2018 IEEE 4th International Conference on Computer and Communications

A Click Fraud Detection Scheme based on Cost sensitive BPNN and ABC in Mobile Advertising

Xin Zhang, Xuejun Liu*, and Han Guo
School of Computer Science and Technology
Nanjing Tech University
Nanjing, China
e-mail: 1078683220@qq.com; lxj_njgd@163.com; 1577717382@qq.com

***Abstract*—Click fraud happens in cost per click ad networks where publishers charge advertisers for each click. Click fraud is posing the huge loss to the mobile advertising industry. The conventional technologies use ensemble machine learning methods, neglecting the cost of incorrect classification for a fraud publisher is higher than a normal publisher. An effective classification model for variable click fraud is proposed in this paper. Cost-sensitive Back Propagation Neural Network is combined with the novel Artificial Bee Colony algorithm in this research (CSBPNN-ABC). Feature selection is synchronously optimized with BPNN connection weights by ABC to reduce the interaction between features and weights. Cost Parameters are added to BPNN by correcting the error function. Experiments on real world click data in mobile advertising show that its superior classification performance compared with the state-of-the-art technology.**

Keywords- click fraud; cost-sensitive; Back Propagation Neural Network; Artificial Bee Colony; feature selection

I. INTRODUCTION

Nowadays, mobile advertising enjoys 51% share of the whole digital market with the rapid development of mobile devices [1]. While, almost 30% income has been wasted because of frauds generated by bot networks or paid human beings [1]. Cost Per Click (CPC) is the main billing method of search engine advertising, such as Google, YAHOO. The business model of CPC bidding rankings makes the publishers get great benefits and objectively contributes to the occurrence of click fraud [2]. Fraud clicks on an ad with a malicious intent but advertisers need to pay for valueless clicks [1]. Naturally, effective and stable click fraud detections and prevention mechanisms are necessary.

At present, the state-of-the-art technology is ensemble machine learning methods [1]. However, ensemble method will make the model too complex and may cause the model to be overfitting. Neural network has the advantages of self-learning, self-organization, nonlinear mapping, and large scale parallel computing. It is very suitable for the variable click fraud detection environment [3]. Back Propagation Neural Network (BPNN) is a multilayer feedforward network trained by error back propagation algorithm, which is one of the most widely used neural network models with strong robustness and good fault tolerance. To capture the latent patterns of fraud clicks and avoid the model overfitting, BPNN is used to complete classification work of publishers.

Click fraud detection is faced with the problem of high-dimensional feature sets, leading to a decrease in accuracy and an increase in training complexity. So, we use Artificial Bee Colony (ABC) algorithm [4] to search for the best feature subset and improve the performance of classifier. Because the feature subset and the parameters of neural network are interacted, ABC is used to realize simultaneous optimization of feature selection and parameters selection [5].

Most click fraud detection models so far do not take into account the cost of fraud samples and normal samples. In such a problem, data are generally extremely imbalanced. Some studies use sampling techniques to deal with the imbalance of training datasets [1]. While different misclassification costs exist simultaneously with the imbalance of click data. The cost of incorrect classification for a normal publisher is far below a fraud publisher. In this paper, we use SMOTE to solve the imbalanced of the training click dataset and cost parameters are added to BPNN to minimize misclassification costs, but not global error [6].

The contributions of this paper are summarized below:

1) BPNN is applied in the variable click fraud detection environment;
2) To avoid local optimization of BPNN and feature redundancy, Artificial Bee Colony is introduced to simultaneously optimize the feature selection and weights simultaneously. Genetic Algorithm (GA), Particle Swarm Algorithm (PSO) and Ant Colony Optimization (ACO) have been compared with ABC;
3) Cost Parameters are added to BPNN by changing cost coefficients to correct the error function;
4) An oversampling technique, SMOTE is applied to deal with the extreme imbalanced training click dataset;
5) Existing techniques mostly pay attention to web advertising, but the research in mobile advertising is not enough. Our research is based on real world click data in mobile advertising and compare with the state-of-the-art technology.

