International Journal of Modern Electronics and Communication Engineering (IJMECE) ISSN: 2321-2152
Volume No.-7, Issue No.-1, January, 2019

A Predictive System for Forecasting of Bankruptcy using Decision Tree – Ant Colony Optimization

Jayanthi J
Parul Institute of Engineering and Technology
Parul University
Vadadora, India
E-mail: jjay.mtech@gmail.com

K. Suresh Joseph
Department of Computer Science
Pondicherry University
Pondicherry, India
E-mail: ksjoseph.c@gmail.com

Abstract: Forecasting bankruptcy is an important and challenge task for both academic researchers and business practitioners. The problem has been tackled using various models statistical and intelligent technique in the past. This with our proposed approach. This hybrid approach (DTACO) will capable of achieve an improve predictive accuracy and providing guidance for decision makers to detect and prevent potential financial crisis in the early stages.

Keywords: Bankruptcy prediction, Decision Tree, Ant Colony Optimization and Decision Tree Ant Colony Optimization.

I. INTRODUCTION

Bankruptcy prediction is an effective tool to help financial institutions and relevant people to make a decision on business performance of companies. The prediction of bankruptcy for financial firms has been extensively researched area since 1960s. The tool of Bankruptcy provides creditors, auditors, stakeholders and senior managers a chance to identify the problems early. Thus, relevant people have an opportunity to intervene early into affairs problems to reduce the expected cost of business failure. The problem is stated as follows: given a set parameters that describe the situation of a company over a given period, predict the probability that the company may be bankrupted during the following year. Common reasons of bankruptcy include lack of financial knowledge, weakness in debts management, incapability in coordination of interest plans, lack of having enough liquidity to face the unpredicted events and lack of using proper chances of investment in financial markets.

This paper provides a performance comparison of decision tree methods for bankruptcy prediction. The paper is organized as the follows. In section 2, describes the relevant background knowledge on bankruptcy prediction and related work to easily understand the analysis conducted in our experiments; in section 3. Section 5 explores the methodologies used in this work and further discussions there on. Section 4, describes the DTACO work.

II. BACKGROUND OF BANKRUPTCY PREDICTION

The problem is stated as follows: given a set of parameters that describe the situation of a company over a given period, predict the probability that the company may become bankrupted during the following year. There exists numerous algorithm produced to construct classification model for bankruptcy prediction **[2]** such as statistical techniques , Case Based Reasoning, Neural Networks, Operational Research, Rough set , Evolutionary technique, Fuzzy logic, isotonic separation, wavelet, decision tree and hybridization techniques. Among statistical techniques, the method covered are: Linear Discriminant Analysis (LDA), Multivariate Discriminant Analysis (MDA), Quadratic Discriminant Analysis (QDA), logistic regression (logit), linear probability models(probit) , Principal Component Analysis (PCA), Independent Component analysis (ICA), Z score, Zeta model and Factor Analysis (FA). The intelligent technique covered in the study belongs to different neural network (NN) architecture including Multi Layer Perception (MLP), Radial Basis Function network (RBFN), and Cascade Correlation Neural network (CASCOR). Operational research techniques are including linear programming (LP), Data Envelopment Analysis (DEA) and Quadratic Programming (QP).

III. LEARNING MODELS WITH DECISION TREE AND SWARM INTELLIGENCE

A Data mining (Knowledge Discovery in Databases (KDD)) **[7]** is the process of discovering meaningful patterns in huge databases. In addition, it is also an application that can provide significant competitive advantages for making right decision. It helps to predict future trends and behavior allowing business to make proactive, knowledge driven decision. Data mining involves an integration of technique from multiple disciplines such as database technique, statistics, machine learning, high performance computing etc.

A. Decision Tree

Decision tree **[1,4]** is one of common data mining methodologies that provide the both classification and predictive functions simultaneously. In classification try to predict target attribute by means of some of the other available attributes. A decision tree is a tree whose internal nodes can be taken as tests and whose leaf nodes can be taken as categories. These tests are filtered down through the tree to get the right output to the input pattern. The advantage of using decision trees here was that it not requires any statistical knowledge.

