International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 12
DOI: https://doi.org/10.17762/ijritcc.v10i12.5894
Article Received: 04 October 2022 Revised: 09 November 2022 Accepted: 12 December 2022

Development of Deep Learning based Intelligent Approach for Credit Card Fraud Detection

Dr. V. Gokula Krishnan[1], Dr. M. V. Vijaya Saradhi[2], T. A. Mohana Prakash[3], Dr. K. Gokul Kannan[4], AG. Noorul Julaiha[5]

[1]Professor, Department of CSE, Saveetha School of Engineering
Saveetha Institute of Medical and Technical Sciences
Thandalam, Chennai, Tamil Nadu, India
e-mail: gokul_kris143@yahoo.com
[2]Professor & Head, Department of CSE,
ACE Engineering College,
Ghatkesar, Hyderabad, Telangana, India
e-mail: meduri_vsd@yahoo.co.in
[3]Associate Professor, Department of CSE,
Panimalar Engineering College,
Poonamallee, Chennai, Tamil Nadu, India
e-mail: tamohanaprakash@gmail.com
[4]Assistant Professor, Department of ECE,
Loyola Institute of Technology,
Thandalam, Chennai, Tamil Nadu, India
e-mail: gokulkannanme@gmail.com
[5]Assistant Professor, Department of CSE,
Rajalakshmi Institute of Technology,
Kuthambakkam, Chennai, Tamil Nadu, India
e-mail: ag.nooruljulaiha@gmail.com

Abstract— Credit card fraud (CCF) has long been a major concern of institutions of financial groups and business partners, and it is also a global interest to researchers due to its growing popularity. In order to predict and detect the CCF, machine learning (ML) has proven to be one of the most promising techniques. But, class inequality is one of the main and recurring challenges when dealing with CCF tasks that hinder model performance. To overcome this challenges, a Deep Learning (DL) techniques are used by the researchers. In this research work, an efficient CCF detection (CCFD) system is developed by proposing a hybrid model called Convolutional Neural Network with Recurrent Neural Network (CNN-RNN). In this model, CNN acts as feature extraction for extracting the valuable information of CCF data and long-term dependency features are studied by RNN model. An imbalance problem is solved by Synthetic Minority Over Sampling Technique (SMOTE) technique. An experiment is conducted on European Dataset to validate the performance of CNN-RNN model with existing CNN and RNN model in terms of major parameters. The results proved that CNN-RNN model achieved 95.83% of precision, where CNN achieved 93.63% of precision and RNN achieved 88.50% of precision.

Keywords- Synthetic Minority Over Sampling Technique; Credit Card Fraud; Convolutional Neural Network; Machine Learning; Class Inequality.

I. INTRODUCTION

Over the past few years, cashless transactions by using cards have been made due to the growth of the e-commerce sector in the financial industry. However, the frauds in credit card and attacks of cyber-crime have been happened nowadays around many industries. CCF occurs when a person uses a credit card to make unauthorized purchases without the credit card holder's permission [1]. CCFD systems are important to researchers and financial institutions, as CCF causes billions of dollars in losses worldwide. Although fraud detection methods have been around for a long time, the ever-evolving supply due to new attacks and seasons remains a challenging and changing problem for researchers [2]. Due to the predominance of the effectiveness of ML techniques, stratification inequality (a rare proportion of fraudulent transactions in the data) presents a significant fraud detection challenge. Various techniques have been proposed to address the challenge of class inequality in fraud detection tasks [3]. CCF can be divided into two types: 1) "offline fraud" perpetrated by a personal card stolen elsewhere, such as a call center, and 2) "online fraud" when the owner's phone, shopping, internet, or card is not available.

CCFD is based on an analysis of the cardholder's cost behavior. Most data processing technologies are used to detect

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 12
DOI: https://doi.org/10.17762/ijritcc.v10i12.5894
***Article Received**: 04 October 2022 **Revised**: 09 November 2022 **Accepted**: 12 December 2022*

CCF and security machines [4-6]. Several researchers have used artificial neural networks and genetic algorithms [7-12], as well as CCFD using logistic regression (LR) and CCFD using K-neighbors [13]. Hybrid approaches for CCFD is carried out by using algorithms, random forests as RF, neural networks, decision trees as DT, KNN models, and support vector machines as SVM [14-15]. The use of CCFD [16] is based on Baking Group Classifier, Hidden Markov Model (HMM) is used in ref. [17, 18], Migratory Bird Update Algorithm in Reference [19], and smart self-regulatory map (SSM) [20-21]. This paper [22] evaluates four techniques in an effort to detect CCF, including NB (Naïve Bayes), SVM, KNN and RF, that explore the effectiveness of these techniques. However, we encountered a number of challenges in this study, including the fact that 'fraudulent' behavior appears to be 'real' and that real database transactions are not publicly available data. Therefore, feature selection in this study is problematic. Diagnosis is very difficult and complex.