II. RELATED WORKS

It is very common to use malicious code, bot network and other automatic clicks to increase clicks, which can be screened by a professional third-party monitor or graphic verification codes. However, human malicious clicks with weak randomness are very similar to normal clicks. The common way is mining traffic logs to detect this type of frauds [1]. Kitts [7] designed a data mining system to detect massive click fraud attacks. Different machine learning models have been proposed to overcome the click fraud detection problem. J48, Repetition Tree, Random Forest (RF) [1, 3, 8], Naive Bayes (NB), tree based methods, Support Vector Machines (SVM) [9, 10] are some of the algorithms used for this problem. Yet click fraud has a complex pattern with weak randomness. The ensemble machine learning methods were proposed to improve the performance of classifier. Such as, Booting, Bagging, stacking [1], [9], [10]. Neural network has been widely studied over the recent decades in various fields. In other field of detection, artificial neural networks have been well applied [3], [6].

In order to better train model parameters, some optimization algorithms are applied, such as GA, PSO, and ACO. ABC optimization algorithm is introduced recently [4], which has been applied to optimize the neuron connections [5]. At same time, feature selection is required to avoid feature redundancy. Feature selection is actually a combinatorial optimization problem. So we can also use ABC to search for the best feature combination. Finally, the hybrid methodology by ABC and cost-sensitive neural BPNN (CSBPNN-ABC) is proposed for the click fraud defection in mobile advertising.

III. Dataset

Real world click fraud data from a global mobile advertising company-BuzzCity are commonly used for the click fraud detection problem in mobile advertising [8], [10]. The raw data consist of two parts: *publisher dataset* and *click dataset*. The publisher dataset contains the profile information of the publisher (bankaccount, address and publisherid) and label of the publisher (Ok, Fraud and Observation). The Observation label included small number of publishers who are new or have high click traffic but not yet deemed as fraudulent. In this paper, we put Observation and OK to Normal. The click dataset contains click traffic associated with publisher dataset by publisherid. Table I lists the basic feature in the click dataset. Table II shows the statistics of training and testing datasets used in this paper. The statistics of samples shows an imbalanced distribution of data reached above 1: 35. To overcome this problem, we use the over sampling technique, SMOTE, to generate a more balanced training set. Yet, the misclassification cost exists at the same time, we also introduce cost parameters to BPNN.

TABLE I. Description of Feature in Click Dataset

Basic feature	Description
publisherid	Identifier of a publisher
id	Identifier of a click
numericip	IP address of the clicker
deviceua	Mobile device used by the clicker
campaignid	Campaign ID of an advertisement campaign
usercountry	Country of the click
clicktime	Timestamp of the click
channel	Publisher's channel type
publisherid	Identifier of a publisher
referredurl	An URL where the ad is clicked

TABLE II. Statistics of Datasets Used in This Study

Dataset	format	No. of clicks	No. of Publishers		
			Fraud	*Normal*	*Total*
Training	CSV	5,862,839	157	5,988	6,145
Testing	CSV	2,598,815	82	2,918	3,000

The basic features do not reflect the global behavior of clicks. It cannot be used directly for the construction of the model. Statistical methods are used to derive new features. Oentaryo [10] derived 118 features. The list can be available at http://clifton.phua.googlepages.com/feature-list.txt. To improve accuracy, and decrease training complexity, feature selection is necessary. In this study, an optimal feature subset as inputs of BPNN are selected by ABC. Specific methods we proposed will be introduced in section IV.

IV. The Proposed Methodology

In this section, we firstly provide a description of CS-BPNN architecture for click fraud detection. Secondly, we use ABC to optimize BPNN connection weights and feature selection synchronously. Thirdly, the error function is corrected by adding cost parameters to BPNN. Finally, the cost sensitive BPNN model based on ABC is employed to the problem of click fraud detection.

