses always strive to keep customers satisfied. Consumer complaints often lead to a loss of a consumer if those complaints are not dealt properly. These complaints, if genuine, need to be addressed by 'User Experience' department of a company.

DATASET AND PROBLEM STATEMENT

The dataset used for this analysis is taken from Kaggle. Link: https://www.kaggle.com/cfpb/us-consumer-finance-complaints. In 2011, Congress created the CFPB to ensure the protection of consumer interests in many financial markets. The CFPB receives and processes consumer complaints pertaining to various financial services, including credit cards, mortgages, bank accounts, student loans, consumer loans, credit reports, payday loans, and debt collection. The CFPB updates raw consumer complaints data every night and makes it publicly available to download from the CFPB's website (http://www.consumerfinance.gov/data-research/consumer-complaints/). This database, believed to be the largest public collection of consumer financial complaints, includes basic information about the complaints such as submission date, consumer's zip code, the company, the product type, the relevant issue, the consumer narratives, and how company has addressed the complaint. [2]

In this study, consumer complaints are classified in 11 pre-defined categories with consumer complaints narrative as input. For the classification various traditional machine learning and deep learning models are covered.

EXPLORATORY DATA ANALYSIS

The consumer complaints per category is captured in Fig 1.

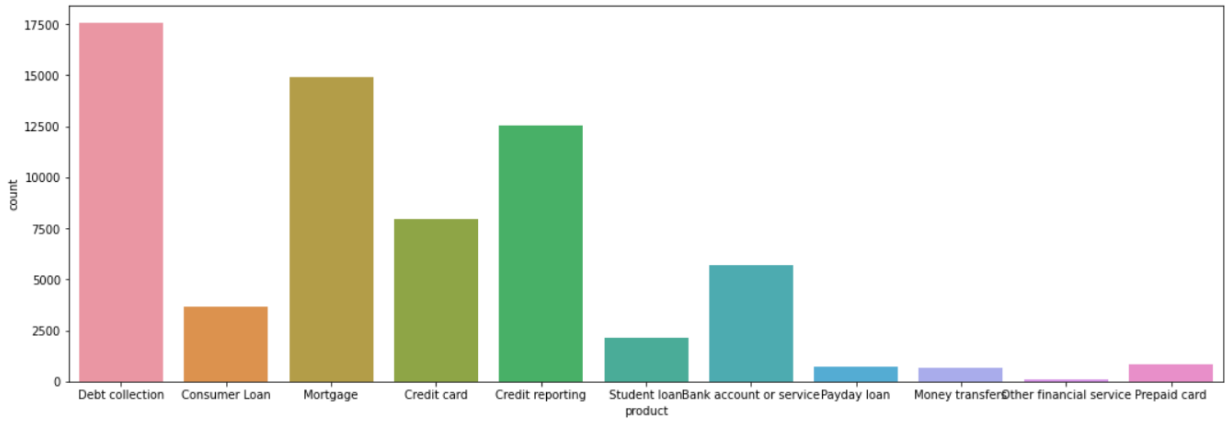

Fig 1. Consumer complaints per category

From the above figure it is evident that most of the consumer complaints belongs to Debt Collection, Mortgage, and Credit reporting categories. The word cloud for these categories is represented in Fig 2, Fig 3, and Fig 4.

