2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) | 978-1-7281-8595-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCCNT51525.2021.9579632

CLASSIFICATION OF FRAUD CALLS BY INTENT ANALYSIS OF CALL TRANSCRIPTS

Neha Kale*, Shivangi Kochrekar†, Rishita Mote‡ and Surekha Dholay§
Department of Computer Engineering, Sardar Patel Institute Of Technology
Mumbai, India
Email: *neha.kale@spit.ac.in, †shivangi.kochrekar@spit.ac.in,
‡rishita.mote@spit.ac.in, §surekha_dholay@spit.ac.in

***Abstract*—While the rapid growth of technology makes life easier for consumers it has also brought new threats along with it. People can be duped to reveal sensitive data such as banking and credit card details, personally identifiable information, and passwords. One such technique is phishing through phone calls. The number of people who fall for such scams is an astounding figure. The conventional approach to detecting phishing call fraud depends on a blacklist of known fraud numbers. This creates a problem when new numbers or numbers which have not been encountered by the system before are used. To solve this, we propose a system that will classify a fraudulent call by analyzing the conversation between the potential victim and the caller. We used various machine learning techniques to perform intent analysis of call transcripts. We created two models and compared them. The models based on the Naive Bayes Algorithm and CNN have an accuracy of 94.57% and 97.21% respectively.**

***Index Terms*—Classification algorithm, CNN architecture, Fraud, Machine learning, Phishing.**

I. INTRODUCTION

With the advent of new technologies, banks and other financial institutions, the government, and various other companies are shifting their base online and providing convenient and faster services to their customers. This new avenue however has brought new risks to the money and data of customers. One such threat is phishing phone calls.

Phone call frauds are a cause of severe monetary and personal information loss to a lot of people in India. The elderly and techno-phobic people who are naive and have no experience with these types of scams are more likely to fall prey to these devious crimes. In the financial year 2019-2020, more than 50,000 cases of deceitful usage of internet banking, debit cards and credit cards were reported in India as revealed by the Minister of State for Electronics and IT, Sanjay Dhotre[1]. The money involved in these fraudulent transactions is about Rs 228 crore which is around 80 crores more than the financial year 2018-2019.

Table 1. shows the data on frauds in India as reported by Standard Chartered Banks(SCBs) and First Investment Banks(FIs) in the category 'Card/Internet - ATM/Debit Cards, Credit Cards and Internet Banking'. The data is for the last three Financial Years and the period up to the quarter which ended in December 2019 of the current Financial Year based on Reporting (Rs. in cr.) :

2016-2017		2017-18		2018-19		2019-20	
No. of frauds	Amt. Involved	No. of frauds	Amt. Involved	No. of frauds	Amt. Involved	No. of frauds	Amt. Involved
1372	42.29	3471	168.99	52304	142.92	52006	228.44

TABLE I
STATISTICS OF FRAUDS FOR THE LAST 3 FINANCIAL YEARS

According to the information reported to and tracked by the Indian Computer Emergency Response Team (CERT-In) a total of 454, 472 and 194 phishing incidents were noticed during the year 2018, 2019 and 2020 (till August) respectively. Further, a total of 6, 4 and 2 financial fraud incidents involving ATMs, Cards, Point-of-Sale (PoS) systems and Unified Payment Interface (UPI) have been reported during the year 2018, 2019 and 2020 (till August) respectively [2].

The conventional approach to detecting phishing call fraud depends on a blacklist of known fraud numbers. However, fraudsters can simply change their number to evade detection. To solve this problem, we propose a system that will classify a fraudulent call by analyzing the conversation between the potential victim and the caller.

In particular, we used data collected through various sources having reports or testimonies of the victims of such crimes. In addition, we passed a survey to collect data about frauds calls. The survey collected experiences of the people who were a victim to fraud calls. We used conversational data sets and then combined the fraud and not fraud data sets. Data analysis and data visualization was then performed to better understand the data we had. The data set was found to be highly skewed since the number of non-fraud calls was much greater than the number of fraud calls. Then we cleaned the data and used various pre-processing techniques on it. After splitting the data set into training and testing data sets, we performed oversampling on the training data set to make the minority and the majority class equal. We created two models based on two different approaches: Naive Bayes and CNN models. Then we compared both these models and evaluated them. Our results are positive and the project would be beneficial to further researchers.