To summarize, Fig. 1 shows our detection framework, it consists of four components: data preprocess, feature selection, CS-BPNN classification model training phase, prediction phase. For data preprocess, we calculated feature vector values from millions of click data by user behavior analysis. In the general process of feature selection, we adopt Artificial Bee Colony algorithm to search for the best feature subset. The data of click fraud is extremely imbalanced and the cost of incorrect classification for a normal publisher is far below a fraud publisher. We use SMOTE to generate synthetic fraud samples and convert BPNN into a cost-sensitive classifier. In order to improve the performance of the classifier, BP neural network weights are optimized at the same time with feature subset selection.

A. Back Propagation Neural Network

The click data are mass and random, and the association between derived click features and click fraud tendentiousness is often complex and nonlinear, so it is not easy to establish the detection model. Neural network is a machine learning technology that simulates human brain neural systems to realize artificial intelligence [3], and establishes the detection model based on the existing data. Therefore, BPNN is a good choice for fraud detection to complete the classification of publishers. In fact, the essence of neural network is to fit the real functional relationship between feature and target through weights and activation function. The output of each neuron in the hidden layer is

$$H_j = f_i\left(\sum_{i=1}^{n} w_{ij}^{(1)} x_i + b_j^{(1)}\right) \tag{1}$$

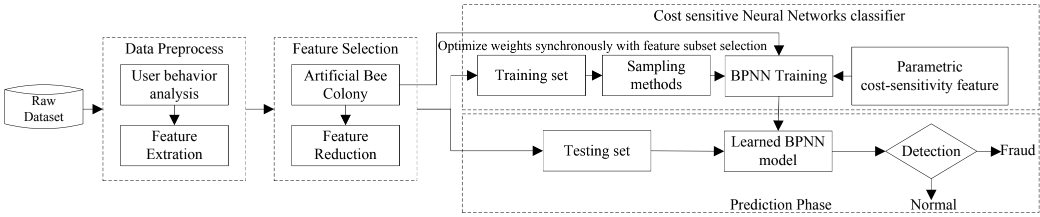

Figure 1. The overall framework of the proposed CSBPNN-ABC click fraud detection system

where j=1, 2,…, q, q is the number of neurons in this layer, l represent the neuron in the l layer, n is the total number of inputs to neuron j, x_i is the i-th input of the neuron, $w_{ij}^{(l)}$ represents the connection weights between the neurons i and the neurons j, and $b_j^{(l)}$ is the bias. $f(\square)$ is an activation function. The output of each neuron in the output layer is

$$O_k = f(\sum_{j=1}^{q} w_{jk}^{(l)} H_j + b_k^{(l)}) \tag{2}$$

where k=1, 2,…, p, and p is the number of outputs. $f(\square)$ can be a step, sign, sigmoid, Gaussian or linear function, which is defined according to the different task. In this paper, we choose sigmoid function. The neural network utilizes the error function to carry out the back propagation to adjust the weights. The define of conventional error function is

$$E = \frac{1}{2}\sum_{k=1}^{p}(d_k - O_k)^2 \tag{3}$$

where d_k is expected outputs. As in (3), adjusting the connection weights will change the error and BP neural network is prone to fall to local optimality. So the ABC algorithm is used to global optimize the weights to minimize error. The speed of convergence rate is improved at the same time. The inputs of BPNN is feature subset also searched by ABC. Because of the different misclassification costs of fraud and normal class, we introduce the cost feature to correct the error function. That does not try to minimize global misclassification error, but misclassification costs. The process will be described in next subsection. The CS-BPNN with two hidden layers created for click fraud detection is shown in Fig. 2.