Fig 2. Word cloud for Debt Collection

Fig 3. Word cloud for Mortgage

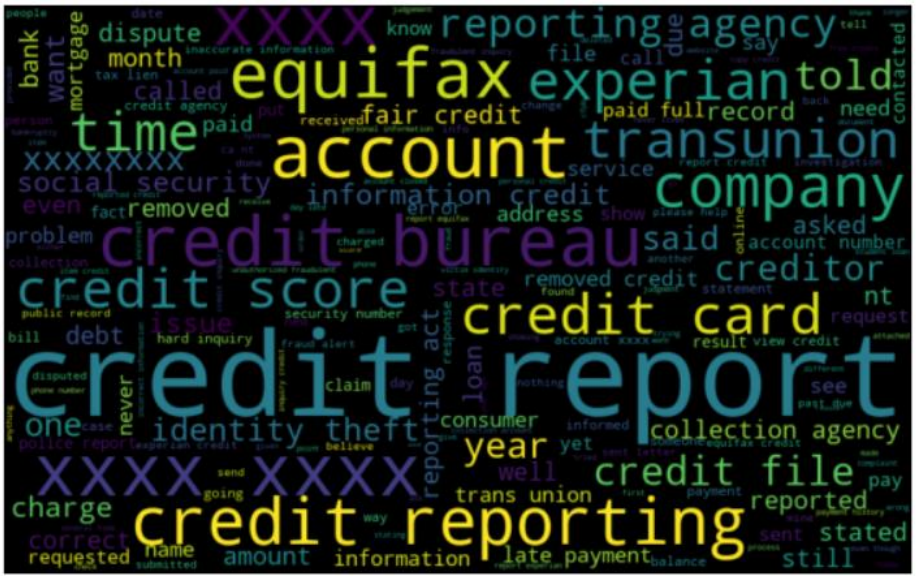

Fig 4. Word cloud for Credit Reporting

Fig 5 represents how many consumer complaints disputed per category. Category Debt collection and Mortgage has higher number of disputes.

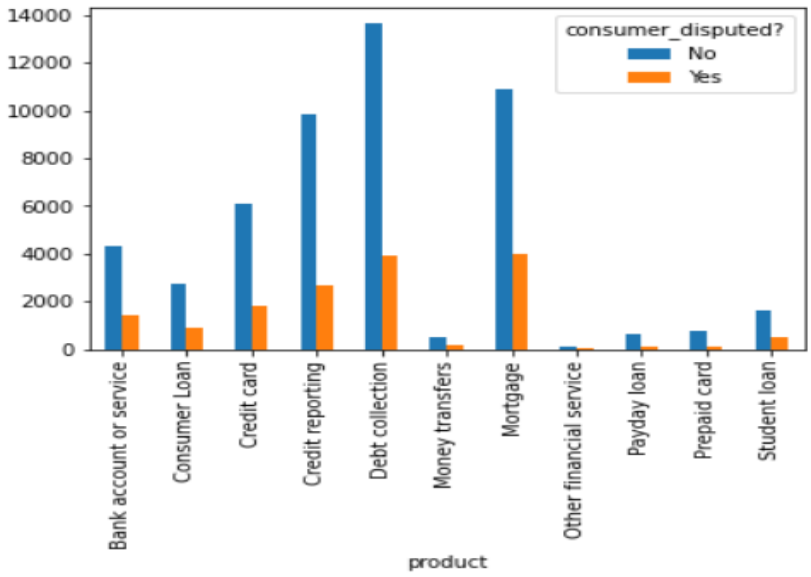

Fig 5. Consumer Disputed per category

The number of cases where timely response was provided or not is presented by Fig 6.