The main objective of the research activity is to solve the imbalance dataset by SMOTE technique, where features are extracted by CNN model and RNN model is used for final prediction of CCF. In order to get better accuracy, the data preparation is majorly focused before feature extraction process. The experiments are conducted on European dataset in terms of various parameters to test the efficiency of proposed CNN-RNN model with existing DL techniques. The remaining draft is structured as: Section 2 has existing techniques' survey, Section 3 presents the brief discussion of CNN-RNN model for prediction of CCF, where Section 4 contains the experimental study and finally, the contribution is given in Section 5.

II. LITERATURE REVIEW

The authors [23] proposed six well-known data processing technologies, namely DT, RF, BN, NB, KNN and SVM, and used their specific model to detect CCF. They have integrated a set of Artificial Intelligence (AI) models used in real transactions from one of the leading banks in Turkey. Results based on performance measurements: DT results accuracy of 95.19%, sensitivity of 52.53% and 97.35% of specification. From the experimental analysis, it is clearly shows that DT achieved better performance than other ML classifiers. However, imbalance problem is not solved by this technique.

Three tracking methods have been used to predict paper CCF, [24] which are LR, gradient Bayesian trees (GBT), and DL. In comparison to the above three techniques, the authors explore achievements in terms of features, using domain expertise and feature engineering. They decided that it would be best to use domain expertise for feature engineering, and after using cross-validation 5 times their results were: LR (83.8), GBT (87, 4), DL (86, 2).

The study [25] shows a method for extracting the right attributes from transactions to develop an approach to detecting CCF, thus extending the transaction integration strategy by proposing the creation of a new pool of assets.

A comparative analysis of algorithms is often used to detect CCF in the workplace reference. [26] These include LR, DT, and random forest. A database of generally available German credit data is provided for evaluation among algorithms. The results from the analysis show that RF are preferable. Still, ref. [27] focused only on two different types of random forest, viz. Tree-based random forest classification and regression (CART). Using forest-based models in the database collected from China, CART has the highest proportion of tree-based algorithms.

Many existing techniques focused only on detecting the CCF and improves its algorithm efficiency by using different datasets. However, the problem of imbalance data is not considered by even DL and ML algorithms. Moreover, the features plays a major role for prediction process of CCF, which is also not considered by existing techniques.

III. PROPOSED METHODOLOGY

A. Data Collection

The unbalanced database for this work was collected in September 2013 from European card holders. However, the dataset is publicly available only in 2016 [28], because the real credit card transactions with its details are presented in this dataset. In this data, 31 variables are presented, where Amount and Time features are marked with unknown description, since it is related with the personal information of card users. During the collection of data, these are not real-time variables obtained, where principal component analysis is used to transform and protect the real information from the examiners, who handles the data. The real data values are presented in V1 - V28, which are principal components, where time and class are integers and 28 variables with amount are defined as numerical. The time between each transaction is defined by the attribute "time", where total spent money is defined by attribute "amount". The database provides transactions for 284,807 days, of which the feature (category) used for binary classification, the value representing fraudulent transactions is 1 and the value of the actual transaction is 0. These (492) transactions contain fraudulent transactions which account for approximately (0,172%) of the total transactions.

These small positive trades (scams) are highly unbalanced. In this case, we need to model the curved class with the existing methods. In our study, we proposed the use of an unbalanced class under the model. Under the model, a technology commonly used to reduce distortion in the stratified distributions of asymmetric data sets. The sub-model was used to approximate observational (true) values from the majority class until the database reached, where the minority (fraudulent) class was very small compared to the majority class and is equivalent to the fraudulent true class. Equilibrium (1:1), the minimal model will be useful for managing asymmetric data sets [29].