A decision tree is the most widely used tool for decision making. To accomplish this one should draw a decision tree with different branches and leaves. These branches and leaves should point to all the various factors concerning a particular situation. A decision tree is almost like a decision support tool. It uses a tree- like graph of decisions and their possible outcomes which include resource costs, event outcomes, and utility. It is one way to display an algorithm. Depending on the situation and desired outcome there are various types of decision trees methods that we can use (Decision Stump, C4.5, Random Forest and CART).

B. Decision Stump

Decision stump **[9]** is a decision tree with one internal node (root) which is immediately connected to the terminal nodes. It makes a prediction based on the value of just a single input feature. Another name of decision stump is 1-rules. Each node in decision stump represents a feature in an instance to be forecasted, and each branch represents a value that node can take. Instances are classified starting at the root node and sorting them based on their feature values.

C. C4.5

C4.5 is depth–first constructions of the decision tree with DC (divide and conquer) method. Its performance of runtime is sacrificed for the consideration of the limited memory at run time. In this algorithm, each node in a tree is associated with a set of assigned weights to take into account unknown attribute values. It can deal both continuous and discrete types of attribute values. If the data is continuous, it must be discrete first.

Missing values **[5]** are a widespread problem in data analysis. Many real world data are incomplete as some instances may have missing attribute-values. Attribute values can be missing for various reasons. Imputation method and case deletion are used to handle missing data. In imputation method, missing values are replaced with estimates derived from applying statistical methods to the available data. In the case of deletion method, the deletion of all the instances with missing values can lead to the loss of useful information, thus it introduces some bias in the data. A way to rectify the problem of missing data is to employ a sound method of imputation, which replaces missing values with reasonable estimates.

C4.5 handles the missing values as i) Filing a missing attribute value with most common occurring value mode if its type nominal and mean if its type is numerical. ii) Assigned a special label M for those missing nominal attributes and treat M as if it is another attribute value. This often performed poorly as compared with filling up with mean or mode.

D .Random Forests

Random Forests **[8]** is an extension of decision tree. The Random Forests is consists of many decision trees and outputs of the class that is the mode of the class's output by individual tree. The Random Forests is work based on the randomly selecting a limited number of features from all available features for node splitting, and each tree cast a vote for final prediction.

E. CART

CART **[10]** is a decision tree method that partitions a set of samples into groups and it offered for handling missing value. It analysis has a specified outcome variable and is based on the sense of reducing impurity to build tree. CART alike to C4.5 techniques but Gini index used as divide criteria.

F. Swarm Intelligence

Swarm intelligence is a branch of evolutionary computation, which is the application of methods inspired by the natural world to hard problems in artificial intelligence. For example, the collective foraging behavior of ants has inspired several computational models – ant-based algorithms or ant colony optimization algorithms (ACO).

G. Ant Colony Optimization

The ant colony optimization **[3]** technique has emerged recently as a novel meta-heuristic belongs to the class of problem-solving strategies derived from natural. The ant system optimization algorithm is basically a multi-agent system where low level interactions between single agents result in a complex behavior of the whole ant colony. Ant system optimization algorithms have been inspired by colonies of real ants, which deposit a chemical substance (called pheromone) on the ground. It was found that the medium used to communicate information among individuals regarding paths, and used to decide where to go, consists of pheromone trails. A moving ant lays some pheromone (in varying quantities) on the ground, thus making the path by a trail of this substance. While an isolated ant moves essentially at random, an ant encountering a previously laid trail can detect it and decide with high probability to follow it, thus reinforcing the trail with its own pheromone.