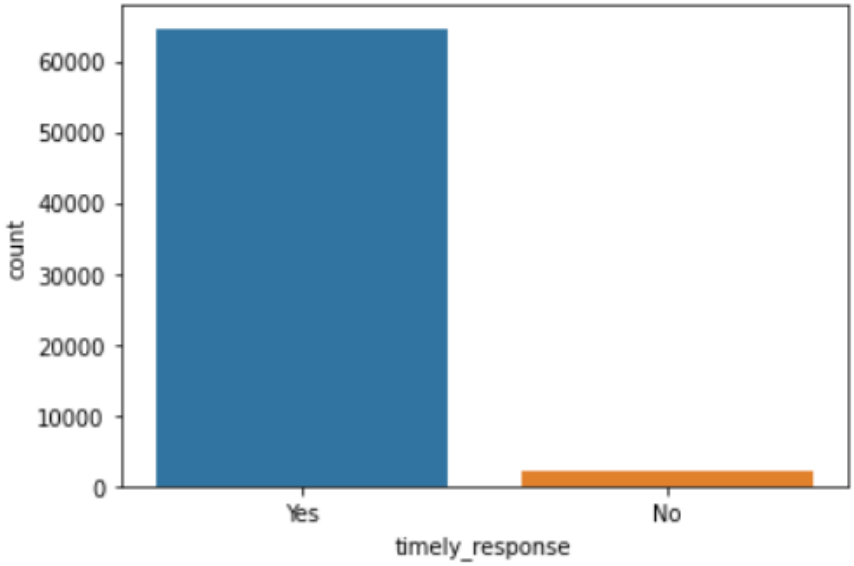

Fig 6. Timely response provided to consumer or not

TEXT CLEANING AND PRE-PROCESSING

In Natural Language Processing (NLP), most of the text and documents contain many words that are redundant for text classification, such as stop words, misspellings, slangs, etc. In many algorithms like statistical and probabilistic learning methods, noise and unnecessary features can negatively affect the overall performance. So, the elimination of these features is extremely important.[6]

First step is removing unnecessary spaces and special characters. Text documents generally contains characters like punctuations or special characters, and they are not necessary for text mining or classification purposes. Although punctuation is critical to understand the meaning of the sentence, but it can affect the classification algorithms negatively.

Text and document classification are usually affected by the noisy nature (abbreviations, irregular forms) of the text corpora, so in next step stop words are removed.

Sentences can contain a mixture of uppercase and lowercase letters. Multiple sentences make up a text document. To reduce the problem space, the next step is to reduce everything to lower case.

Lemmatization is the process of converting a word to its base form. The difference between stemming and lemmatization is, lemmatization considers the context and converts the word to its meaningful base form, whereas stemming just removes the last few characters, often leading to incorrect meanings and spelling errors [4]. In this study TextBlob library is used for lemmatization.

FEATURE ENGINEERING

The next step is the feature engineering step. In this step, raw text data will be transformed into feature vectors and new features will be created using the existing dataset. Vectorization is done using TF-IDF vectorizer.

TF-IDF score represents the relative importance of a term in the document and the entire corpus. TF-IDF score is composed by two terms: the first computes the normalized Term Frequency (TF), the second term is the Inverse Document Frequency (IDF), computed as the logarithm of the number of the documents in the corpus divided by the number of documents where the specific term appears.

TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)

IDF(t) = log_e(Total number of documents / Number of documents with term t in it)

Next, the target category classes are encoded as target labels with value between 0 and n_classes-1 [5].

WORD EMBEDDINGS

A word embedding is a form of representing words and documents using a dense vector representation. The position of a word within the vector space is learned from text and is based on the words that surround the word when it is used. Word embeddings can be trained using the input corpus itself or can be generated using pre-trained word embeddings such as Glove, FastText, and Word2Vec. For this study Glove is used.

The advantages of using Glove is

- It captures the position of the words in the text (syntactic)
- It captures meaning in the words (semantics)
- Trained on a huge corpus [6]

TRADITIONAL MODELS

To start model building for text classification, first, the classification task was performed using traditional machine learning models. Different traditional text classification methods have been explored in this study, each using a Tfidf and Count Vectorizer feature extraction, and machine learning pipeline.

Machine learning pipeline that remembers the complete set of preprocessing steps in the exact same order, so that whenever any new data point is introduced, the machine learning pipeline performs the steps as defined and uses the machine learning model to predict the target variable.

Table 1. Classification accuracy of Traditional models

Models	Accuracy	
	Count Vectorizer	Tfidf Vectorizer
Logistic Regression	85%	85.1%
Naïve Bayes	68.6%	69.1%
Support Vector Machine	85%	84.6%
Random Forest	77%	78%

Since Logistic regression and Support Vector Machine (Linear SVC) models are equally good for this dataset, in next step, classification reports for both are printed. The Precision, Recall, F-Measure and Accuracy are selected as evaluation metrics for the classifier.