B. Artificial Bee Colony

Artificial Bee Colony proposed by Karaboga solve the optimization problems by simulating the foraging behavior of bees [5]. Bees carry out different activities and realize the sharing and communication of information so as to find the optimal solution. The process of Artificial Bee Colony optimization algorithm is given in Fig. 3. There are three types of bees (employed, onlooker, and scout) [4] and the amount of onlooker bees are equal to employed bees. Considering that ABC algorithm has strong optimization ability in multivariable function problems and weights are affected by the input of features, ABC is used to optimize feature subset selection and BPNN connection weights synchronously. In this paper, the optimal process is:

1) Initialization Phase: As bees, N solutions occur randomly. Each solution S_i (i = 1, 2, …, N) is a food source D-dimension vector containing the weights, bias and features to be optimized within the problem as showen in Table III.

2) Employed Bees Phase: The duty of employed bees is increasing nectar amounts (fitness). A parameter x_i^j in S_i is randomly chosen as initial solution. Then, Use (4) to produce a new parameter v_i^j. If the fitness of v_i^j is better than x_i^j, v_i^j is assigned to x_i^j.

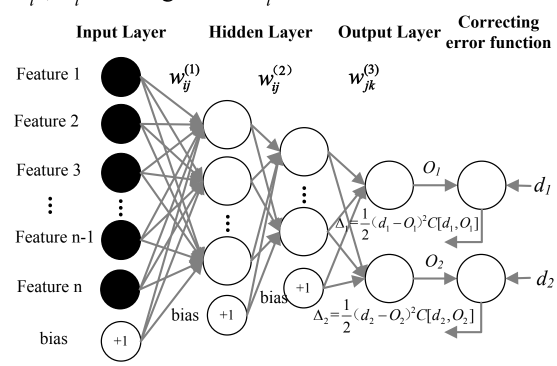

Figure 2. Architecture of the CS-BPNN for click fraud detection

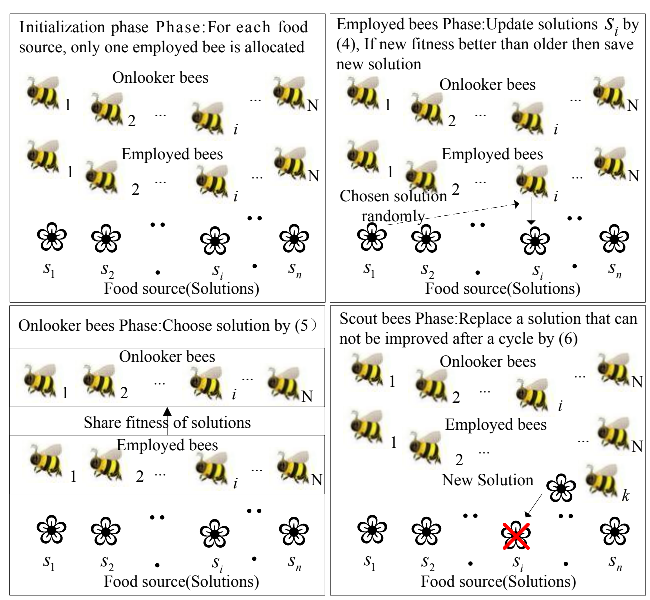

Figure 3. Diagram of the ABC algorithm

TABLE III. PARAMETER IN A FOOD SOURCE VECTOR

Connection weights w_{ij}	Bias b_j	Feature selection information
$x_w^1, x_w^2, \ldots, x_w^r$	$x_b^{r+1}, x_b^{r+2}, \ldots, x_b^t$	$x_F^{t+1}, x_F^{t+2}, \ldots, x_F^D$

$$v_i^j = x_i^j + \phi_i^j (x_i^j - x_k^j) \tag{4}$$

where $j \in \{1, 2, \cdots, D\}$, $k \in \{1, 2, \cdots, N\}$, x_k^j is a same parameter value from one randomly chosen solution k, and $\phi_i^j \in [-1, +1]$.

3) Onlooker Bees Phase: The employed bees share collected information about food sources with onlooker bees [5]. The onlooker bees recalculate the fitness of solutions and the probability of food source visited is calculated.

$$p_i = fit_i \Big/ \sum_{i=1}^{N} fit_i \tag{5}$$

where fit_i is the fitness of S_i. The fitness function in this paper is global misclassification costs.

