IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON INTELLIGENT BIG DATA ANALYTICS FOR INTERNET OF THINGS, SERVICES AND PEOPLE

Received February 10, 2021, accepted February 21, 2021, date of publication February 26, 2021, date of current version March 23, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3062467*

# Internet Financial Fraud Detection Based on a Distributed Big Data Approach With Node2vec

**HANGJUN ZHOU[1], GUANG SUN[1,2], SHA FU[1], LINLI WANG[1], JUAN HU[1], AND YING GAO[1]**
[1]Department of Information Technology and Management, Hunan University of Finance and Economics, Changsha 410205, China
[2]College of Engineering, The University of Alabama, Tuscaloosa, AL 35401, USA

Corresponding author: Sha Fu (zhjnudt@gmail.com)

This work was supported in part by the Scientific Research Project of Education Department of Hunan Province under Grant 20K021 and Grant 20A080, in part by the Hunan Provincial Higher Education Teaching Reform Research Project under Grant HNJG-2020-1124, in part by the Social Science Foundation of Hunan Province under Grant 17YBA049, and in part by the 2011 Collaborative Innovation Center of Big Data for Financial and Economical Asset Development and Utility in Universities of Hunan Province.

**ABSTRACT** The rapid development of information technologies like Internet of Things, Big Data, Artificial Intelligence, Blockchain, etc., has profoundly affected people's consumption behaviors and changed the development model of the financial industry. The financial services on Internet and IoT with new technologies has provided convenience and efficiency for consumers, but new hidden fraud risks are generated also. Fraud, arbitrage, vicious collection, etc., have caused bad effects and huge losses to the development of finance on Internet and IoT. However, as the scale of financial data continues to increase dramatically, it is more and more difficult for existing rule-based expert systems and traditional machine learning model systems to detect financial frauds from large-scale historical data. In the meantime, as the degree of specialization of financial fraud continues to increase, fraudsters can evade fraud detection by frequently changing their fraud methods. In this article, an intelligent and distributed Big Data approach for Internet financial fraud detections is proposed to implement graph embedding algorithm Node2Vec to learn and represent the topological features in the financial network graph into low-dimensional dense vectors, so as to intelligently and efficiently classify and predict the data samples of the large-scale dataset with the deep neural network. The approach is distributedly performed on the clusters of Apache Spark GraphX and Hadoop to process the large dataset in parallel. The groups of experimental results demonstrate that the proposed approach can improve the efficiency of Internet financial fraud detections with better precision rate, recall rate, F1-Score and F2-Score.

**INDEX TERMS** Internet of Things (IoT), Internet finance, fraud detection, graph embedding algorithm, Node2Vec.

## I. INTRODUCTION

With the rapid development of the information technologies like Internet of Things, Big Data, Artificial Intelligence, Blockchain, etc., the digital life led by financial technology has profoundly affected people's consumption behaviors and changed the development model of the traditional financial industry to a certain extent [1]. In particular, technical products such as mobile payment, IoT financial services and Internet financial wealth management have penetrated into lots of aspects of economic and social activities. From 2014 to the present, the development momentum of China's Internet consumer finance industry has been good, and various mobile e-commerce companies have entered the consumer finance field through installment payments and small loans, which has promoted the development of related industries. Internet financial services based on consumer credits in China, such as Huabei launched by Ant Financial and Alipay of Alibaba Group, Jingdong Baitiao operated by JD.com, WeiLiDai launched by WeBank of Tencent, etc., have enabled consumers to enjoy the online shopping experience of ''consumption first, pay later'', and covered the e-commerce installment shopping, cash borrowing and other businesses. Especially in 2020, the COVID-19 pandemic [2] has caused a surge in online transaction volume and brought a large number of online customers to online service providers. It has

The associate editor coordinating the review of this manuscript and approving it for publication was Yang Xiao.

cultivated the habit of more groups of users to make online purchases and payments through mobile phones and IoT devices, which brings continuous impetus to the development of the Internet financial industry.

The rapid development of mobile and IoT financial payment services has not only provided convenience and efficiency for consumers, but also brought more hidden fraud risks. Due to the concealment of the complex network, there could be a breeding ground for fraudulent activities by criminals. The control of fraud risks is becoming more and more difficult and fraud cases occur frequently, which causes the fraud losses to commercial banks and financial institutions are also increasing. The continuous happening of Internet financial fraudulent problems, such as the agreement cash-out incident of Huabei and Taobao merchants, and ''Baitiao'' multiple fraud incidents, have not only damaged the legitimate rights and interests of the service platform, but also caused consumers to question the company's account security and risk identification capabilities.

A large number of violations are beyond the scope of the industry's existing laws and regulations, and industry regulation has always lagged behind the innovative development of Internet consumer finance, which makes the regulatory laws and regulations are often in a state of absence so that it impossible to deal with industry violations in a timely manner. Fraud, arbitrage, vicious collection and other phenomena are becoming more and more rampant in online financial service platforms, which has caused bad effects and huge losses to the development of consumer finance on Internet and IoT.

Fraud is an illegal or criminal deception aimed at obtaining financial or personal benefits. Fraud generally has the attributes of abnormal or unfair transactions. Due to the inconsistency with previous fund operation rules or other normal behaviors, fraudulent behavior presents various abnormal characteristics, including abnormal transaction amount, abnormal transaction time, abnormal transaction account, abnormal transaction IP, or abnormal personal credit rating.

Currently, fraud detection schemes in the industry mainly include rule-based expert systems and machine learning-based model systems. The rule-based expert system requires anti-fraud experts to manually analyze a large amount of normal and abnormal transaction data, accurately identify the behavior of fraudsters, find important features that can effectively distinguish fraud, and write expert rules for fraud detection. Therefore, the rule-based expert system strongly relies on the professional knowledge and business knowledge of the anti-fraud experts. If the experts cannot detect increasingly complex fraud patterns in a timely and keen manner, it will cause huge losses.