II. Literature Survey

Research has been conducted on email phishing, which is a traditional way of attack. The work of Şentürk et al. [3] is based on machine learning and data mining techniques to detect phishing emails. Phishing through phone calls wherein the fraudster manipulates the victim to divulge sensitive information during a phone call is a newer form of fraud.

Jabbar and Suharjito [4] employed unsupervised machine learning techniques which used Call Detail Records(CDR) to detect fraud calls. The variables used are number dialed, destination city, duration, caller number and fee of the data set which are similar to the traditional variables used in the detection of email phishing.

Maseno [5] proposed a theoretical model that can be used to detect such attacks. The model aims to help the user to effectively and quickly identify if the caller is trying to divulge information from them. The study conducted cross-sectional survey research on 20 respondents who were selected using random sampling. Data was collected through a structured questionnaire for mobile phone users. An interview guide was used for the key informants in Kenya. Content analysis was used to analyze qualitative data while quantitative data was analyzed using SPSS. The findings revealed that the major contributing factors in such attacks are information sensitivity, psychological factors, and technical factors. A model was developed to aid users in the detection of phishing phone call attacks based on these three main factors.

While the previous study provided a theoretical model, Zhao et al. [6] have created an Android application to detect telecommunication frauds. When a call is answered, the application can actively analyze the contents of the call so that frauds can be identified. In particular, they collected descriptions of such scams from news reports and social media. Then they used machine learning algorithms to scan this data and to choose high-quality descriptions to form data sets. After this, they leveraged natural language processing to perform feature extraction from the textual data. Then they created rules for a model that identifies similar content for further fraud detection. The content used for matching was dependent on news reports from social media and not from actual calls.

Hollmén et al. [7] used a hierarchical regime-switching model to detect call-based frauds. This research was performed in 1998 and the model is outdated now.

Kedem et al. [8] have targeted vishing attacks using a new approach wherein the attacker provides step-by-step instructions to the victim over the phone which tell the victim to log in to his account and perform a banking transaction. Their proposed system monitors the gestures performed via input units, transactions, timing and speed of data entry, online operations, user engagement and user interactions with user interface elements. It ascertains that the victim is operating under dictated instructions by detecting the data entry rhythm and many other typical behaviors exhibited while performing an online banking transaction while also speaking on the phone.

Peng and Lin [9] found that most existing fraud detection models mark the phone numbers of fraudsters and warn the users according to the marked results. They proposed a scam call analysis method that is based on the label propagation community detection algorithm (LPA). The call content is converted into a complex network. Then the LPA algorithm is used to create fraud communities on that complex network.

Marzuoli et al. [10] performed a large-scale data-driven analysis of the telephony spam and fraud ecosystem. They uniquely identified bad actors potentially operating several phone numbers. The data set was collected from a website called ”honeypot”. It contains around 8,000,000 calls that were received in 2015. Out of these 880,000 were from distinct sources and 80,000 had distinct destinations. They collected about 40,000 such call recordings data. They then demonstrated that only a few bad actors are responsible for the majority of telephony spam and fraud and that they can be uniquely identified by their audio signature. They studied the semantic information obtained from call recordings using NLP and clustering algorithms. Then the audio features of the call were extracted from each cluster and analyzed to detect whether the call is fraudulent or not. So their work was majorly based on identifying the existing fraudsters but would be unable to work on new fraudsters.

Choi et al. [11] have assessed the modus operandi of voice phishing using crime script analysis. The results of their study showed that the preparation for this kind of voice phishing includes readying for the crime, recruiting telemarketers and creating the scripts. The next step involves randomly making international and voice-over-internet calls to a vast amount of people, which constitutes its major activity. The post-activity involves the withdrawing and wiring of the amount of money deposited by victims to the perpetrators.