B. Preparing the CCF Data

Training and testing process of data must be done to achieve high performance on DL algorithms, for this task, data must be well-prepared before processing it directly to the classifier. While generating the data, missing and duplicate values may be presents, unwanted features will be presents and even more data anomalies may cause damage to the classifier for final prediction process. A poor sampling results and high computation cost can be achieved, when low-quality data product is used. Because of all these factors, data preparation is a very difficult and time-consuming step in data processing. No missing values and duplicates data are presents in the considered datasets. Hence, feature management is alone required that is described as follows:

B. 1. Measuring Features in the Data

When comparing with time and amount features, all other features have significant difference. Hence, “RobustScalar” method is used to avoid the negative impact of this measurement gap between features on our sample performance by measuring the time and size of features.

C. Selecting the Features

The performance of the model is enhanced by one of the most used techniques called feature selection, which is carried out once scanning and analyzing associations between features are done. This technique is used to get rid of unwanted variables, which leads to a reduction in feature space, which may improve the overall performance of the model [30]. In our research, we used standardized murals with ANOVA F-values. The specific features depend on the different statistical test scores. Here, we can identify the best K features by setting the filter threshold [31].

D. Problem of Imbalancing Data

In determining CCFD, inequality database classification is a major problem. Compared to one category variable, another category variable, offers higher training programs called unbalanced data. There are two groups on this topic, one is the minority group and the other is the majority group. In the CCF determination process, different rating models are in the majority to achieve higher results, and less emphasis is placed on the minority group which leads to poor performance in the rating model.

In this work, SMOTE technology is used to overcome the problem of class inequality and to perform more accurate tests. Instead of amplifying data by replacing/copying, SMOTE technology creates unconstrained examples. In addition, the learning process of the CCF detection algorithm has been improved using SMOTE technology. This technique can be developed based on Kumari and Mishra [32], which are described below:

Initial Step: Set A as minority class, for each x ∈ A, here, the distance between x and every other sample are calculated by using the Euclidean distance to achieve the k-nearest neighbors of x as C in set A. Set the sampling rate as N that is related to the imbalanced amount. The samples are chosen randomly from the achieved C for x ∈ A, N and finally build a new set as A1.

Final Step: A new example is produced by using the following expression for the every sample data as $xk \in A1(k = 1, 2, 3 \ldots N)$.

$$x = x + rand\ (0, 1) * |x - xk| \qquad (1)$$

Where, random number in the range of 0’s and 1’s is denoted as $rand\ (0, 1)$.

E. Classification

E. 1. Convolutional Neural Network

It involves multiplexing the output metrics to incorporate a CNN into an additional training process, which is known as confusion. Training of CCF data is done by selecting a size of kernel and multiple filters, because it can be multidimensional. The unified CNN (Conv1D) is commonly used for the prediction process of text data. The representation of word vectors is done by managing the uniform arrays in 1D-Conv. In CNN, the standard window reactivates the filter with the tutorial data, which multiplies the input by the filter weight at each step and returns the output stored in the output queue. This output sequence is the output filter for the feature or data map. This method finds a feature of the tutorial data entry. This process can be seen in Figure 1.

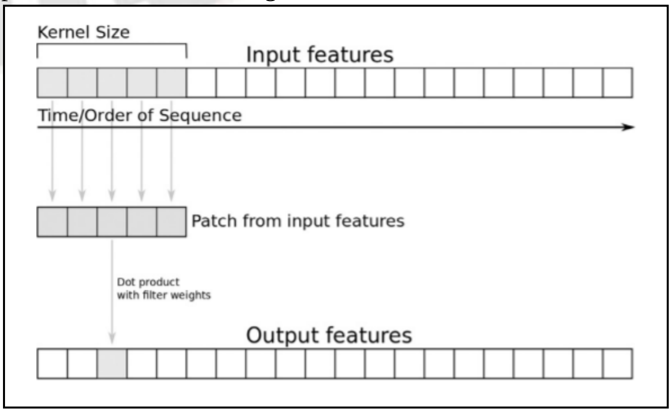

Figure 1. Architecture of Operational 1-D CNN Model

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 12
DOI: https://doi.org/10.17762/ijritcc.v10i12.5894
***Article Received**: 04 October 2022 **Revised**: 09 November 2022 **Accepted**: 12 December 2022*

The number of used feature maps is denoted by total number of filters, where kernel size describes the size of those filters. From this training data, local features are directly learned by using this CNN.