The collective behavior where that emerges is a form of autocatalytic behavior where the more the ants following a trail, the more attractive that trail becomes for being followed. The process is thus characterized by a positive feedback loop, where the probability with which an ant chooses a path increases with the number of ants that previously chose the same path. It is follow the basic steps as initialization, local heuristic, probability calculation (transition rule& constraint satisfaction, fitness function), global update for pheromone trails and terminal node. It has some of the parameter such as population size, stagnation limit, generalization limit, alpha, and beta.

IV. Decision Tree Ant Colony Optimization System Model

Feature selection (FS) is the technique of selecting a subset of relevant features for building learning models. FS provides better understanding of the data by selecting important features within the data. However, except for datasets with only a very small set of features. In this study, a FS is proposed which combines Ant Colony Optimization (ACO) with C4.5 decision tree builder. An ACO is setup for a given dataset and each ant probability selects features on the basis of pheromone and heuristic values associated with each link. When an ant completes its tour then for evaluating fitness of the sub-set of features selected by it , we use C4.5 algorithm for constructing a rule set based only on the features in the sub-set and the evaluate the accuracy of the rule set which is considered the fitness of the solution found by the ant.

In DTACO each ant chooses the appropriate attribute for splitting in each node of the constructed decision tree according to the heuristic function and pheromone values. The heuristic function is based on the entropy criterion, which helps ants divide the objects into two groups, connected with the analyzed attribute values. In this way, the attribute, which well separate the objects is treated as the best condition for the analyzed node. The best splitting is observed when we classified the same number of objects in the left and right sub trees with the maximum homogeneity in the decision classes. Pheromone values represent the best way (connection) from the superior to the subordinate nodes – all possible combinations in the analyzed sub trees. For each node we calculate the following values according to the objects classified using the entropy criterion of the superior node.

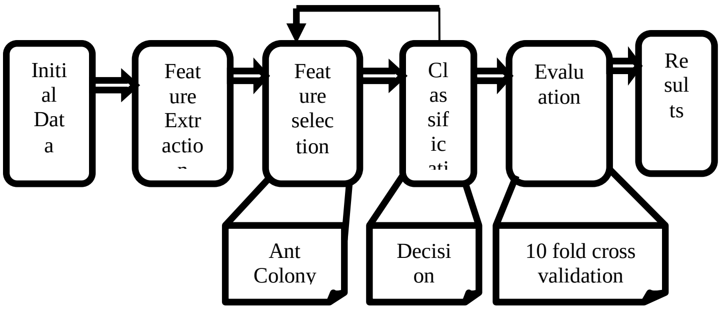

Fig-2: Bankruptcy prediction – Decision Tree Ant Colony Optimization

Algorithm:

1. Load the dataset.
2. Calculate information gain (heuristic value) of each attribute.
3. Generate a population of Ants.
4. Initialize the parameters of ACO.
5. FOR each Ant, Generate a subset S.
6. Evaluate each feature sub set S.
7. IF the fitness or accuracy is better than previous global best.
8. Set the current subset S accuracy as global best accuracy.
9. END IF
10. Update the pheromone values.
11. Repeat this process until stopping criteria do not meet.
12. END FOR
13. Report best feature subset as final more appropriate set.
14. END

V. Experimental Setup and Discussion

A. *Data set*

The data set used in this research is the bankruptcy data set (http://www.pietruszkiewicz.com) in literature. It includes 120 companies from a period of two consecutive years. Among the companies, 56 companies were bankrupted 2-5 years later. Table 1 illustrates each company is described by 30 attributes as below:

Table-1: Attributes

X1	Cash/current liabilities
X2	Cash/ total assets
X3	Current assets/ current liabilities
X4	Current assets/ total assets
X5	Working capital/ total assets
X6	Working capital/ sales
X7	Sales/ inventory
X8	Sales/ receivables
X9	Net profit/ total assets
X10	Net profit/ current assets
X11	Net profit/ sales
X12	Cross profit/ sales
X13	Net profit/ liabilities
X14	Net profit/ equity
X15	Net profit/ (equity + long term liabilities)
X16	Sales/ receivables
X17	Sales/ total assets
X18	Sales/ current assets
X19	(365* receivables)/sales
X20	Sales/total assets
X21	Liabilities / total income
X22	Current liabilities/ total income