Table 2. Classification report of Logistic Regression model

	precision	**recall**	**f1-score**	**support**
Debt Collection	0.82	0.80	0.81	1428
Consumer Loan	0.80	0.61	0.69	920
Mortgage	0.81	0.82	0.81	1982
Credit card	0.87	0.86	0.86	3132
Credit reporting	0.82	0.91	0.86	4388
Student Loan	0.81	0.55	0.65	166
Bank account or service	0.92	0.95	0.94	3730
Payday Loan	0.00	0.00	0.00	27
Money Transfers	0.70	0.24	0.35	182
Other financial service	0.83	0.62	0.71	215
Prepaid Card	0.93	0.76	0.84	532
accuracy			0.85	16702
macro avg	0.75	0.65	0.68	16702
weighted avg	0.85	0.85	0.85	16702

Table 3. Classification report of Linear SVC model

	precision	recall	f1-score	support
Debt Collection	0.80	0.78	0.79	1428
Consumer Loan	0.74	0.62	0.68	920
Mortgage	0.79	0.80	0.80	1982
Credit card	0.88	0.86	0.87	3132
Credit reporting	0.83	0.89	0.85	4388
Student Loan	0.75	0.61	0.67	166
Bank account or service	0.92	0.95	0.94	3730
Payday Loan	0.50	0.07	0.13	27
Money Transfers	0.57	0.36	0.44	182
Other financial service	0.80	0.73	0.76	215
Prepaid Card	0.92	0.81	0.86	532
accuracy			0.85	16702
macro avg	0.77	0.68	0.71	16702
weighted avg	0.84	0.85	0.84	16702

EXTREME GRADIENT BOOSTING

Extreme Gradient Boosting (xgboost) is like gradient boosting framework but more efficient. It has both linear model solver and tree learning algorithms. So, what makes it fast is its capacity to do parallel computation on a single machine. This makes xgboost at least 10 times faster than existing gradient boosting implementations. It supports various objective functions, including regression, classification and ranking [7].

The classification report of xgboost model is as shown in table 4.

Table 4. Classification report of XGBoost model

	precision	recall	f1-score	support
Debt Collection	0.82	0.79	0.80	1428
Consumer Loan	0.75	0.61	0.67	920
Mortgage	0.80	0.81	0.80	1982
Credit card	0.83	0.84	0.84	3132
Credit reporting	0.81	0.87	0.84	4388
Student Loan	0.72	0.60	0.66	166
Bank account or service	0.92	0.94	0.93	3730
Payday Loan	0.00	0.00	0.00	27
Money Transfers	0.64	0.37	0.47	182
Other financial service	0.82	0.66	0.73	215
Prepaid Card	0.86	0.82	0.84	532
accuracy			0.84	16702
macro avg	0.73	0.66	0.69	16702
weighted avg	0.83	0.84	0.83	16702

DEEP LEARNING MODELS

Deep Learning has already proven successful in text classification tasks, outperforming the benchmark Machine Learning techniques [9]. The most distinctive features are evaluated automatically during the model training process. To further improve the classification performance, the pre-trained embeddings are commonly incorporated into the mode Deep Learning has already proven successful in text classification tasks, outperforming the benchmark Machine Learning techniques [9]. The most distinctive features are evaluated automatically during the model training process. To further improve the classification performance, the pre-trained embeddings are commonly incorporated into the model. The concept of embeddings assumes terms semantic relationship, i.e. the pair 'assault' and 'abuse' will display closer distance in the vector space than the pair 'love' and 'abuse'. Still, the effectiveness of embeddings in classification tasks depend on the volume, quality, and the relevance to the domain knowledge of data used for their training. Thus, the domain-specific embeddings generation is getting increasing amount of attention among the researchers Deep Learning has already proven successful in text classification tasks, outperforming the benchmark Machine Learning techniques [9]. The most distinctive features are evaluated automatically during the model training process. To further improve the classification performance, the pretrained embeddings are commonly incorporated into the model [10].