Saini [12] explains how the bad guys use social engineering techniques to steal the personal and sensitive information of a user. The modus operandi of voice phishing that is used by these fraudsters nowadays is also explained. Based on the survey, some of the cases studies and examples are mentioned along with the protective measures that a user can take to protect their personal information.

Zhang et al. [13] have proposed a design (CATINA) a novel content-based approach, using the TF-IDF information retrieval algorithm. They have implemented and discussed the evaluation of several heuristics to reduce false positives. This method was however limited to websites.

Tu et al. [14] performed a study to work out the methodology, design, execution, results, analysis and evaluation for why vishing attacks work and what countermeasures to take against them. The study was performed using 10 telephone phishing call experiments on 3,000 of their university participants including staff and faculty without prior awareness. The results were analyzed by performing linear regression and statistical hypothesis testing methods through which they identified that spoofed Caller ID had a significant impact on tricking the victims into revealing their Social Security number.

Maseno et al. [15] worked on a study aimed at finding out the pivot factors of vishing attacks. Their study was cross-sectional survey research. The sample space of respondents was selected using random sampling and their data was collected using a structured questionnaire for mobile phone users. The study revealed that the pivot factors for such attacks are psychological, technical and information sensitivity based. Based on these factors, mitigation measures were proposed.

Aleroud and Zhou [16] created a multidimensional phishing taxonomy based on a comprehensive survey of the related literature. The taxonomy provides an integrated view of phishing that consists of four dimensions: communication media, target environments, attacking techniques, and countermeasures. Their phishing countermeasures provide a classification consisting of five categories: Human Users, Profile Matching, Machine Learning, Text Mining, and the last category consists of ontology, search engines, honeypot countermeasures and client-server authentication.

Alabdan [17], performed a comprehensive analysis on the characteristics of the existing classic and modern phishing attack techniques. It explains the various characteristics of the different approaches and types of phishing techniques, may serve as a base for developing a more holistic anti-phishing system.

We observed that most of the current work is focused on theoretical models or fraud mitigation techniques. The few automated systems developed rely on call-related features like blacklisted phone numbers, the fraudster's location, caller id, and voice to detect the fraud calls. The system we proposed will use the content of the call rather than other external features.

III. Proposed Methodology

In this section, we describe our system, its key features and the procedure we employed to create the system. From our study of the existing technologies and related work, we identified some limitations as mentioned in the Literature Survey. Then we came up with a feasible solution that is better equipped at solving this problem. Our system will classify a fraud call based on the transcript of the call conversation. We strive to analyze the intent of the caller based on the content of the calls. Our algorithm is built using call transcripts in the English language.

The steps to create the model are shown diagrammatically in Fig. 1 and are described as follows:

A. Assembling Data

We searched for phone call transcripts data sets. However such data sets are not abundantly available to the public due to privacy concerns. So we also searched for conversational data sets and found various data sets which with a little processing could suit our needs. These would be the part of the data set which are not fraud calls. For the fraud phone calls, we passed a survey amongst people and based on the responses created the fraud calls transcripts part of our data set. We also used data collected through various sources having reports or testimonies of the victims of such crimes. Then we merged these two parts and formatted it to create our data set. The data set has a total of 2775 transcripts.