With the continuous increase of machine computing power, model systems based on machine learning have emerged. The machine learning-based model system is generally divided into four modules: data preprocessing, feature engineering, model training and model prediction [3]. Data preprocessing includes missing value processing, sampling and other steps. After the processing is completed, cumulative calculations are usually performed based on historical transaction data to convert the original data into characteristic data. After that, models such as machine learning regression or classification are used for training and evaluation on the data set. Finally, the model goes online for fraud detection.

However, as the scale of financial transaction data continues to increase dramatically, it is more and more difficult for rule-based expert systems and traditional machine learning model systems to detect transaction frauds or fraudulent behavior patterns from large-scale historical data when faced with massive data levels. In the meantime, as the degree of specialization of financial fraud continues to increase, fraudsters can evade fraud detection by frequently changing their own fraud methods. Nevertheless, it is difficult for fraudsters to change all their associated relationships. When the associated network graph can cover a large area, even if a fraudster or fraudulent behavior is careful, it may unwittingly reveal clues. Therefore, in the context of large-scale financial data, how to effectively mine the topological structure characteristics of the association network graph in real time and improve the effect of models for financial fraud detection is a new direction for researchers to explore. In this article, an intelligent and distributed Big Data approach for Internet financial fraud detection is proposed to implement graph embedding algorithm Node2Vec to learn and represent the topological features in the financial network graph into low-dimensional dense vectors, so as to intelligently and efficiently classify and predict the data samples of the large-scale dataset with the deep neural network. The approach is distributedly performed on the clusters of Apache Spark GraphX and Hadoop to process the large dataset in parallel. The groups of experimental results demonstrate that the proposed approach can improve the efficiency of Internet financial fraud detections with better precision rate, recall rate, F1-Score and F2-Score.

The rest of the article is organized as follows. Literature of related works is described in Section 2. Section 3 demonstrates the graph embedding algorithm of Node2Vec representation learning. An intelligent and distributed Big Data approach for Internet financial fraud detection is proposed in Section 4. In Section 5, groups of experiments are implemented to evaluate the efficiency of the proposed approach. Conclusions and future works are summarized in Section 6.

## II. RELATED WORKS

Beck points out that with the development of information technology, the threshold for people to accept financial services has been lowered, and some credit risks are caused by the problem of information asymmetry [4]. Weiss *et al.* indicate out that Internet financial risks are mainly due to adverse selection and moral hazard caused by information asymmetry, and the entire Internet financial industry will be affected [5]. Houston *et al.* believe that P2P online lending is beneficial to the development of small and medium enterprises and can effectively supplement traditional banking services [6]. Allen *et al.* find that there are many credit channels

in the United States and based on the research of American household credit models, and that household consumption, household income, credit banks and credit scale are obviously related [7]. Kregel studies the development trend of consumer finance and finds that the development of Internet consumer finance companies must fully consider the current market legal environment, financial market and consumer behavior factors, etc. Internet consumer finance is directly related to the current development of the national financial system [8]. Momparler *et al.* take the American Internet consumer finance company as the research object, study the risks and advantages of the Internet consumer finance platform, and design a related risk management model [9]. Jambulapati *et al.* point out that the function of Banking Act to prevent credit card risks and discuss the content of credit card bank supervision [10]. Through the data analysis of the consumption and credit segment in credit card usage, Shefrin *et al.* explore how families can make credit card usage decisions quickly and relatively frugally and provide online financial tools by a large credit card company to assist consumers in making decisions on credit card usage [11]. Hem *et al.* analyze the relationship between the Internet and credit card balances in American households through a survey of American consumer financial data [12]. By studying the different effects of factors such as education, income status, gender, age, race, etc. on credit card balances, analysis shows that education reduces credit card debt, while the Internet increases credit card debt. Andrew *et al.* use the time cross-sectional data of the financial situation of American consumers to analyze the difference between credit card interest rates and credit lines and study the changes that are taking place in the credit card market, and the results show that the lenders are using more information of digital finance than before [13]. Ficawoyi *et al.* analyze the positive relationship between Internet exposure levels and credit card default through surveys on consumer finance and income nodes [14]. The research points out that Internet access, low income, and male families are more likely to cause credit card defaults. Giudici *et al.* propose how to improve credit risk accuracy of P2P Internet financial platforms and of those who lend to small and medium enterprises [15]. The augment traditional credit scoring methods are put forward with ''alternative data'' that consist of centrality measures derived from similarity networks among borrowers and deduced from their financial ratios. The experimental findings suggest that the proposed approach improves predictive accuracy as well as model explainability.

In recent years, research in the field of financial fraud has mainly focused on bank fraud, insurance fraud, securities and commodity fraud, and other related types. Bank fraud includes fraud scenarios such as credit card fraud and money laundering. Insurance fraud includes fraud scenarios such as auto insurance fraud, group insurance fraud, and medical insurance fraud. Other related financial frauds include fraud scenarios such as marketing fraud and corporate fraud [16]. SOM model (Self Organizing Map) is proposed to build an unsupervised model for credit card fraud detections [17]. The SOM model does not require prior information, so the proposed automation system can continuously update the model by using newly added transaction data. Srivastava *et al.* use the K-means clustering algorithm to classify the transaction data set and build a fraud detection model based on the similarity of credit card fraud characteristics [18]. Zhou *et al.* use data mining techniques, such as decision trees, neural networks, Bayesian networks and other algorithms, to detect fraud in financial statements [19]. Liu *et al.* use the random forest algorithm on the financial fraud data set and compare it with other algorithms like logistic regression, nearest neighbor, decision tree and support vector machine, and find that the random forest algorithm has the highest accuracy and good interpretability [20]. Torgo *et al.* implement a hierarchical agglomerative clustering algorithm on the transaction data set to detect fraudulent transactions [21]. Dharwa *et al.* propose a kind of density-based clustering algorithm for fraud identification on the credit card transaction data sets [22]. Akoglu *et al.* believe that graph structure data has a strong expressive ability, so in the field of fraud detection association analysis methods could pay more attention to the connection between fraudsters and other individuals in a relationship network graph [23]. Aggarwal *et al.* construct a connected behavior model by dynamically dividing the network to detect structural anomalies in large-scale network flows [24]. Based on the viewpoint that anomalous nodes belong to multiple communities, Moradi *et al.* use a community detection algorithm to detect anomalies by data mining and finding communities that violate the community boundary rules [25]. Paula *et al.* implement Auto Encoder to detect export fraud related to data patterns, and verify it on Brazil's export data of goods and products in 2014 [26]. The model is able to detect the abnormal situation of at least 20 exporters. Pandey uses UCSD2009 data to prove the effectiveness of deep learning in the field of credit card fraud, but the model used is a shallow model containing only 2 fully connected layers and the framework used does not support GPU implementation [27]. Rushin *et al.* compare the effects of logistic regression, gradient descent tree, and deep learning in credit card fraud detection, and prove that the predictive ability of deep learning methods is better than the other two methods [28]. The classification result depends on the features constructed by domain expertise, and it does not consider other attributes of the data such as time attributes. Jurgovsky *et al.* take the fraud detection problem as a sequence classification task, and use long and short-term memory (LSTM) to make predictions [29]. Experimental results prove that LSTM effectively improves the accuracy of credit card fraud compared to random forest. Fang *et al.* propose an assessment of Light Gradient Boosting Machine model to achieve a higher total recall rate in real dataset and fast feedback comparing with Random Forest and Gradient Boosting Machine algorithm, and the proposed model's performance and efficiency in detecting credit card fraudulence are evaluated in the experiments [30].