In this study, two main variations of deep learning frameworks have been covered for the text classification problem.

- CNN with Pre-trained word embeddings (GloVe)
- RNN with Bidirectional LSTM

Convolutional Neural Network

Convolution is a mathematical combination of two relationships to produce a third relationship. In Convolutional neural networks, convolutions over the input layer are used to compute the output. This results in local connections, where each region of the input is connected to a neuron in the output. Each layer applies different filters and combines their results. The modus operandi for text classification involves the use of a word embedding for representing words and a Convolutional Neural Network (CNN) for learning how to discriminate documents on classification problems.

As told by Yoav Goldberg [8], *"The non-linearity of the network, as well as the ability to easily integrate pre-trained word embeddings, often lead to superior classification accuracy."*

Unfortunately, a downside to CNN-based models – even simple ones – is that they require practitioners to specify the exact model architecture to be used and to set the accompanying hyperparameters. To the uninitiated, making such decisions can seem like something of a black art because there are many free parameters in the model [9].

This model uses pre-trained embeddings such as Glove which provides word-based vector representation trained on a large corpus. It is trained on a dataset of one billion tokens (words) with a vocabulary of 400 thousand words. The glove has embedding vector sizes, including 50,

100, 200 and 300 dimensions. In next step this embedding matrix is loaded into an Embedding layer using Sequential API to form a Convolutional Neural Network model. The first layer embeds words into low-dimensional vectors. The next layer performs convolutions over the embedded word vectors using multiple filter sizes. For example, sliding over 3, 4 or 5 words at a time. Next, we max-pool the result of the convolutional layer into a long feature vector, add dropout regularization, and classify the result using a softmax layer. Dropout is applied between the hidden layers to factor regularization and prevent overfitting of neural network.

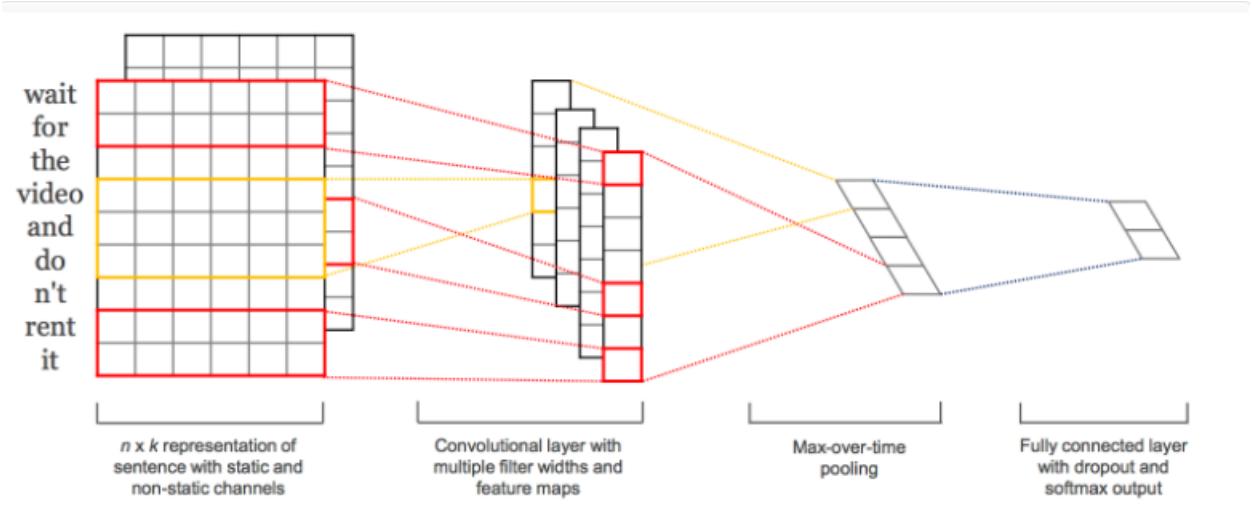

Fig 7. An example of a CNN Filter and Polling Architecture [10]

The classification report of CNN model is shown in table 5.