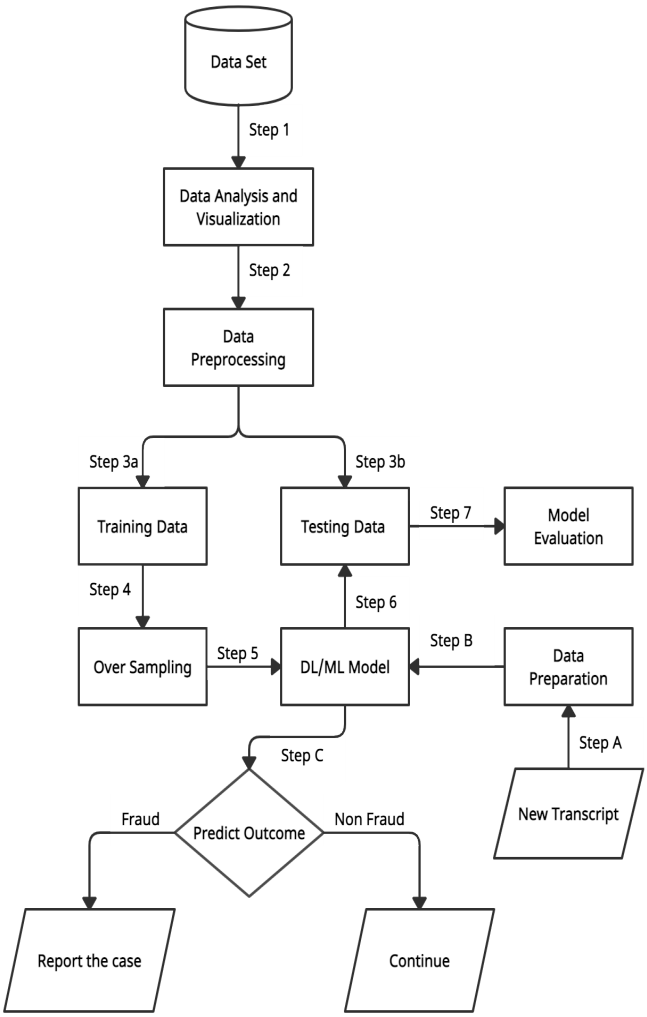

Fig. 1. Flow of Methodology

B. Data Analysis and Visualization

We plotted the data as shown in Fig. 2 and Fig. 3 and performed data analysis. The data was found to be skewed. This is because fraud calls form a very small percentage of the total calls received by someone. When a data set does not represent all classes of data equally, the model might overfit to the class that's represented more in your data set. It might become oblivious to the minority class. Thus it might even give a good accuracy but might fail miserably in real life. In our project, a model that keeps predicting that call is not a fraud call every single time will also have a good accuracy as the occurrence of fraud call itself will be rare among the inputs. But it will fail when an actual case of fraud is subjected to classification, therefore failing its original purpose. Hence, we had to balance the data set.