E. 2. Recurrent Neural Network (RNN)

It is defined as continuous data processing for continuous learning of a neural network (RNN). The ability to hold the memory that existed before the current sequence was executed justifies this sequence operation, which is known as iteration because each time features output is used for the next input. This allows us to know the long-term dependence on the training data.

For CCFD, instead of studying each data individually, several CCF's data can be considered for interrelated study. In this algorithm, various memory cells are used, because it is made up of layers of memory cells. One of the finest cell units is the long-short term memory unit (LSTM) unit or cell [33]. Since the sequence is executed at the cell level and at each time level during operation, current word vector is added with the vectors of the LSTM. During the sequential process, gate of LSTM is responsible for ensuring that no data loss occurs. Figure 2 shows the basic architecture of this LSTM model in RNN.

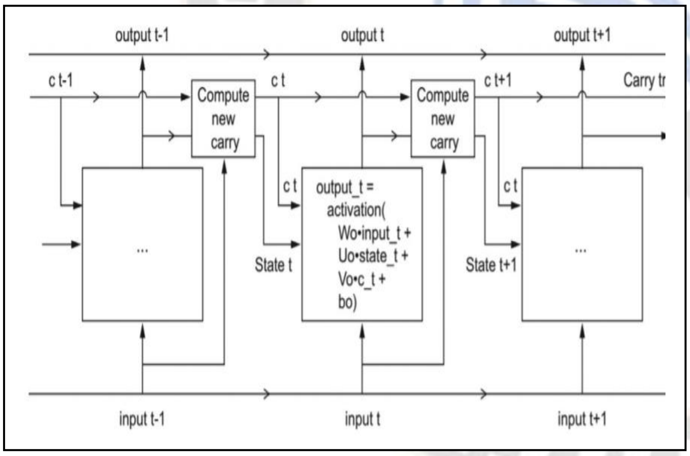

Figure 2. The Cell Structure of LSTM in RNN

Three different gates are used in LSTM cell such as weigh-and-study gates. Each time, input gate is used for collecting the data of CCF as input, where the predicted values is given by output gates and irrelevant information are removed by using forgetting gate.

E. 3. Hybrid CNN-RNN Model

The extraction of local features is carried out by CNN and long-term dependencies are studied by LSTM model in this proposed work. First, the input vector is processed by 1D-CNN layer and extracts local features in the text case. The LSTM modules of RNN obtained the input as feature maps, which is the output of the CNN layer. The CNN extracts the local features of CCF data, which is used by the RNN layer and also studies the long-term dependence of local characteristics of CCF data. The proposed model is illustrated in Figure 3.

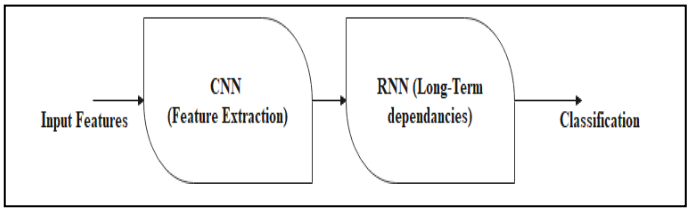

Figure 3. The Proposed Hybrid Model

In many classification and regression tasks, CNN-RNN integration has been successful, because of its ability to capture both local and sequential properties of the input data. For example, they use their ability to detect emotions [34] and sign language [35] from video streams, as well as visual features using CNN features and serials using RNN. In the case of CCFD tasks, the RNN can learn the temporal and contextual features of CCF and capture key features that can be exploited using the ability of the CNN to manage long-term dependencies between texture organizations and spatial relationships [36].

Despite their advantages, DL models have some practical limitations, such as the difficulty of defining the highest standards for each problem, the need for large training data sets, and ambiguity that directly affects their performance in new ones. In addition, the unknown functions make it behave like a black box oracle [37]. Recent advances in bio-drive methods enable the development of DL parameters and the basis for next-generation ML optimization algorithms. The proposed hybrid model will benefit greatly from the improvement of its higher parameters, and will be part of the next step in the field to explore different bio-stimulation strategies [38] and find those most suitable for the current task.

IV. Results and Discussion

The proposed model is implemented using Python language with the system configuration of 8 GB RAM and Intel core i7 processor. Imbalance classes are presented in the analyzed dataset, i.e. fraudulent transaction is less, while comparing with total number of normal class. A cross-validation (CV) test is applied in this model to compare different DL algorithms' performance.