Table 5. Classification report of CNN model

	precision	recall	f1-score	support
Debt Collection	0.89	0.74	0.81	1428
Consumer Loan	0.82	0.60	0.69	920
Mortgage	0.85	0.79	0.82	1982
Credit card	0.87	0.88	0.87	3132
Credit reporting	0.83	0.89	0.86	4388
Student Loan	0.76	0.54	0.63	166
Bank account or service	0.95	0.94	0.94	3730
Payday Loan	0.00	0.00	0.00	27
Money Transfers	0.58	0.19	0.29	182
Other financial service	0.81	0.65	0.72	215
Prepaid Card	0.95	0.78	0.85	532
accuracy			0.85	16702
macro avg	0.75	0.64	0.68	16702
weighted avg	0.87	0.84	0.85	16702

The accuracy curve of CNN model is as represented below

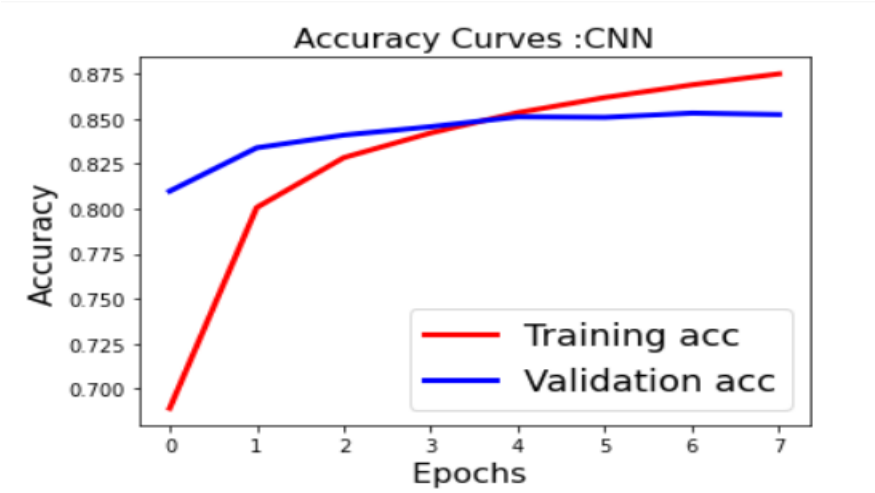

Fig 8. Accuracy curve of CNN model

Recurrent Neural Network – LSTM

Speech is a complex time-varying signal with complex correlations at a range of different timescales. Recurrent neural networks (RNNs) contain cyclic connections that make them a more powerful tool to model such sequence data than feedforward neural networks. RNNs have demonstrated great success in sequence labeling and prediction tasks such as handwriting recognition and language modeling [11]. Unlike Feed-forward neural networks in which activation outputs are propagated only in one direction, the activation outputs from neurons propagate in both directions (from inputs to outputs and from outputs to inputs) in Recurrent Neural Networks. This creates loops in the neural network architecture which acts as a 'memory state' of the neurons.

However, training conventional RNNs with the gradient-based backpropagation through time (BPTT) technique is difficult due to the vanishing gradient and exploding gradient problems [12]. An alternative solution called Long Short-Term Memory (LSTM) was proposed in [14]: The network architecture is modified such that the vanishing gradient problem is explicitly avoided, whereas the training algorithm is left unchanged [13]. Long Short-Term Memory (LSTM) is a specific recurrent neural network (RNN) architecture that was designed to model temporal sequences and their long-range dependencies more accurately than conventional RNNs [11].

Bidirectional recurrent neural networks (RNN) are just putting two independent RNNs together. This structure allows the networks to have both backward and forward information about the sequence at every time step. Bidirectional LSTMs train two instead of one LSTMs on the input sequence. The first on the input sequence as-is and the second on a reversed copy of the input sequence. This can provide additional context to the network and result in faster and even fuller learning on the problem.