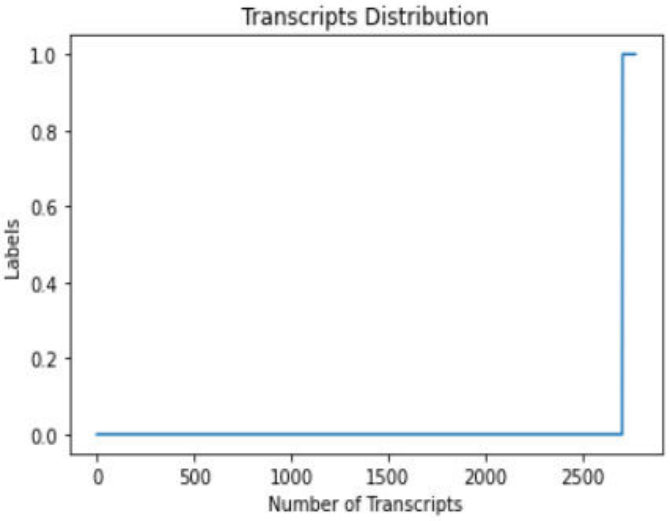

Fig. 2. Line Plot depicting the data set

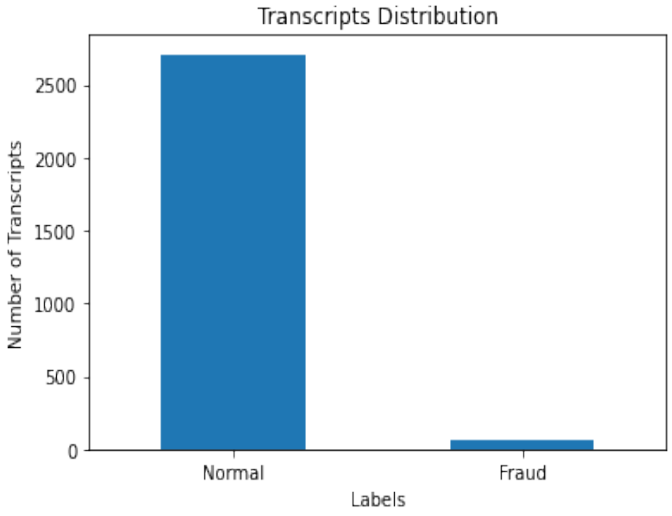

Fig. 3. Box Plot showing the skewness of the data set

C. Preprocessing Data

Text preprocessing means clearing the text to make it understandable to the computer. Our application removes the information from the call transcripts that is not useful for the model. This is achieved in the following ways:

- **Removing Punctuations and Stopwords:** We first converted all the words to lowercase letters. Then we removed the punctuation. Stopwords are the words (data) that are not useful for the model, e.g., 'for', 'the', 'as', etc. So using the NLTK library in python we filtered out all the stopwords.
- **Digits to words:** The call transcripts consists of digit, e.g., Rs 25,000. These digits are converted into words which can then be converted easily into word vectors for the algorithm.
- **Stemming:** Stemming in Natural Language Processing means removing the affixes of the word and representing it by its stem word. The stem of the words 'sleeping', 'sleeps', 'sleeper' is sleep. This step helped in normalizing the corpus.
- **Lemmatization:** Lemmatization is the same as stemming but instead of stem word, the word is replaced by its root word. Stemming sometimes creates new words that may not have any meaning. So to resolve the problem we performed stemming first followed by lemmatization.

D. Labelling and splitting the data

After preprocessing the data, we labeled the data with unique values 1/positive and 0/negative. 1 represents that the call transcript is a phishing call i.e positive and 0 represents that it is a normal call transcript. Then we split the data set into training and testing data sets.

E. Over Sampling

Johnson and Khoshgoftaar [18] carried out a survey that provides the most comprehensive analysis of deep learning methods for addressing the class imbalance data problem. A widely used technique for dealing with unbalanced data sets is called resampling. It is done after the data is split into training and test sets. Resampling is done only on the training set otherwise the performance measures could get skewed. Resampling can be of two types: Over-sampling and Under-sampling. We used oversampling to balance our data set. Oversampling in simplified terms is duplicating random records from the minority class to make it equal to the majority class.

F. Built and trained the models

We created a Naive Bayes and a CNN model. Then we trained these models on the training data set.

1) CNN Model: A convolutional neural network, which is a class of deep neural networks, is a Deep Learning Algorithm that is mainly used in image classification [19]. However, it has its uses in text classification and sentiment analysis too. Kumar and Zymbler [20] used CNN to analyze customer satisfaction from airline tweets. A CNN has hidden layers which are known as convolutional layers that are added one after another. These layers consist of multiple filters that help in detecting specific features. CNN for text classification mainly uses three such layers that are: embedding layer, convolutional one-dimensional layer and global max-pooling layer. Each of these layers and the steps involved in the CNN algorithm are explained in the following sections:

- **Embedding Layer:** Machine learning algorithms require numeric data. There are various encoding techniques such as Bag Of Words(BOW), TFIDF, Word2Vec that encodes the given corpus in a numeric form. This process of converting each word into its vector is "Word Embedding". The advantage of word embedding is that it collects more information in fewer dimensions. It maps the semantic meaning of the word in a geometric space called an embedding space. Our application uses one

of the most efficient techniques for word embedding "Word2Vec". Word2vec is developed by Google which has pre-trained word embeddings. To train our word embeddings, we used the Gensim Python package which uses Word2vec calculations. Gensim expects the input of sentences sequentially. It trains the word and stores it in the KeyedVector instance. Gensim has several pre-trained models. Once the word vectors are trained they are stored in a format that is compatible with word2Vec implementation.