4) Scout Bees Phase: If the fitness of a solution has not improved after a round of attempts, then the employed bee is changed into a scout. The scount will abandoned this solution and produce a new solution to replace it. Each parameter in the new solution is created by (6)

$$x_i^j = x_{\min}^j + rand(0,1)(x_{\max}^j - x_{\min}^j) \tag{6}$$

where $x_{\min}^j$ and $x_{\max}^j$ are the lower and upper bounds of created parameter values.

Finally, the best fitness of solution is judged whether or not to reach the stop condition. If it is reached, we get the best feature subset and parameters of neural network. Otherwise, the above steps will be repeated until reach the stop condition. When using the ABC optimization algorithm, several control parameters need to be determined. There are colony size of bees (2 × N), abandonment limit, maximum number of search cycles (MCN), and lower and upper bounds of search space. In section V, the final values of these control parameter values will be given.

C. The proposed cost sensitive classifier

Another advantage of our proposed classifier is the introduction of cost sensitive parameters. In the field of detection, samples are usually extremely imbalanced. Minority class, which we are interested in, not only number is small but also the cost of misclassification is larger. Considering that data imbalance and different misclassification cost exist simultaneously, we firstly use SMOTE to structure a more balanced training set by generating synthetic minority samples. The synthetic minority sample is generated by (7).

$$Pub_{new} = Pub_i + rand(0,1) \times (Pub_{ij} - Pub_i) \tag{7}$$

where Pub_i is a minority sample, and Pub_{ij} is one of the K neighbor sample of Pub_i. Sampling rate and nearest neighbor number K is given in section 5.

Then, we convert BPNN into a cost-sensitive learner by correcting the error function as in Fig.2. The error function of a sample regardless of cost-sensitivity as shown in (3). We introduce cost parameter to minimize global misclassification costs, not the classification error.

$$\min \quad \frac{1}{2m} \sum_{u=1}^{m} \sum_{k=1}^{p} (d_k(u) - O_k(u))^2 C[d_k(u), O_k(u)] \tag{8}$$

where u=1, 2,…, m, m is the number of training samples, and $C[d_k(u), O_k(u)]$ is cost of classifying $d_k(u)$ to $O_k(u)$. The cost of classifying samples correctly is 0. There are two kinds of misclassification cost in the classification process: the cost of incorrectly classifying a Fraud publisher to Normal (C_{FN}), and the cost of classifying a Normal publisher to Fraud (C_{NF}) ($C_{FN} > C_{NF}$).

V. EXPERIMENTS AND ANALYSIS

A. Experiment Environment and Settings

We conducted our experiments in a server with Intel core i7-6800k CPU, 32G RAM, GTX 1080Ti GPU and Ubuntu 16.04LTS system. To build the click fraud detection system, data preprocessing, calculation of features value, data balance processing, feature selection and CS-BPNN model optimized by ABC algorithm are all implemented in Python.

In our experiments, the dataset contains two parts as described earlier in Section III: 5,862,839 clicks for 6,145 publishers in training dataset and 2,598,815 clicks for 3,000 publishers in testing dataset. Firstly, we compute the value of features for each publisher by scientific computing libraries, such as numpy, pandas, scipy. The missing data in raw data are set to 0 and all value of features are normalized in the range [0.0, 1.0]. As there are a great deal of click stream data, computing their features was a major challenge.

SMOTE is applied to generate synthetic minority samples. We fix the sampling rate R = 8 and nearest neighbor number K = 11 after many experiments.