## III. GRAPH REPRESENTATION LEARNING WITH NODE2VEC

Through studying a large number of Internet financial fraud cases, two important characteristics are found:

(1) The pattern of Internet financial fraud continues to evolve and develop over time, not just repeating the existing individual behavior patterns appeared in historical cases;

(2) With the advancement of anti-fraud technology, it is getting harder for individuals to commit Internet financial fraud. It needs to be organized and conducted through related and connected groups.

A graph is an abstract graph formed by a number of nodes and the edges connecting each node [31], [32]. It is usually used to describe a specific relationship between things. A relational network graph refers to a graph-based data structure composed of nodes and edges. Each node represents an entity, and each edge is the relationship between an entity and the other connected entity. The relationship network graph connects different entities together according to their relationships, thus it could provide the ability to analyze problems from the perspective of ''relationship''.

In anti-fraud applications, entities in the network graph, such as people, equipment, mailboxes, card numbers, etc., can be represented by nodes, and the relationships between these nodes in the business can be represented by edges. Through continuous construction and reproduction of the associated relationships hidden covertly in Internet financial frauds, fraud characteristics can be detected and corresponding risk control strategies can be designed. The graph algorithms can characterize various high-risk features in the Internet finance, such as batch attacks, intermediary participation, etc., which is more effective to identify abnormal group frauds from normal behaviors.

Graph embedding is an efficient technique to map a node in a graph from a high-dimensional sparse vector to a low-dimensional dense vector [33], which learns and represents the topological structure of the node in the network graph and the internal information of the node. Compared with traditional graph data mining methods, by applying graph embedding algorithms in anti-fraud business scenarios, it could obtain a global perspective to gain a clearer insight into the potential associations of different entities. Moreover, graph embedding algorithms can use graph models to process big data sets in the security field, which might be difficult for the computing ability of traditional graph data mining methods.

Node2Vec is a graph embedding algorithm [34] that introduces two biased random walk methods—BFS (Breadth First Search) and DFS (Depth First Search) on the basis of DeepWalk [35], so as to respectively learn and represent the structural equivalence and homophily of the network graph. Compared with random walk without any weight guidance, Node2Vec achieves the purpose of biased walk by introducing Return Parameter and In-out Parameter, that is, the entire random walk process is switching between BFS and DFS by setting different biases.

Structural equivalence is mainly used to characterize the structural similarity between nodes, so the vertices of the same structure should be similar in the representation learning of structural equivalence. BFS can traverse the adjacent vertex information around the vertex as much as possible, so BFS is more suitable for representing the structural equivalence of vertices. Through structural equivalence, we can find the vertices of two similar structures that are completely disconnected in the entire graph, which has important practical significance in anomaly fraud detection, risk control, roboadvisor recommendation, etc.

Homophily characterizes the similar homogeneity of adjacent vertices, which is similar to Word2Vec, that is, words that often appear together have the similar meaning in a high probability. Because DFS can macroscopically reflect the neighborhood of each vertex, DFS-based network homophily representations are more applied for group community discovery.

In the following Figure 1, it is a simple example of BFS and DFS in a graph. In the example, the vertex $u$ and vertex $S_1$ have the structural equivalence, while $u$ and $S_6$ are more similar in the homophily.

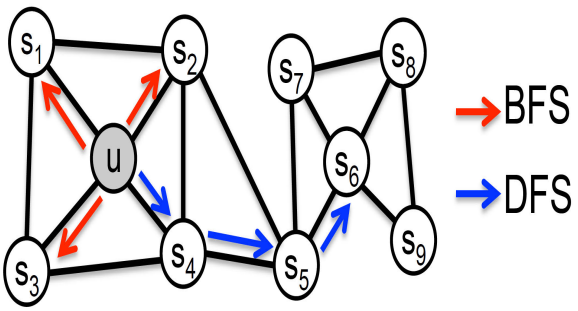



**FIGURE 1.** BFS and DFS search strategies from vertex *u*.

Suppose there is a graph $G = (V, E)$, $V$ is the set of vertices in the graph and $E$ is the set of edges in the graph. Then, the goal of graph embedding learning is to learn a function $f$ to map a vertex to a feature representation vector, where

$$f : V \rightarrow R^d \tag{1}$$

and $d$ is a pre-set hyperparameter that represents the dimension of the feature representation of each vertex. R denotes the real number set.