- **Convolutional Layer:** Before implementing the CNN model, we first added padding to the sequence to make each sentence of the same length. This is achieved by finding the length of the longest sequence. After padding the sequences, we implemented the Convolutional 1-D layer using the Keras library in Python. This layer is in between the Embedding layer and GlobalMaxPooling1D layer. This layer has several parameters and the important ones are Kernel size, filter size and activation type. Typically, in word embedding, each sentence is represented in a matrix form. The rows of the matrix represent the tokens in the sentence and the columns represent vectorize words. This matrix is convolved with different filter sizes in the Convolutional 1-D layer.
  We used the filter sizes of [2,3,4,5,6]. The kernel size in CNN represents the sequence of words it will convolve at a given time. So, during the convolution process the sequence of words according to the kernel size are taken into consideration and are multiplied by the filter size. These multiplication results are then summed together and then feed to an activation function. The activation function that we used is the Rectified Linear Unit(relu). This function gives the feature value and the mathematical formula used is as shown in (1):

$$c_i = f(w * x_{i:i+m-1} + b) \tag{1}$$

  Here, c = convolutional process, w = word matrix, x = element wise multiplication operation, b = bias term b from that row. Once the convolution process is completed for one filter, all the features obtained by the relu function are mapped to the feature map as [c1, c2, c3...c(m-1)].
- **Global Max Pooling I-D:** We then applied the GlobalMaxPolling1D layer on the convolution layer to get the maximum value of the features in a pool for each feature dimension. When all the filters are applied to the convolutional layer, a list of feature values is made using this max-pooling feature.
  The final step in CNN is to form a full connection layer which includes the dropout and regularization from the final feature vector to the output layer. We then summarised our model on the training set by displaying the type of layer used, the Output Shape of each layer and the connection between layers.

2) Naive Bayes: The Naive Bayes classifier is a classification algorithm based on the Bayes Theorem. The main principle of this algorithm is that every pair of features that are being classified are independent of each other.

In this model, we first stored the positive and negative i.e fraud and non-fraud call transcripts present in the training data set and tokenized each word. The tokens of positive and negative classes are stored in different dictionaries. Both the dictionaries are then combined and then passed to the model.

IV. Implementation & Results

To implement the proposed methodology, the system should have a stable internet connection and the required data set. For the application to run successfully a system having a minimum RAM capacity of 4GB and a maximum of 8GB is required.

After implementing our methodology, we obtained an accuracy of 95.47% for the Naive Bayes model and 97.21% for the CNN model respectively.

As shown in the confusion matrix in Fig. 4 where the rows represent the actual labels and columns represent the predicted labels, for the Naive Bayes algorithm, most of the calls were classified correctly. Also, the number of normal calls is more than the number of fraud calls even while testing for a small subset of calls. Furthermore, none of the normal calls were classified to be fraud calls. A less number of fraud calls were classified to be normal calls.

Since the data set is highly imbalanced, we cannot rely on the model's accuracy only. So to check the model's performance we plotted the graph to display the precision, recall and F1 score of both models. We checked our results using the evaluation parameters: precision, recall and F1 scores.

Precision, also known as true positive rate, tells us the number of positive class predictions that truly belong to the positive class. From Fig. 5 and Fig. 6 we observed that for both the algorithms our precision is high which means that the model does not give out many false positives. On the other hand, recall tells us how correctly the model identifies the True Positives. The recall for the models in the case of the positive class is low which implies that there are quite a few instances of positive class i.e. a fraud call to be predicted as negative i.e. normal call.