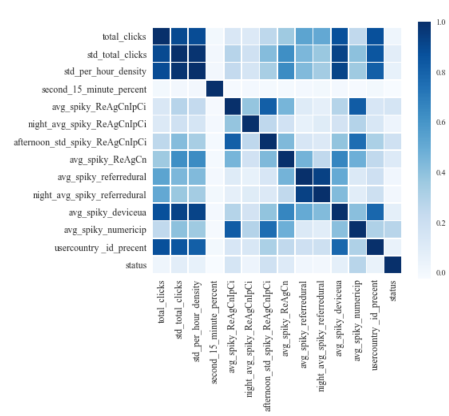

Figure 4. Correlation plot of 13 features including status

The feature selection and parameters of CS-BPNN is optimized by ABC at the same time. We initial the weights and bias of BPNN and the feature subset. The control parameters of the ABC are generated in an automated manner [11]. Finally, we obtain that colony size N=30, limit is 1000, MCN = 620, and lower/upper bounds is [-4.5, 1.0]. The optimal feature subset contains 13 features. Fig.4 plots correlations among 13 behavior features including status in the training set, which displays feature diversity. From Fig.4, we can know the similarity between those features is not high, and our proposed method effectively avoid feature redundancy. The different cost ratio (C_{FN}/C_{NF}) will be compared in next experiments (C_{NF} is usually set to 1).

B. Evaluation metrics

TABLE IV. CONFUSION MATRIX

Confusion Matrix		**Predicted Class**	
		Fraud	***Normal***
Actual Class	***Fraud***	*TP*	*FN*
	Normal	*FP*	*TN*

Confusion matrix is a visualized display tool for generally evaluating the performance of classification models, which is shown Table IV. In this study, accuracy, $\text{precision} = \frac{TP}{TP+FP}$, $\text{recall} = \frac{TP}{TP+FN}$, and F_1_score = $\frac{2 \times recall \times precision}{recall + precision}$ are used to show and compare our results with other methods.

Since the cost sensitive classification problem is dealt with in this paper, it is very necessary to take into account the misclassification cost. As well as above indicators, the expected cost of misclassification (ECM) in (9) is used to evaluate the cost sensitive classification method.

$$ECM = P_{Normal} \times FPR \times C_{NF} + P_{Fraud} \times FNR \times C_{FN} \qquad (9)$$

where P_{Normal} and P_{Fraud} indicate prior percentage of Normal publishers and fraud publishers respectively, $FNR = \frac{FN}{TP+FN}$ and $FPR = \frac{FP}{FP+TN}$.

C. Results and Discussion

1) Comparisons ABC with other optimization algorithms

The input features and parameters of BPNN are interacted, and separately optimized will affect the results. Therefore, this paper uses ABC algorithm to realize simultaneous optimization. In addition to the ABC algorithm, many other optimization algorithms, such as GA, PSO, and ACO, are also commonly used to better train model parameters. Therefore, we compare ABC with those algorithms. In order to prove the validity of the algorithm, each group of experiments is repeated 15 times and mean value is calculated. The results of comparisons ABC with other optimization algorithms are reported in Table V. In the testing set, normal publishers account for 97.27%, so classification accuracy is all very high for 4 algorithms. However, the ABC algorithm gave the best result in terms of precision, recall and F_1_score. At the same time, average number of selected features of ABC is least, which means improving the efficiency of classification.

2) Performances of different cost ratio

The BPNN classifier regarding cost-sensitivity is another advantage of the proposed method. The different cost value will be compared in this subsection to determine the best cost ratio (C_{FN}/C_{NF}). Table VI represents accuracy, precision, recall, F_1_score and ECM results according to five different cost ratios. Each group of experiments is still repeated 15 times and mean value is calculated. As can be seen from the table, when the cost of increases (higher cost ratio), recall performance increases also. Note that higher precision values indicate a better performance in terms of false classification. So, it does not mean that the higher cost ratio, the better performance of the classifier. F_1_score and ECM as global evaluation indexes, when cost ratio C_{FN}/C_{NF}=5, ECM is minimum and F_1_score is best, which means that the precision of classification is guaranteed while the recall of fraud class is improved. As fraud samples are oversampled at a certain rate, the distribution of samples has changed from extreme imbalance to relative balance. If the cost is higher, it may cause the model overfitting.