Thus, the final learning result is a matrix of size $|V| \times d$ parameters, and for each vertex $u \in V$, $N_S(u) \subset V$ denoted the network neighborhood of vertex $u$ with the sampling strategy $S$. Through extending the Skip-Gram neural network model of Word2Vec, the cost function with maximal log-probability is as follows:

$$\max_f \sum_{u \in V} \log Pr\left(N_S(u) | f(u)\right) \tag{2}$$

In order to make the optimization problem easier to handle, two assumptions are made.

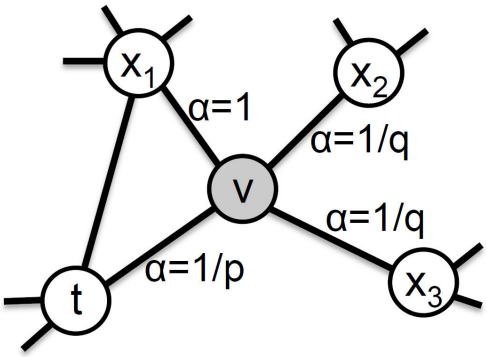



**FIGURE 2.** A case of Node2Vec random walk procedure.

(1) Conditional independence. Sampling each neighbor is independent of each other, so if the probability of sampling all neighbors need to be calculated, it can only multiply the probabilities of sampling each neighbor. The formula is as follows:

$$Pr\left(N_S\left(u\right)|f\left(u\right)\right) = \prod_{n_i \in N_S(u)} Pr\left(n_i|f\left(u\right)\right) \tag{3}$$

(2) Symmetry in feature space. In the feature space, the effect between two vertices is symmetrical. For example, an edge connects vertices $a$ and $b$, then when mapped to the feature space, the effect of $a$ on $b$ and the effect of $b$ on $a$ should be the same. The formula is as follows:

$$Pr\left(n_i|f\left(u\right)\right) = \frac{\exp\left(f\left(n_i\right)\cdot f\left(u\right)\right)}{\sum_{v\in V}\exp\left(f\left(v\right)\cdot f\left(u\right)\right)} \tag{4}$$

Combine the above three formulas to get the final result to be optimized:

$$\max_f \sum_{u\in V}\left[-\log Z_u + \sum_{n_i\in N_S(u)} f\left(n_i\right)\cdot f\left(u\right)\right] \tag{5}$$

For each vertex in $G$,

$$Z_u = \sum\nolimits_{v\in V}\exp\left(f\left(u\right)\cdot f\left(v\right)\right) \tag{6}$$

Because the computation for function $Z_u$ is particularly time-consuming in large network graph, Negative Sampling method is used to reduce time complexity. For each vertex $v$ in $V$, originally $f(v)$ would be computed. Then, negative sampling is applied to accelerate the training speed and improve the quality of the embedding vectors. Unlike the original update of all weights for each training sample, negative sampling only updates a small part of the weights of a training sample at a time, which will reduce the amount of computation in the gradient descent process.

Node2Vec uses the biased random walk that can achieve a smooth transition in BFS and DFS. For each walk, bias $\alpha$ is introduced to generate the biased random walk. In Figure 2, an example is illustrated about the 2nd order biased random walk procedure in Node2Vec.

Assuming that the walk in Figure 2 has been transitioned from the vertex $t$ to the vertex $v$, the transition probability from the vertex $v$ to the next vertex $x$ is:

$$\pi_{vx} = \alpha_{pq}\left(t, x\right)\cdot w_{vx} \tag{7}$$

The $\alpha_{pq}$ is calculated in the following formula:

$$\alpha_{pq}\left(t, x\right) = \begin{cases} \frac{1}{p}, & if\ d_{tx} = 0 \\ 1, & if\ d_{tx} = 1 \\ \frac{1}{q}, & if\ d_{tx} = 2 \end{cases} \tag{8}$$

And $d_{tx}$ denotes the shortest distance from the vertex $t$ to the vertex $x$. The value of $d_{tx}$ must be one of {0, 1, 2}. Parameter $p$ is called Return parameter and parameter $q$ is called In-out parameter.

Return parameter $p$ controls the possibility of returning to the last time vertex in one walk. If the value of $p$ is set relatively larger, then the probability of a walk from the vertex to the previous vertex is smaller, that is, the walk will go further away from the starting point. In this way, it is possible to control whether to walk a certain starting point field or a certain starting point deeper field. The parameter $p$ does not directly control whether the whole walking process is DFS or BFS. It only controls whether the walking area is always close to the starting point or gradually away from the starting point.

In-out parameter $q$ controls whether a walk moves inward or outward. When q>1, the random walk is biased towards the vertices close to the vertex $t$, that is to say, the walking area is more inclined to the neighborhood of the vertex $t$, which is BFS search mode. When q<1, the random walk is biased towards the vertices far from the vertex $t$, which is DFS search mode.

Therefore, the parameter $p$ determines the random walk area and the parameter $q$ determines the random walk mode. After the combination of the two parameters, through several biased walks, the structural equivalence and homophily of the vertices on the network graph could be more fully learned and represented.

## IV. AN INTELLIGENT AND DISTRIBUTED BIG DATA APPROACH FOR INTERNET FINANCIAL FRAUD DETECTION

In order to deal with the massively growing data set, distributed Big Data clusters are used to construct the intelligent risk management platforms for Internet financial fraud detections. In this article, our distributed Big Data approach deploys Apache Spark 3.0 as the big data infrastructure to distributedly implement the machine learning algorithms so as to improve the efficiency of Internet financial fraud detections. The Spark cluster manager provides resources to all worker nodes as per need and it operates all nodes accordingly. The Spark cluster manager mode is Hadoop Yarn, which works as a distributed computing framework to maintain job scheduling as well as resource management. In the cluster, master nodes and slave nodes are highly available. Hadoop Yarn splits up the functionalities of resource management

and job scheduling/monitoring into separate daemons by running a global Resource Manager (RM) and per-application Application Master (AM). The Hadoop HDFS is initiated on the cluster of data nodes where the dataset is distributedly stored. Then Spark environment is created and client node uses SparkContext to transform the processing request into Directed Acyclic Graph (DAG) in driver program. Once a DAG is generated, the graph is submitted to DAG scheduler. The role of DAG scheduler is to create physical execution plan and submit it to a real computation. This plan consists on physical unit of execution known as stages. In order to optimize the pipelining work by operations, sometimes several transformations will be merged into a single stage. Usually a DAG is analyzed into stage tasks and sent to the Resource Manager that has initiated a Node Manager on each Spark worker node. Each Node Manager receives one or several computing tasks and initiates Executor containers to run the tasks in parallel.