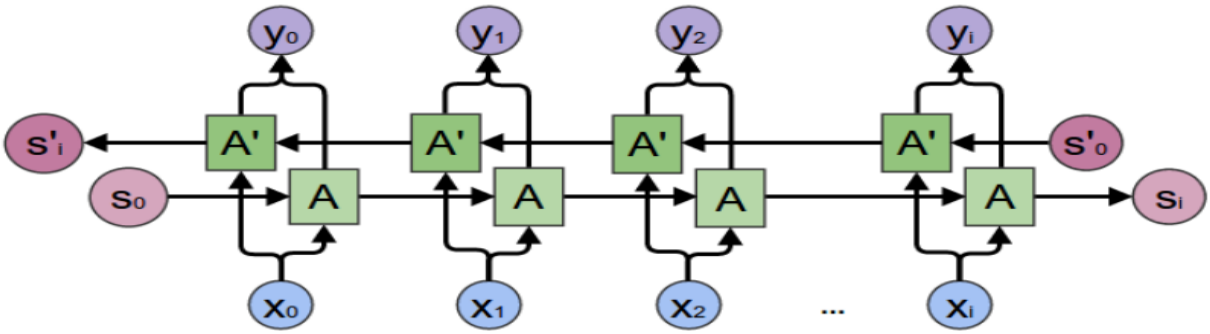

Fig 9. General structure of Bidirectional RNNs [17]

The classification report of RNN-Bidirectional LSTM model is shown in table 6.

Table 6. Classification report of RNN model

	precision	recall	f1-score	support
Debt Collection	0.89	0.75	0.82	1428
Consumer Loan	0.76	0.68	0.72	920
Mortgage	0.83	0.80	0.82	1982
Credit card	0.90	0.87	0.88	3132
Credit reporting	0.88	0.86	0.87	4388
Student Loan	0.80	0.62	0.70	166
Bank account or service	0.95	0.94	0.95	3730
Payday Loan	0.00	0.00	0.00	27
Money Transfers	0.57	0.37	0.45	182
Other financial service	0.82	0.76	0.79	215
Prepaid Card	0.94	0.84	0.89	532
accuracy			0.86	16702
macro avg	0.76	0.68	0.72	16702
weighted avg	0.88	0.84	0.86	16702

The accuracy curve of RNN-Bidirectional LSTM model is as represented below

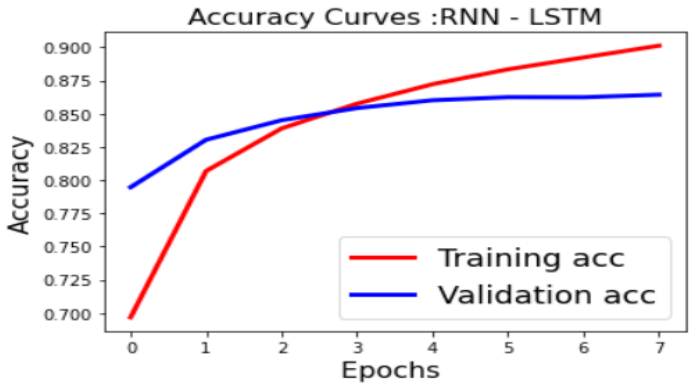

Fig 10. Accuracy curve of RNN model

CONCLUSION

Text classification is one of the most common natural language processing tasks. In this study we performed different types of feature engineering like Count Vector/TF-IDF/ Word Embedding on text dataset and we applied several models for text classification from logistic regression to increasingly more advanced methods leading to LSTM- recurrent neural networks. The performance of different models is shown in Table 7.