For the validation process, the valuation parameter includes Recall (R), sensitivity (SE), specificity (SP), F1-Measure (FM) accuracy (AC), Recall (R), G-mean (GM) and Precision (P). The distinct presentation factors as:

$$R= tp/(tp+fn) \tag{11}$$

$$P= (tp+tn)/(tp+tn+fp+fn) \tag{12}$$

$$FM= (2.R.P)/(R+P) \tag{13}$$

$$SE= tp/(tp+fn) \tag{14}$$

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 12
DOI: https://doi.org/10.17762/ijritcc.v10i12.5894
***Article Received**: 04 October 2022 **Revised**: 09 November 2022 **Accepted**: 12 December 2022*

SP=tn/(tn+fp) (15)

AC=(tp+tn)/(tp+fp+tn+fn) (16)

tp_rate=tp/p (17)

tn_rate=tn/p (18)

GM= √ (tp_rate)*tn_rate (19)

Where tp,tn,fp, and fn denote the sum of cases such as a true positive, true negative, false negative, and false positive.

A. Proposed Model Evaluation

In this section, the validation of hybrid proposed model is tested with single RNN, single CNN and LSTM model in terms of various parameters for different four iterations. The results are tabulated in Table I.

TABLE I. COMPARATIVE ANALYSIS OF PROPOSED HYBRID MODEL IN TERMS OF VARIOUS PARAMETERS

Method	Iteration	SE	SP	P	R	GM
LSTM	5	75	77.50	81.07	75	73.80
LSTM	10	83.75	51	69.71	83.75	49.21
LSTM	15	59.33	79.83	--	59.33	60.79
LSTM	20	64.37	75	--	64.37	58.83
RNN	5	79	84.00	81.35	79	78.79
RNN	10	73.75	84.25	85.69	73.75	73.89
RNN	15	88.33	85.16	86.27	88.33	86.13
RNN	20	79.87	88.50	88.50	79.87	82.99
CNN	5	99.50	92	92.85	99.50	95.97
CNN	10	99.75	93.75	94.46	99.75	96.64
CNN	15	99.50	92.83	93.56	99.50	96.05
CNN	20	100	93	93.63	100	96.40
Hybrid CNN-RNN	5	99.50	94	94.56	99.50	96.66
Hybrid CNN-RNN	10	99.50	93.50	94.17	99.50	96.39
Hybrid CNN-RNN	**15**	**99.16**	**95.33**	**95.71**	**99.16**	**97.19**
Hybrid CNN-RNN	**20**	**99.37**	**95.50**	**95.83**	**99.37**	**97.38**

For 10th iteration, the LSTM achieved very poor performance on SP (51%) and GM (49.21%), RNN achieved 84.25% of SP and 73.89% of GM, CNN model achieved 93.75% of SP and 96.64% of GM and finally, the proposed CNN-RNN model achieved 93.50% and 96.39% of GM. While combining RNN and CNN model, their effectiveness are equally distributed and uses its structure for predicting the CCF by extracting the features and learn the long-term dependencies. When the number of iterations are increased to 20, all techniques provides good performance, for instance, LSTM achieved 64.37% of SE, 75% of SP, 64.37% of P and 58.83% of GM. RNN 79.87% of SE, 88.50% of SP, 88.50% of P, 79.87% of R and 82.99% of GM, where CNN achieved 100% of SE, 93% of SP, 93.63% of P, 100% of R and 96.40% of GM. But, the proposed hybrid model (CNN-RNN) achieved 99.37% of SE, 95.50% of SP, 95.83% of P, 99.37% of R and 97.38% of GM. From this analysis, it is clearly proves that the hybrid model achieved little bit better performance than single CNN model, due to the capturing of both sequential and local properties of input CCF data. However, in some parametric values, CNN achieved better performance than hybrid model, the reason is that CNN acts a feature extraction model in this research work. Figure 4 and 5 shows the graphical representation of existing DL with proposed model in terms of various parameters for 5th and 15th iterations.