X23	Receivable/ liabilities
X24	Net profit/sales
X25	Liabilities/ total assets
X26	Liabilities/ equity
X27	Long term liabilities/ equity
X28	Current liabilities/ equity
X29	EBIT(earnings before interests and taxes)/ total assets
X30	Current assets/ sales

B. *Evaluation Metrics*

Performance metrics were evaluated based on the classification confusion matrix. Here TP, TN, FP, FN represent the usual notation for the matrix in terms of true and positive results from the classifier. Recall and precision measures are good indicators of the classifier performance. Type I error indicates the misclassification of a healthy firm as distressed and Type II error indicates the misclassification of distressed as healthy one. Accuracy refers to the total correct classification for the set regardless of type. F1 score quantifies the tradeoff between recall and precision and indicative of the performance of the overall algorithm.

i) Recall R= (TP/(TP+FN))
ii) Precision P = (TP/ (TP+FP))
iii) Type I error (false positive) = FP/FP+TN
iv) Type II error (false negative) =FN/FN+TP
v)Accuracy= (TP+TN)/ (TP+FN+FP+TN)
vi) F-score F1=2(R*P/R+P)

1) True positive (TP) = the number of predicted positive cases that are actually positive.
2) True negative (TN) = the number of predicted negative cases that are actually negative
3) False positive (FP)= the number of predicted positive cases that are actually negative
4) False negative (FN) = the number of predicted negative cases that are actually positive.

Table-2: Confusion Matrix

Confusion matrix - Predicted class			
Actual class		C1	C2
	C1	TP	FN
	C2	FP	TN

C. *Experimental Design*

All experiments described in this paper were performed using libraries from Weka 3.7.4 **[6]** machine learning environment. A lot of studies used in Weka in classification task, for examples. Fifteen selected decision tree classifiers are used to build the classification models; this classifier was briefly described above.

Each classification method was used as it is in Weka environment which means that no additional parameter tuning was performed before or during classification performance comparison. As well as we evaluate AUC of the classification methods using, each test we used 10 fold cross validation. We summarize our machine learning work for bankruptcy prediction in three main stages. First stage is attribute selection, second is choosing appropriate predictor, and third is produced model evaluation as shown in Fig.1.

VI. RESULT AND ANALYSIS

We report the results from four methods that used to study the usefulness of the decision tree to predict bankruptcy. For each method we evaluated classification accuracy on the original dataset. We notice that CART and Random Forest gave the higher accuracy on original data set (83.3%). We observe that F1 score also higher and Error Type I & II are better than other methods. Table 3 summarized the accuracy of these methods. An overall view of the binary classifier performance is observed.

Table-3: Classification Result

Class	Metrics					
	F1-score	Type-I	Type-II	Recall	Precision	Accuracy
Decision Stump	72.7%	32.0%	22.3%	77.6%	67.9%	72.5%
J48(C4.5)	80.9%	14.8%	20.5%	79.4%	82.4%	82.5%
Random forest	82.2%	16.4%	16.9%	83.0%	81.5%	**83.5%**
CART	81.9%	14.8%	18.7%	81.2%	82.7%	**83.3%**

VII. CONCLUSION

This experimental study compares classification performance of decision tree methods via using bankruptcy dataset. The algorithms taken into consideration are Decision Stump, C4.5, Random Tree, Random Forest and CART. From which we obtained experimental results conclude CART and Random Forest are efficient rather than other decision tree methods. And C4.5 handled missing values in efficient manner. DTACO has been proved theoretically and must be implemented for bankruptcy prediction

REFERENCES

[1] S.Malathi, Dr. S. Sridhar, "A Classical Fuzzy Approach for Software Effort Estimation on Machine Learning", IJCSI