Table 7. Performance comparison of different models

Models	Accuracy
Logistic Regression	85%
Naïve Bayes	69%
Support Vector Machine	85%
Random Forest	78%
XGBoost	84%
CNN	85%
RNN	86%

Based on the results it can be concluded that RNN model shows better performance than traditional models for this dataset. This can be improved by hyperparameter tuning such as no of epochs and dropout for regularization. In case we have less variables than observations we can perfectly use logistic regression model for text classification. RNN based model is more useful in case we have more complex data and more variables for text classification.

The benefit of this analysis of the CFPB data is that each of the complaint can be tagged to a category automatically, hence areas of greatest concern for consumer can be identified and measures can be taken. With this information, businesses can uncover trends surrounding the actions for a category and improve it in future.

REFERENCES

1. M. Taleghani, M. S.Largani and S. Gilaninia, S. J. Mousavian, The role of customer complaints management in consumers satisfaction for new industrial enterprises of Iran, International Journal of Business Administration, vol. 2, no. 3, pp. 140-147, August 2011.
2. (Ayres, I., Lingwall, J., & Steinway, S. 2016. Skeletons in the Database: An Early Analysis of the CFPB's Consumer Complaints
3. Guide to Text Classification with Machine Learning Link: https://monkeylearn.com/text-classification/
4. Lemmatization Approaches with Examples in Python by Selva Prabhakaran Link: https://www.machinelearningplus.com/nlp/lemmatization-examples-python/
5. Scikit Learn official Label Encoder code Link: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html
6. Text Classification Algorithms: A Survey by Kamran Kowsari Link: https://medium.com/text-classification-algorithms/text-classification-algorithms-a-survey-a215b7ab7e2d
7. How to use XGBoost algorithm in R in easy steps by Tavish Srivastava Link: https://www.analyticsvidhya.com/blog/2016/01/xgboost-algorithm-easy-steps/
8. Yoav Goldberg (2015). "A Primer on Neural Network Models for Natural Language Processing" Link: https://arxiv.org/pdf/1510.00726.pdf
9. Ye Zhang, Byron Wallace (2015). "A Sensitivity Analysis of Convolutional Neural Networks for Sentence Classification" Link: https://arxiv.org/pdf/1510.03820.pdf
10. Yoon Kim (2014), "Convolutional Neural Networks for Sentence Classification" Link: https://arxiv.org/pdf/1408.5882.pdf
11. Hasim Sak, Andrew Senior, Francoise Beaufays, Google USA (2014). "Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling". Link: https://arxiv.org/pdf/1402.1128.pdf
12. Y. Bengio, P. Simard and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," in IEEE Transactions on Neural Networks, vol. 5, no. 2, pp. 157-166, March 1994, doi: 10.1109/72.279181.
13. Sundermeyer, Martin / Schlüter, Ralf / Ney, Hermann (2012): "LSTM neural networks for language modeling", In INTERSPEECH-2012, 194-197.
14. Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. Neural Comput. 9, 8 (November 15, 1997), 1735–1780. DOI: https://doi.org/10.1162/neco.1997.9.8.1735
15. Mike Schuster and Kuldip K. Paliwal. "Bidirectional Recurrent Neural Networks", IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 45, NO. 11, NOVEMBER 1997. Link: https://maxwell.ict.griffith.edu.au/spl/publications/papers/ieeesp97_schuster.pdf
16. Best Practices for Text Classification with Deep Learning by Jason Brownlee. Link : https://machinelearningmastery.com/best-practices-document-classification-deep-learning/#:~:text=Text%20classification%20describes%20a%20general,of%20standard%20academic%20benchmark%20problems.
17. Neural Networks, Types, and Functional Programming, Colah's Blog. Link: http://colah.github.io/posts/2015-09-NN-Types-FP/

18. Understanding Convolutional Neural Networks for NLP, NOVEMBER 7, 2015 BY DENNY BRITZ. Link: http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/

APPENDIX

1. Code Link: https://colab.research.google.com/drive/1k2x7VXNGWVqX3mvCm-8Qg3ckkrmxr-r_?usp=sharing