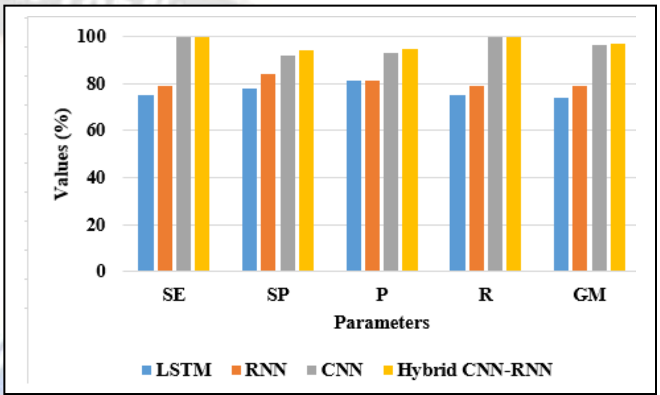

Figure 4. Graphical Representation of Proposed CNN-RNN Model for 5th Iteration using Different Parameters

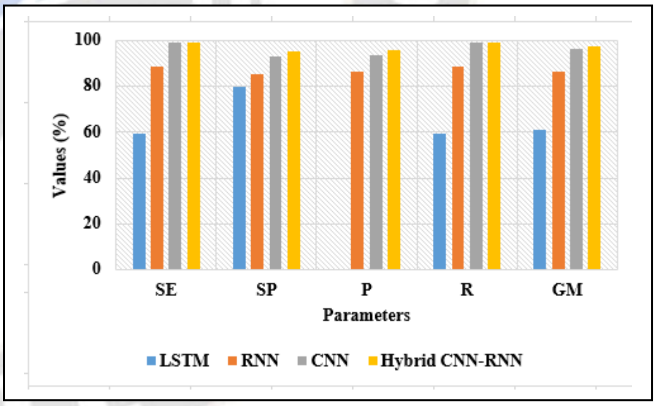

Figure 5. Graphical Representation of Proposed CNN-RNN Model for 15th Iteration using Different Parameters

Table II shows the experimental analysis of existing DL with proposed hybrid model in terms of accuracy and FM.

TABLE II. ACCURACY AND FM ANALYSIS OF THE PROPOSED SYSTEM

Method	Iteration	ACC	FM
LSTM	5	76.25	74.68
LSTM	10	67.37	70.01
LSTM	15	69.58	--
LSTM	20	69.68	--
RNN	5	81.51	77.44
RNN	10	79.00	71.88
RNN	15	86.75	86.43
RNN	20	84.18	82.19

CNN	5	95.75	83.67
CNN	10	96.75	96.93
CNN	15	96.16	96.35
CNN	20	96.50	96.66
Hybrid CNN-RNN	5	96.75	96.89
Hybrid CNN-RNN	10	96.50	96.68
Hybrid CNN-RNN	**15**	**97.25**	**97.35**
Hybrid CNN-RNN	**20**	**97.43**	**97.52**

For the analysis of 15th iterations, LSTM model achieved 69.58% of AC, RNN achieved nearly 86% of AC and FM, CNN model achieved nearly 96% of AC and FM, where the proposed CNN-RNN model achieved 97.25% of AC and 97.35% of FM. The reason for better performance than existing DL is that imbalance data are balanced by using SMOTE technique, before the classification process occurs. When the iteration is 20, LSTM model achieved 69.68% of AC, RNN achieved nearly 84.18% of AC and 82.19% FM, CNN model achieved nearly 96% of AC and 96.66% of FM, where the proposed CNN-RNN model achieved 97.43% of AC and 97.52% of FM. From this analysis, it is clearly proves that the proposed hybrid model achieved better performance than existing techniques in terms of different parameters for the given dataset to predict the CCF. Figure 6 shows the comparative graphical representation of proposed model in terms of AC and FM for 5th and 10th iterations.

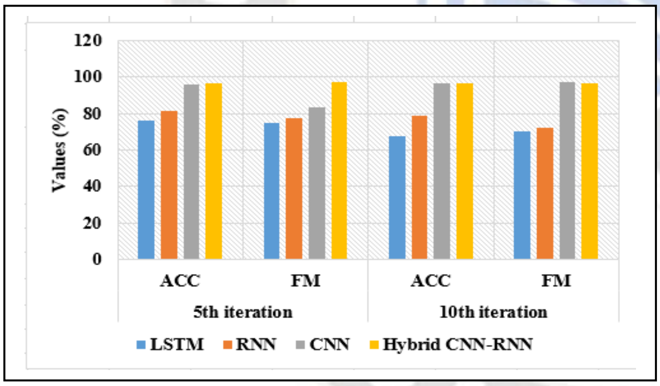

Figure 6. Comparative Graphical Representation of Proposed Model with Existing DL Classifiers in terms of Accuracy and F1-Measure