International Journal of Computer Science Issues, November 2011.
[2] Petronio L. Braga, Adriano L.I Oliveria, Gustave H.T Ribeiro, L.Meira, "Bagging predictors for estimation of software project effort", IEEE conf, August 2007.
[3] Andreas S. Andreou, Efi Papatheocharous, "Software cost estimation using fuzzy decision tree". IEEE, 2008.
[4] Tarun kumar sharama , Millie pant, " Halton based initial distribution in artificial bee colony application in software effort estimation", IEEE conf, 2011.
[5] Adirano L.I, Oliveira, "Estimation of software effort with support vector regression", Elsevier, 2006.
[6] Syona Gupta, Geeta sikka, Harsh verma, "Recent methods for software effort estimation by anology", ACM, July 2011.
[7] Bilge baskekes, Burak Turhan, Ayse Bener, "software effort estimation using machine learning mehods", IEEE, 2007.
[8] Adirano L.I, Oliveira, Petronio L. Braga, Ricardo m.F, Lima, Marcio L, Cornellio, "GA based method for feature selection and parameters optimization for machine learning regression applied to software effort estimation", Elsevier, 2010.
[9] Ruchika Malhotra, Ankita jain, "software effort prediction using statistical and machine learning methods", International journal of advanced computer science and applications (IJACSA), January, 2011.
[10] M.Jorgensen, "A review of studies on expert estimation of software development effort", Journal of systems and software, 2004.
[11] Spareref.com, "Nasa to shut down checkout & launch control system", August, http://bit.ly/eiYxlf.
[12] Mrinal kanti ghose, Robeet haatnager, Vandama bhattacharjee,"Comparing some neural network models for software development effort prediction", IEEE, 2011.
[13] B.Boehm, "software engineering economics", New Jersey: prentice hall, 1981.
[14] B.Boehm, R.Madachy, and B.Steece, "software cost estimation with cocomo II", New York, Prentice hall, 2000.
[15] Arundhati Navada, Aamir Nizam, Ansari Siddharth Patil, Balwant A. sonakamble, " overview of use of decision tree algorithms in machine learning", IEEE, 2011.
[16] B.N. Lakshmi. , G.H. Raghunandhan, "A conceptual overview of data mining", proceeding of the national conference on innovations in emerging technology, 2011.
[17] Nikolaos Mallios, Elpiniki papageorgion, Michael Samarians, "Comparison of machine learning technique using the WEKA environment for prostate cancer therapy plan', IEEE trans, 2011.
[18] John. G. Cleary, Leonard E, Trigg," K* An instance based learner using entropic distance measure"
[19] Chengjun Zhan, Albert gan, Mohammed hadi, "Prediction of lane clearance time of freeway incidents using M5P tree algorihm", IEEE, December 2011.
[20] Taghi m.Khoshgoftaar, Andres Folleco, Jason van hulse, Lofton bllard, "software quality imputation in the presence of noisy data", IEEE, 2006.
[21] Guo peng yang, Xin zhou, Xuchu yu, "Hyperspectral imagery classification based on gentle adaboost and decision stump", IEEE, 2009.
[22] http://promise.site.uottawa.ca/SERepository
[23] Zexuan Zhu, Yew-Soon Ong, Mnoranjan Dash ," Wrapper-Filter Feature Selection Algorithm using a Memetic Framework", IEEE , 2007.
[24] Zexuan Zhu, Sen Jia, Zhen Ji, " Towards a Memetic Feature Selection Paradigm", IEEE, 2010.

AUTHOR'S BIOGRAPHIES

[1] Jayanthi J
Assistant Professor
Parul Institute of Engineering and Technology
Parul University
Vadadora, India
E-mail: jjay.mtech@gmail.com

[2]Dr. K. Suresh Joseph
Assistant Professor
Department of Computer Science
Pondicherry University
Pondicherry, India
E-mail: ksjoseph.csc@gmail.com