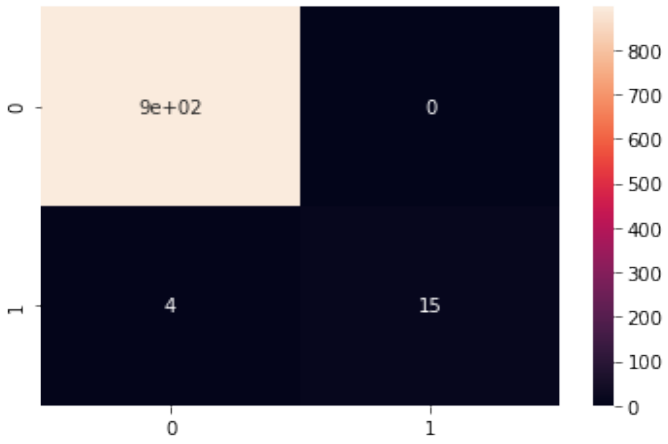

Fig. 4. Confusion Matrix of the Naive Bayes Model

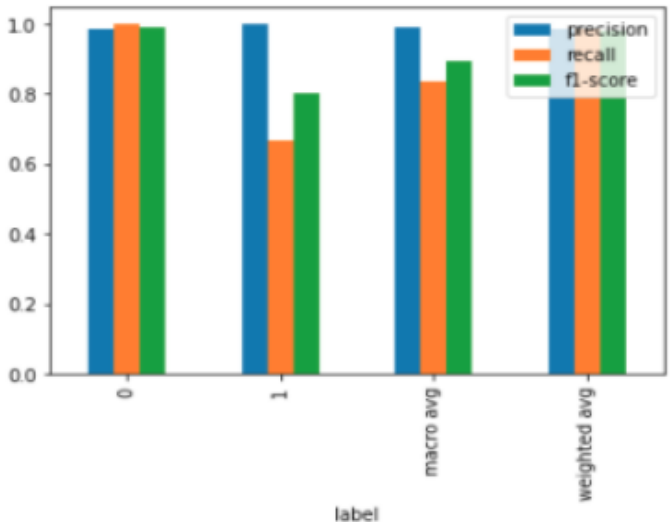

Fig. 5. Results of the CNN Model

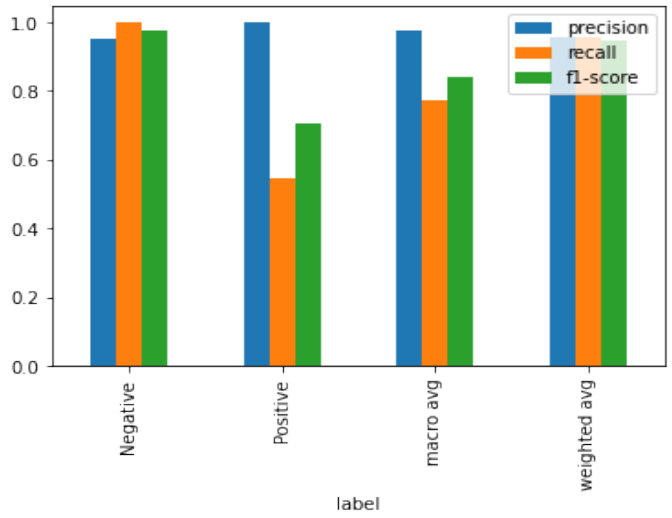

Fig. 6. Results of the Naive Bayes Model

V. Conclusion

After implementing two different algorithms we determined that the CNN model gives an accuracy of 97.21% and the Naive Bayes model gives an accuracy of 95.47%. The recall i.e. improper classification of fraud calls, an important factor for our problem statement is comparatively higher in the case of CNN than Naive Bayes. Hence we can conclude that the performance of the CNN model is better and is well equipped to classify the calls.

There are a few limitations to this project. Newer algorithms provide scope to improve the model performance. An interface is needed to implement this model. The ways and methods of duping people are always evolving and hence the data will need to be updated periodically.

Phishing through phone calls is a modern way to attack people and seek their personal information. It could be used to trick people from rural areas or elderly people since they may be unaware of such heinous frauds. Our models will help further research which will aid users to avoid such phishing call attacks. In this way, our work will help people from getting tricked through fraudulent phone calls and thus will help to reduce phishing cases.