Graph computing is widely used in networks that include graph structures, such as credit networks and social networks with complex financial interactions. In these kind of networks, graph computing is required to calculate the connections between each other.

Especially, when the scale of a graph is very large, a distributed graph computing framework needs to be used. Spark GraphX is Apache Spark's API for graphs and graph-parallel computation, with a built-in library of common algorithms. Spark GraphX introduces a new graph abstract data structure by extending Spark RDD (Resilient Distributed Datasets): a directed multigraph that puts valid information into vertices and edges. Like every module of Spark, there is an abstract data structure based on RDD that is convenient for self-calculation. The scale of graphs in industrial applications is usually very large. In order to increase the processing speed and data volume, a distributed method is used to store and process graph data. There are roughly two ways of distributed storage of graphs: Edge Cut and Vertex Cut. In the early graph computing framework, the edge-segmented storage method was used. Later, considering most of the large-scale graphs in the real world are graphs with more edges than points, so it is more reasonable to store them in the Vertex Cut way. As shown in Figure 3, Vertex Cut can reduce the overhead of network transmission and storage. The underlying implementation is to store edges on each node, and when data exchange is performed, vertices are broadcasted between various machines for transmission. GraphX maintains a routing table internally, so that the required attributes can be transferred to the edge partition through the routing table mapping. For the interactive operation between a vertex and its neighbors, as long as the commutative law and associative law are satisfied, Vertex Cut is effective. However, the price of Vertex Cut is that the attributes of some vertices may be redundantly stored in multiple copies, and there is data synchronization overhead when updating vertex data. For the partitioning strategy of Vertex Cut, EdgePartition2D strategy is applied to assign edges to partitions using a 2D partitioning

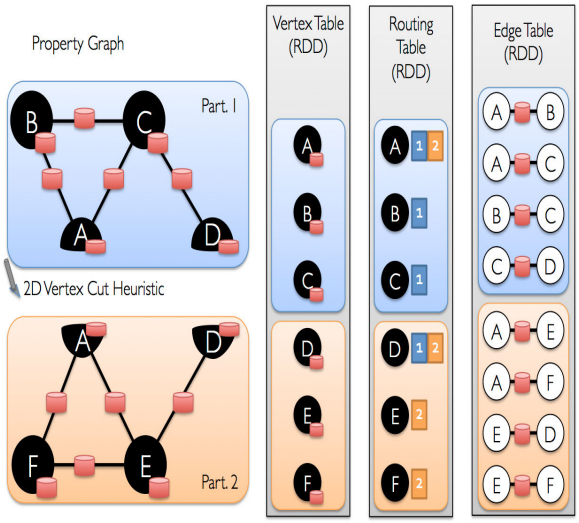



**FIGURE 3.** The distributed storage with vertex cut.

of the sparse edge adjacency matrix, so as to guarantee a bound on the number of vertex replication.

To improve the efficiency of Internet financial fraud detections, a distributed Big Data approach is proposed, which majorly includes four modules: data preprocessing module, normal data feature module, graph embedding module, prediction module. As it shown in Figure 4, data preprocessing module removes most of the empty value fields and repeated fields in the Internet finance dataset at first, and then extract and generate the graph topological dataset and normal sample dataset. Normal data feature module includes two procedures to divide the dataset into multiple data partitions and perform a statistical analysis on each field of each data partition so as to obtain the normal data features of the dataset. Graph embedding module constructs the network graph and implements the Node2Vec algorithm on Spark GraphX to learn and represent the topological features of a vertex in the network graph into a low-dimensional dense vector. Prediction module implements the classification model of deep neural network and accomplishes the final prediction results. The classification model contains four parts of the input part, convolution part, fully connected part and output part. Each predicted result is a floating number between 0 and 1, which represents the probability that a data sample is a fraudulent one.

In the implementation of Node2Vec algorithm on Spark GraphX, the modules of *spark.graphx.{EdgeTriplet, Graph, _}* and *graph.{GraphOps, EdgeAttr, NodeAttr}* are imported for the realization. The format of nodes in a graph is initiated the as (*nodeId*, *NodeAttr*(*neighbors*: Array[(long, Double)], *path*: Array[long])), where *nodeId* denotes the ID of a node, *neighbors* denotes the adjacent node array, *path* denotes the list of random walk. The format of edges in a graph is initiated the as (*srcId*, *dstId*, *EdgeAttr*(*dstNeighbor*: Array[long], *J*: Array[Int], *q*: Array[Double])), where *srcId* and *dstId* respectively denote the start node and end node of an edge, *dstNeighbor* denotes the neighbor nodes of end node, *J* and *q* are the values related to Alias sampling. Through