3) Comparisons with other methods

We firstly compare our proposed model with other common cost-sensitive neural networks: Cost-Sensitive Boosting Neural Networks with Threshold-Moving (CSBNN-TM), and Cost-Sensitive Boosting Neural Networks with Weight-Updating (CSBNN-WU1, CSBNN-WU2) [12]. In the process of data preprocessing, fraud samples are all oversampled at 8 times rate. The results according to different cost ratios are shown in Fig.5. Only when the cost ratio is 7, does it not give best results. In addition, to better illustrate the superiority of our detection model proposed in this paper, the representative Random

TABLE V. COMPARISON OF DIFFERENT OPTIMIZATION ALGORITHMS

Algorithms	**accuracy**	**precision**	**recall**	**F_1**	**number of selected features**
GA	0.9890	0.8520	0.8527	0.8517	15.7
PSO	0.9810	0.8448	0.8473	0.8453	17.8
ACO	0.9873	0.8369	0.8496	0.8426	16.6
ABC	0.9914	0.8827	0.9187	0.9003	13.4

TABLE VI. COMPARISON OF DIFFERENT COST RATIO (C_{FN}/C_{NF}).

Cost Ratio	**accuracy**	**precision**	**recall**	**F_1**	**ECM**
C_{FN}/C_{NF} =3	0.9899	0.8567	0.8535	0.8553	0.0129
C_{FN}/C_{NF} =4	0.9907	0.8712	0.9096	0.8918	0.0130
C_{FN}/C_{NF} =5	0.9930	0.8963	0.9239	0.9102	0.0113
C_{FN}/C_{NF} =6	0.9867	0.8528	0.9276	0.8883	0.0182
C_{FN}/C_{NF} =7	0.9834	0.8276	0.9280	0.8745	0.0233

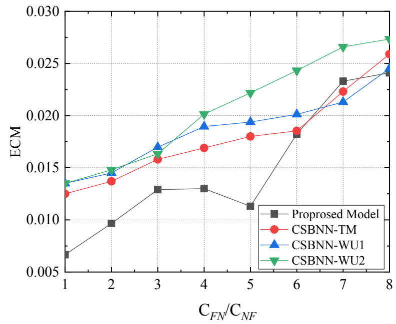

Figure 5. ECM comparison of the proposed model with other cost-sensitive neural networks

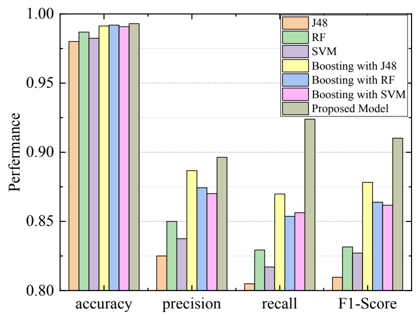

Figure 6. The performances of different models

forest (RF), J48, SVM and the ensemble method [1],[10] (Boosting with RF, Boosting with J48, Boosting with SVM) are selected as comparisons. Fig.6 shows the comparison results. Our proposed model achieves the best classification performance.

VI. Conclusion

With the rapid development of mobile advertising, click fraud is common and means of click are constantly improved. Most researches about click fraud base on ensemble machine learning methods, which actually cause model to complex and overfitting. This paper aims at solving the above problems and improve the accuracy of classification. To better capture the latent patterns of fraud clicks, BPNN is applied in the variable click fraud detection environment. To avoid local optimization of neural network and feature redundancy, we use Artificial Bee Colony Complete to complete simultaneous optimization of feature selection and weights, considering the input features and parameters of BPNN are interacted. While data imbalance and different misclassification cost exist simultaneously. We use SMOTE to deal with the imbalanced of the training click dataset and cost parameters are added to error function.