References

[1] 'Sujay Radhakrishna Vikhepatil, Shrikant Eknath Shinde, Hemant Patil, Unmesh Bhaiyyasaheb Patil, Sambhajirao Mane Dhairyasheel', Fraudulent Usage of Credit/Debit Card, http://loksabhaph.nic.in/Questions/QResult15.aspx?qref=15384&lsno=17

[2] 'Sumedhanand Saraswati', Online Frauds and Scams,http://loksabhaph.nic.in/Questions/QResult15.aspx?qref=17288&lsno=17

[3] Ş. Şentürk, E. Yerli and İ. Soğukpınar, "Email phishing detection and prevention by using data mining techniques", 2017 International Conference on Computer Science and Engineering (UBMK), Antalya, Turkey, 2017

[4] M.A. Jabbar, Suharjito, "Fraud Detection Call Detail Record Using Machine Learning in Telecommunications Company", Advances in Science, Technology and Engineering Systems Journal, vol. 5, no. 4, pp. 63-69 (2020)

[5] Elijah M. Maseno, "Vishing Attack Detection Model For Mobile Users", KCA University, 2017

[6] Zhao, Q., Chen, K., Li, T. et al. "Detecting telecommunication fraud by understanding the contents of a call", Cybersecur 1, 8 (2018)

[7] Hollmén, Jaakko & Tresp, Volker, "Call-Based Fraud Detection in Mobile Communication Networks Using a Hierarchical Regime-Switching Model", 889-895

[8] Oren Kedem, Avi Turgeman, Itai NOVICK, Alexander Basil Zaloum, Leonid Karabchevsky, Shira Mintz, Ron Uriel Maor, "Device, System, and Method of Detecting Vishing Attacks", U. S. Patent 16/188,312, May 23, 2019

[9] L. Peng and R. Lin, "Fraud Phone Calls Analysis Based on Label Propagation Community Detection Algorithm," 2018 IEEE World Congress on Services (SERVICES), 2018

[10] A. Marzuoli, H. A. Kingravi, D. Dewey and R. Pienta, "Uncovering the Landscape of Fraud and Spam in the Telephony Channel," 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), 2016

[11] Choi, Kwan & Lee, Ju-lak & Chun, Yong-tae, "Voice phishing fraud and its modus operandi", Security Journal, 2017

[12] Ujjwal Saini, "Voice Phishing Attacks", International Research Journal of Engineering and Technology (IRJET), July 2020

[13] Yue Zhang, Jason I. Hong, and Lorrie F. Cranor, "Cantina: a content-based approach to detecting phishing web sites, In ¡i¿Proceedings of the 16th international conference on World Wide Web¡/i¿ (¡i¿WWW '07¡/i¿). Association for Computing Machinery, New York, NY, USA, 2007, 639–648

[14] Tu, H., Doupé, A., Zhao, Z., & Ahn, G. J, "Users really do answer telephone scams", In Proceedings of the 28th USENIX Security Symposium (pp. 1327-1340). (Proceedings of the 28th USENIX Security Symposium). USENIX Association, 2019

[15] Elijah M. Maseno, Patrick Ogao, Samwel Matende, "Vishing Attacks on Mobile Platform in Nairobi County Kenya", International Journal of Advanced Research in Computer Science & Technology (IJARCST 2017)

[16] Ahmed Aleroud, Lina Zhou, "Phishing environments, techniques, and countermeasures: A survey", Computers & Security, Volume 68, 2017, ISSN 0167-4048

[17] Alabdan, Rana, "Phishing Attacks Survey: Types, Vectors, and Technical Approaches, Future Internet", 12, (2020)

[18] Johnson, J.M., Khoshgoftaar, T.M. "Survey on deep learning with class imbalance", J Big Data 6, 27 (2019)

[19] Xin, M., Wang, Y. 'Research on image classification model based on deep convolution neural network', J Image Video Proc. 2019, 40 (2019)

[20] Kumar, S., Zymbler, M. "A machine learning approach to analyze customer satisfaction from airline tweets", J Big Data 6, 62 (2019)