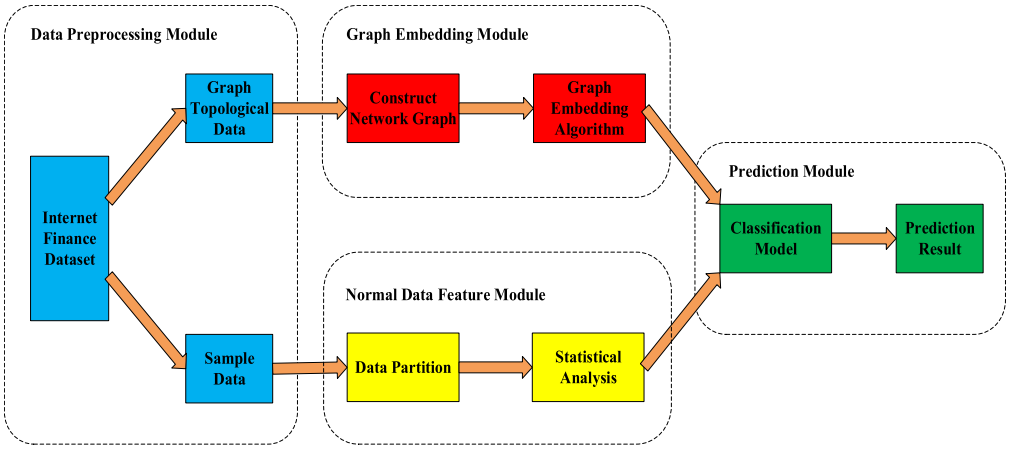



**FIGURE 4.** The distributed big data approach with four modules.

the functions of GraphX, *NodeAttr* and *EdgeAttr* are used in *EdgeTriplet*. According to the rule of Node2Vec biased random walk, the transition probability of *dstId* is calculated and stored in the *dstNeighbor, J, q* of *EdgeAttr*. The class of Node2VecModel is defined to train the data and accomplish the graph embedding according to the Word2Vec model.

## V. EXPERIMENTAL RESULTS

Groups of experiments are carried out on a cluster consisting of 30 identical machines, where one of them is designated as the master node and the rest are designated as worker nodes. Each machine has 8 physical cores and 64 GB of RAM. The operating system is CentOS 7 with Java Development Kit 10 and Scala 2.12. The stable release version of Apache Spark 3.0 is running on top of the cluster resource negotiator Hadoop Yarn and storage file system HDFS.

The original experimental dataset is obtained from a large Internet financial service provider in China. After the data preprocessing, there are 192586 data samples in the dataset in which the number of fraud samples is 4375. There are over 60 data fields of the dataset, such as initial amount, currency, income level, payment records, financial status, balance sheet, sale status, etc. For the reason to maintain data confidentiality of sensitive information, not all the data fields are mentioned. In order to evaluate the classification results of different machine learning models, the dataset is divided into 8 subsidiary datasets to conduct the cross-validation. Each time the ration of training data and testing data is nearly 4:1.

Groups of experiments are performed to compare the machine learning algorithms of Node2Vec, DeepWalk, SVM and the experimental results are evaluated. Evaluation results on precision with different datasets are demonstrated in Figure 5. Node2vec introduces biased parameters and BFS and DFS into the random walk sequence generation process on the basis of DeepWalk, all of the precision test results are over 70% and the highest rate is near 80%. DeepWalk uses the uniformly random walk to generate the sampling sequences so that the highest precision rate is 60% or so. The precision results of SVM are just over 30% and the highest

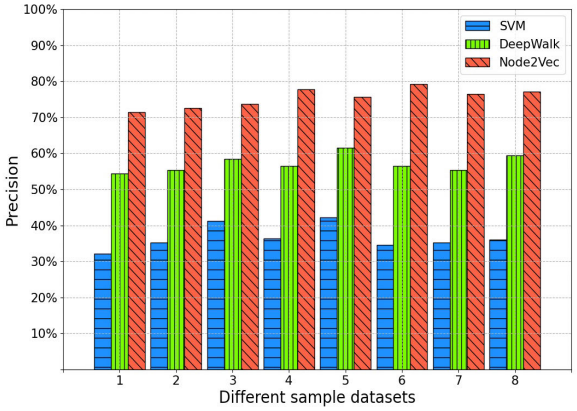



**FIGURE 5.** Evaluation on precision with different datasets.

one is under 43% for it is relatively incapable to learn the topological characteristics of the nodes in the network graph.

Node2Vec applies the structural equivalence to increase the number of sampling occurrences of neighboring nodes and reduce the variance of the neighboring nodes describing the current node, while it applies homophily to reflect the homogeneity between the current node and the further nodes. Therefore, the recall test results of Node2Vec are better than the other two algorithms, as shown in Figure 6. Among the fraudulent samples in the test, over 60% of them are detected and in some tests the results are near 70%. DeepWalk Maximizes the likelihood of random walk sequences and its recall rates are between 40% to 50%, which is better than those of SVM.

F1-Score is a measure indicator of classification problems and it considers recall rate and precision to be equally important. In Figure 7, the F1-Score test results of Node2Vec are between 67% to 73%, which is higher than the results of the other two comparative algorithms. This shows that Node2Vec is more stable in terms of overall performance and has better classification effects.

When detecting the Internet financial fraud behaviors, it is often more important to detect as many real frauds as possible