V. CONCLUSION

In this research work, an intelligent based CCFD system is developed by using DL technique, where the problem of imbalance data is solved by SMOTE technique. In the data, features are measured by using RobustScalar method for avoiding the negative impact of time and amount features. In addition, the important features are selected by using filter threshold technique and CNN model acts as a feature extraction model in this work. Finally, the RNN classifier is used for learning the long-term dependency features to predict the CCF. The experiments are carried out on European datasets to test the effectiveness of proposed CNN-RNN model with existing LSTM, RNN and CNN model. For various iterations, the experiments are conducted and results are tabulated in Table 1 and 2. The proposed CNN-RNN model achieved 96.75% of accuracy, CNN model achieved 95.75% of accuracy, RNN model achieved 81.51% of accuracy and LSTM achieved 76.25% of accuracy for the 5th iterations. The reason for better performance is that sequential and important features are not avoided by the proposed CNN-RNN model, where imbalance problem is solved before the feature extraction process. However, the results must be improved and the performance of the proposed model must be tested with various real-time datasets for predicting the CCF tasks, which is considered as future work.

REFERENCES

[1] Richard, J. B. & David, J. H. (2001). Unsupervised Profiling Methods for Fraud Detection.london.

[2] Sanjeev, J., Kurian, T., & Christopher, W. S. B. (2011). Data mining for credit card fraud: A comparative study. Decision Support Systems, 602-613.

[3] Richard, J. B. & David, J. H. (2002). Statistical Fraud Detection: A Review. Statistical Science, 235-255.

[4] Chen, T. S., Lin, C. C., & Chen, R. C. (2006). A New Binary Support Vector System for Increasing Detection Rate of Credit Card Fraud. International Journal of Pattern Recognition and Artificial Intelligence, 227-239.

[5] Sahin, Y. & Duman, E. (2011). Detecting Credit Card Fraud by Decision Trees and Support Vector Machines. International Multiconference of Engineers and Computer Scientists (IMECS), 442-447.

[6] Whitrow, C., Hand, D. J., Juszczak, P., Weston, D., & Adams, N. M. (2009). Transaction aggregation as a strategy for credit card fraud detection. Data Mining and Knowledge Discovery, 30-55.

[7] Devi, D. U. & Kalyani, K. R. (2012). Fraud Detection of Credit Card Payment System by Genetic Algorithm. International Journal of Scientific & Engineering Research, 1-6.

[8] Hamdi, O. E. D. M. (2011). Detecting credit card fraud by genetic algorithm and scatter search. Expert Systems with Applications, 13057-13063.

[9] Xin, L. & Haiying, M. (2009). Application of Data Mining in Preventing Credit Card Fraud. International conference on management and service science (MASS), 1-6.

[10] Freisleben, B. & Aleskerov, B. R. E. (1997). CARDWATCH: A neural network based database mining system for credit card fraud detection. Computational Intelligence for Financial Engineering (CIFEr), 220-226.

[11] Duman, E. & Sahin, Y. (2011). Detecting Credit Card Fraud by ANN and Logistic Regression.international symposium on innovations in intelligent systems and applications (INISTA), 15-319.

[12] Carsten, A. W. P. (2008). Credit Card Fraud Detection Using Artificial Neural Networks Tuned by Genetic Algorithms (PhD thesis). Hong Kong University of Science and Technology.

[13] Adebayo, O. A. & Samuel, A. (2017). Oluwadare John O. Awoyemi, "Credit card fraud detection using Machine Learning: A Comparative Analysis. Computing Networking and Informatics (ICCNI), 1-9.

[14] Naga, S., Mannem, P., & Ganji, V. R. (2012). Credit card fraud detection using anti-k nearest neighboralgorithm. International Journal on Computer Science and Engineering (IJCSE), 1035-1039.

[15] Yu, W. F. & Wang, N. (2009). Research on Credit Card Fraud Detection Model Based on Distance Sum. International Joint Conference on Artificial Intelligence, 353-356.

[16] Zareapoora, M. & Shamsolmoali, P. (2015). Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier. Procedia Computer Science, 679-685.