In summary, we develop a novel detection model for click fraud based on the Artificial Bee Colony and cost-sensitive neural network. Our model achieves highly competitive performance on the real world click dataset compared with the state-of-the-art methods. Future work could extend this detection model to related works or other click dataset and research deeper neural network architecture when a large number of training data sets are given.

Acknowledgment

The work was supported by the Key Research Program of Jiangsu Province, China (No.BE2015697, No.BE2017617) and the sub project of National Key Research Plan of China (No.2017YFC0805605).

References

[1] C. M. R Haider, A. Iqbal, A. H. Rahman and M. S. Rahman, "An ensemble learning based approach for impression fraud detection in mobile advertising," Journal of Network & Computer Applications, vol.112, Jun. 2018, pp. 126-141, doi: 10.1016/j.jnca.2018.02.021.

[2] MS. Iqbal, M. Zulkernine, F. Jaafar, and Y. Gu, "Protecting Internet users from becoming victimized attackers of click - fraud," Journal of Software: Evolution and Process, vol. 30, Apr. 2017, doi: 10.1002/smr.1871

[3] M. Alauthaman, N. Aslam ,Z. Li, R. Alasem, and M. A. Hossain, "A P2P Botnet detection scheme based on decision tree and adaptive multilayer neural networks," Neural Computing and Applications, vol. 29, Jun. 2018, pp. 991-1004, doi: 10.1007 / s00521-016-2564-5.

[4] D. Ye, and Z. Chen, "A new approach to minimum attribute reduction based on discrete artificial bee colony." Soft Computing, vol. 19, Jul. 2015, pp.1893-1903, doi: 10.1007 / s00500-014-1371-0.

[5] D. Karaboga, and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm." Journal of Global Optimization, vol. 39, Nov. 2007, pp. 459-471, doi: 10.1007 / s10898-007-9149-x.

[6] Y. Xiong and R. Zuo, "Effects of misclassification costs on mapping mineral prospectivity." Ore Geology Reviews, vol.82, Nov. 2016, pp. 1-9, doi: DOI: 10.1016/j.oregeorev.2016.11.014.

[7] B. Kitts, J. Y. Zhang, G. Wu, W. Brandi, J. Beasley, et al, "Click fraud detection: adversarial pattern recognition over 5 years at Microsoft," Real World Data Mining Applications, vol. 17, Nov. 2014, pp. 181-201, doi: 10.1007 / 978-3-319-07812-0_10.

[8] K.S. Perera, B. Neupane, M.A. Faisal, Z. Aung, and L.W. Wei, "A Novel Ensemble Learning-Based Approach for Click Fraud Detection in Mobile Advertising." Proc. 2013 International Conference on Mining Intelligence and Knowledge Exploration, Lecture Notes in Computer Science, Springer International Publishing, vol.8284, Jan. 2013, pp.370-382, doi: 10.1007 / 978-3-319-03844-5_38.

[9] M.A. King, A. S. Abrahams, and C. T. Ragsdale, "Ensemble learning methods for pay-per-click campaign management," Expert Systems with Applications, vol.42, Jun. 2015, pp.4818- 4829, doi: 10.1016/j.eswa.2015.01.047.

[10] R. Oentaryo, E. P. Lim, M. Finegold, D. Lo, F. Zhu, C. Phua, et al, "Detecting click fraud in online advertising: a data mining approach," Journal of Machine Learning Research, vol. 15, Jan. 2014, pp. 99-140, doi: 10.1145 / 2623330.2623718.

[11] J.A. Bullinaria and Y. Khulood, "Artificial bee colony training of neural networks," Nature Inspired Cooperative Strategies for Optimization (NICSO 2013), Springer International Publishing, vol. 512, 2014, pp. 191-201, doi:10.1007/2F978-3-319-01692-4_15

[12] J. Zheng, "Cost-sensitive boosting neural networks for software defect prediction," Expert Systems with Applications, vol. 37, Jun. 2010, pp. 4537-4543, doi: 10.1016/j.eswa.2009.12.056.