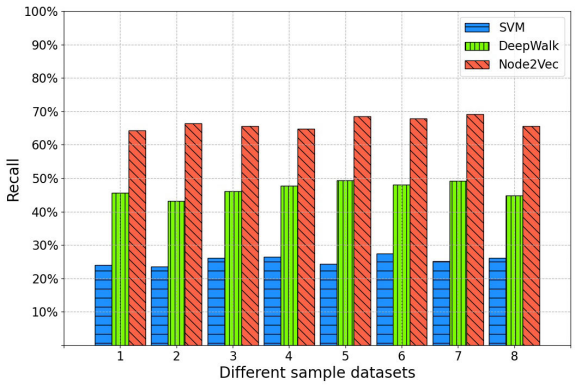



**FIGURE 6.** Evaluation on recall with different datasets.

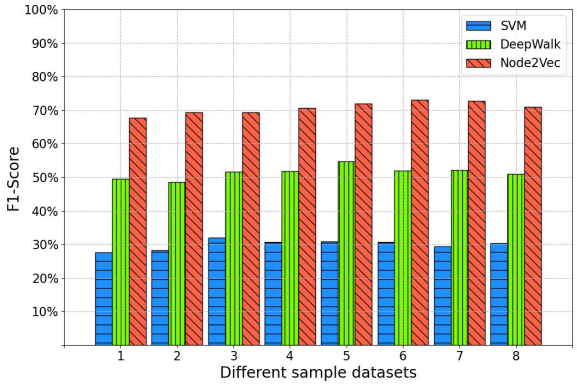



**FIGURE 7.** Evaluation on F1-score with different datasets.

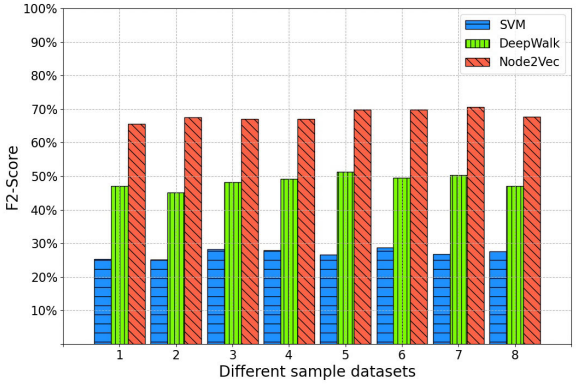



**FIGURE 8.** Evaluation on F2-score with different datasets.

so as to avoid the huge financial losses. Therefore, the F2-Score test results are show in Figure 8, in which the recall rate outweighs the precision rate. The F2-Score results of Node2Vec are higher than the 65% and the highest one is 71% or so. Most of them are close to 70%. For the DeepWalk and SVM, their results are respectively 19.7% and 41.1% lower than those of Node2Vec on average.

## VI. CONCLUSION

The occurrences of Internet financial fraud cases have caused huge losses to commercial banks or financial institutions. In order to enhance the efficiency of financial fraud detections, an intelligent and distributed Big Data approach is proposed in this article. The approach mainly includes four modules: data preprocessing module, normal data feature module, graph embedding module, prediction module. The graph embedding algorithm Node2Vec is implemented on Spark GraphX and Hadoop to learn and represent the topological features of each vertex in the network graph into a low-dimensional dense vector, so as to improve the classification effectiveness of deep neural network and predict the fraudulent samples of the dataset. The experiments evaluate the indicators of precision rate, recall rate, F1-Score and F2-Score, and the results show that due to the Node2Vec properties of structural equivalence and homophily, the features of samples can be better learned and represented and the proposed approach is better than the comparative methods. In future work, the inductive graph embedding network algorithms, such as GraphSage, PinSage, etc., would be improved and implemented to effectively learn the features of newly generated vertices in a dynamic network graph, so as to achieve the better effect of financial fraud detection.

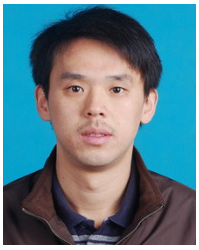
## REFERENCES

[1] U. Paschen, C. Pitt, and J. Kietzmann, "Artificial intelligence: Building blocks and an innovation typology," *Bus. Horizons*, vol. 63, no. 2, pp. 147–155, Mar. 2020.
[2] P. Yu, Z. Xia, J. Fei, and S. K. Jha, "An application review of artificial intelligence in prevention and cure of COVID-19 pandemic," *Comput., Mater. Continua*, vol. 65, no. 1, pp. 743–760, 2020.
[3] L. Shen, X. Chen, Z. Pan, K. Fan, F. Li, and J. Lei, "No-reference stereoscopic image quality assessment based on global and local content characteristics," *Neurocomputing*, vol. 424, no. 2, pp. 132–142, Feb. 2021.
[4] H. Beck, "Banking is essential, banks are not. The future of financial intermediation in the age of the Internet," *Netnomics*, vol. 3, no. 1, pp. 7–22, 2001.
[5] G. N. F. Weiss, K. Pelger, and A. Horsch, "Mitigating adverse selection in P2P lending–empirical evidence from prosper.com," *SSRN Electron. J.*, vol. 19, no. 7, pp. 65–93, 2010.
[6] Y. Houston, C. Jongrong, J. H. Cliff, and H. Y. Chih, "E-commerce, R&D, and productivity: Firm-level evidence from Taiwan," *Inf. Econ. Policy*, vol. 18, no. 5, pp. 561–569, 2013.
[7] F. Allen, J. Mcandrews, and P. Strahan, "E-finance: An introduction," *Center Financial Inst. Work. Papers*, vol. 22, no. 1, pp. 25–27, 2012.
[8] J. A. Kregel, "Margins of safety and weight of the argument in generating financial fragility," *J. Econ. Issues*, vol. 31, no. 2, pp. 543–548, Jun. 1997.
[9] A. Momparler, C. Lassala, and D. Ribeiro, "Efficiency in banking services: A comparative analysis of Internet-primary and branching banks in the US," *Service Bus.*, vol. 7, no. 4, pp. 641–663, Dec. 2013.
[10] V. Jambulapati and J. Stavins, "Credit CARD act of 2009: What did banks do?" *J. Banking Finance*, vol. 46, no. 9, pp. 21–30, Sep. 2014.
[11] H. Shefrin and C. M. Nicols, "Credit card behavior, financial styles and heuristics," *Bus. Res.*, vol. 67, no. 8, pp. 1679–1687, 2014.
[12] H. C. Basnet and F. Donou-Adonsou, "Internet, consumer spending, and credit card balance: Evidence from US consumers," *Rev. Financial Econ.*, vol. 30, pp. 11–22, Sep. 2016.
[13] A. Davis and J. Kim, "Explaining changes in the US credit card market: Lenders are using more information," *Econ. Model.*, vol. 61, no. 2, pp. 76–92, Feb. 2017.
[14] F. Donou-Adonsou and H. C. Basnet, "Credit card delinquency: How much is the Internet to blame?" *North Amer. J. Econ. Finance*, vol. 48, no. 4, pp. 481–497, Apr. 2019.
[15] P. Giudici, B. Hadji-Misheva, and A. Spelta, "Network based credit risk models," *Qual. Eng.*, vol. 32, no. 2, pp. 199–211, Apr. 2020.
[16] E. W. T. Ngai, Y. Hu, Y. H. Wong, Y. Chen, and X. Sun, "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature," *Decis. Support Syst.*, vol. 50, no. 3, pp. 559–569, Feb. 2011.
[17] V. Zaslavsky and A. Strizhak, "Credit card fraud detection using self-organizing maps," *Inf. Secur.*, vol. 25, no. 18, pp. 41–48, 2006.