[17] Patil, V. & Bhusari, S. (2011). Application of Hidden Markov Model in credit card fraud detection. International Journal of Distributed and Parallel Systems (IJDPS), 203-211.

[18] Thool, R. C. & Ingole, A.(2013). Credit Card Fraud Detection Using Hidden Markov Model and Its Performance. International Journal of Advanced Research in Computer Science and Software Engineering, 626-632.

[19] Ayse, B. & Ilker, E. E. D. (2013). A Novel and Successful Credit Card Fraud Detection. IEEE 13th International Conference on Data Mining Workshops, 1-10.

[20] Jon, T. S. Q, & Sriganesh, M. (2008). Real-time credit card fraud detection using computational intelligence. Expert Systems with Applications, 1721-1732.

[21] Dominik O. (2014). Fraud detection using self-organizing map visualizing the user profiles. Knowledge-Based Systems, 324-334.

[22] Maarof, M. A. & Abdallah, A. Z. A. (2016). Fraud detection system: A survey. Journal of Network and Computer Applications, 90-113.

[23] Yiğit, K. & Mehmet, U. Ç. (2016). Hybrid approaches for detecting credit card fraud.wiley - Expert Systems, 1-13.

[24] Cody, S., Muyang, S., Stephen, A., & Peter, B. G. R. (2017). Horse Race Analysis in Credit Card Fraud-Deep Learning, Logistic Regression, and Gradient Boosted Tree. Systems and Information Engineering Design Symposium (SIEDS), pp. 117-121.

[25] Djamila, A., Aleksandar, S., & Björn, O. A. C. B. (2016). Feature engineering strategies for credit card fraud detection. Expert Systems With Applications, 134-142.

[26] S. Patil, V. Nemade, and P. K. Soni, "Predictive Modelling For Credit Card Fraud Detection Using Data Analytics," Procedia Comput. Sci., vol. 132, pp. 385–395, Jan. 2018.https://doi.org/10.1016/j.procs.2018.05.199

[27] S. Xuan, G. Liu, Z. Li, L. Zheng, S. Wang, and C. Jiang, "Random forest for credit card fraud detection," in 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC), 2018, pp. 1–6.

[28] Boracchi, G., Caelen, O., Alippi, C., & Dal Pozzolo, A. (2017). Credit Card Fraud Detection: A Realistic Modeling a Novel Learning Strategy. IEEE Transactions on Neural Networks and Learning Systems, 2162-237X.

[29] Serol, B. & Ekrem, D. Y. S. (2013). A cost-sensitive decision tree approach for fraud detection. Expert Systems with Applications, 5916-5923.

[30] Liu, H. and Motoda, H. (1998). Feature extraction, construction and selection: A data

[31] mining perspective, Vol. 453, Springer Science & Business Media.

[32] Zulfiker, M. S., Kabir, N., Biswas, A. A., Nazneen, T. and Uddin, M. S. (2021). An indepth analysis of machine learning approaches to predict depression, Current Research in Behavioral Sciences 2: 100044.

[33] Kumari P, Mishra SP. Analysis of Credit Card Fraud Detection Using Fusion Classifiers. In: Advances in Intelligent Systems and Computing; 2019.

[34] Graves, A. (2012). Long short-term memory. In Supervised sequence labelling with recurrent neural networks (pp. 37–45). Springer.

[35] Kollias, D., & Zafeiriou, S. P. (2020). Exploiting multi-CNN features in CNN-RNN based dimensional emotion recognition on the OMG in-the-wild dataset. IEEE Transactions on Affective Computing.

[36] Masood, S., Srivastava, A., Thuwal, H. C., & Ahmad, M. (2018). Real-time sign language gesture (word) recognition from video sequences using CNN and RNN. In Intelligent engineering informatics (pp. 623–632). Springer.

[37] Zhang, X., Chen, F., & Huang, R. (2018). A combination of rnn and cnn for attention-based relation classification. Procedia Computer Science, 131, 911–917.

[38] Drumond, T. F., Viéville, T., & Alexandre, F. (2019). Bio-inspired analysis of deep learning on not-so-big data using data-prototypes. Frontiers in Computational Neuroscience, 12, 100.

[39] Kar, A. K. (2016). Bio inspired computing–a review of algorithms and scope of applications. Expert Systems with Applications, 59, 20–32.