[18] A. Srivastava, A. Kundu, S. Sural, and A. Majumdar, "Credit card fraud detection using hidden Markov model," *IEEE Trans. Dependable Secure Comput.*, vol. 5, no. 1, pp. 37–48, Mar. 2008.
[19] W. Zhou and G. Kapoor, "Detecting evolutionary financial statement fraud," *Decis. Support Syst.*, vol. 50, no. 3, pp. 570–575, Feb. 2011.
[20] C. Liu, Y. Chan, S. H. A. Kazmi, and H. Fu, "Financial fraud detection model: Based on random forest," *Int. J. Econ. Finance*, vol. 7, no. 7, pp. 5–7, Jun. 2015.
[21] L. Torgo and E. Lopes, "Utility-based fraud detection," in *Proc. Int. Joint Conf. Artif. Intell.*, 2011, pp. 15–17.
[22] J. N. Dharwa and A. R. Patel, "A data mining with hybrid approach based transaction risk score generation model (TRSGM) for fraud detection of online financial transaction," *Int. J. Comput. Appl.*, vol. 16, no. 1, pp. 18–25, Feb. 2011.
[23] L. Akoglu, H. Tong, and D. Koutra, "Graph based anomaly detection and description: A survey," *Data Mining Knowl. Discovery*, vol. 29, no. 3, pp. 626–688, May 2015.
[24] C. C. Aggarwal, Y. Zhao, and P. S. Yu, "Outlier detection in graph streams," in *Proc. IEEE 27th Int. Conf. Data Eng.*, Apr. 2011, pp. 399–409.
[25] F. Moradi, T. Olovsson, and P. Tsigas, "Overlapping communities for identifying misbehavior in network communications," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*, 2014, pp. 398–409.
[26] E. L. Paula, M. Ladeira, R. N. Carvalho, and T. Marzagão, "Deep learning anomaly detection as support fraud investigation in Brazilian exports and anti-money laundering," in *Proc. 15th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2016, pp. 954–960.
[27] Y. Pandey, "Credit card fraud detection using deep learning," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, pp. 981–984, 2017.
[28] G. Rushin, C. Stancil, M. Sun, S. Adams, and P. Beling, "Horse race analysis in credit card fraud—Deep learning, logistic regression, and gradient boosted tree," in *Proc. Syst. Inf. Eng. Design Symp. (SIEDS)*, Apr. 2017, pp. 117–121.
[29] J. Jurgovsky, M. Granitzer, K. Ziegler, S. Calabretto, P.-E. Portier, L. He-Guelton, and O. Caelen, "Sequence classification for credit-card fraud detection," *Expert Syst. Appl.*, vol. 100, pp. 234–245, Jun. 2018.
[30] Y. Fang, Y. Zhang, and C. Huang, "Credit card fraud detection based on machine learning," *Comput., Mater. Continua*, vol. 61, no. 1, pp. 185–195, 2019.
[31] Z. Pan, X. Yi, Y. Zhang, B. Jeon, and S. Kwong, "Efficient in-loop filtering based on enhanced deep convolutional neural networks for HEVC," *IEEE Trans. Image Process.*, vol. 29, pp. 5352–5366, 2020.
[32] Y. Wu, B. Wang, and W. Li, "Heterogeneous hyperedge convolutional network," *Comput., Mater. Continua*, vol. 65, no. 3, pp. 2277–2294, 2020.
[33] Z. Pan, X. Yi, Y. Zhang, H. Yuan, F. L. Wang, and S. Kwong, "Frame-level bit allocation optimization based on<?*brk*? > video content characteristics for HEVC," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 16, no. 1, pp. 1–20, 2020.
[34] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 855–864.
[35] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 701–710.

**HANGJUN ZHOU** received the Ph.D. degree in computer science from the National University of Defense Technology. He is currently a Professor with the Hunan University of Finance and Economics and a Senior Data Development Engineer with the Ministry of Industry and Information, China. His research interests include big data mining, artificial intelligence, the Internet of Things, and financial risk management. In these fields, he was the first author of more than 40 articles, one national standard, eight national inventions, four treatises, and eight provincial research projects.

**GUANG SUN** received the Ph.D. degree in computer science from Hunan University, China, in 2012. He is currently a Professor with the Institute of Big Data, Hunan University of Finance and Economics, Changsha, Hunan, China. He is also a Visiting Scholar with The University of Alabama. His research interests include sensor networks security, information hiding (with a focus on software watermarking and software birthmarking), big data analysis, and visualization. His research has been supported by the Open Foundation for the University Innovation Platform from the Hunan Province, China.



**SHA FU** received the master's degree in computer science from Hunan University. He is currently a Professor with the Hunan University of Finance and Economics. He joined a project from the National Natural Science Foundation of China. He led the National Science Education "13th Five-Year" Project Planning for one time, the Scientific Research Fund of Hunan Provincial Education Department for three times, and the Hunan Province Science and Technology Program Project for one time. He published one book and more than 40 articles.

**LINLI WANG** was born in Shiyan, Hubei, China, in 1999. She is currently pursuing the degree in information management and information systems with the Hunan University of Finance and Economics. For past four years, she has won the School-Level Merit Student and the First-Class Scholarship for two consecutive years. She participated in writing a paper "Stock Walk Model Based on Data Forecasting", which is included in the China Turing Conference. She owns the software copyright of RHI Model Design: Discussion on Accurate Funding of Universities. Meanwhile, she has interests in research and development of financial risk management with mobile networks and the Internet of Things, big data mining, artificial intelligence, and cloud computing.

**JUAN HU** was born in Changsha, Hunan, China, in 2000. She is currently pursuing the degree in data science and big data technology with the Hunan university of Finance and Economics. She has won three college level first-class scholarships. She often participates in a number of academic practices, such as data modeling for financial risk management, big data engine, and parallel and distributed computing.

**YING GAO** is currently pursuing the degree in data science and big data technology with the School of Information Technology and Management, Hunan University of Finance and Economics. Her main research interests include big data processing, financial risk evaluation, and user behavior analysis.