Prediction of Stock Market Index Using Genetic Algorithm

Article

4 authors, including:

Lakshman Naik Ramavathu
KU College of Engineering and Technology, Kakatiya University
22 PUBLICATIONS **81** CITATIONS

SEE PROFILE

Dr. Manjula Bairam
Kakatiya University
20 PUBLICATIONS **25** CITATIONS

SEE PROFILE

Dr A Govardhan
Jawaharlal Nehru Technological University, Hyderabad
293 PUBLICATIONS **1,355** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Multi-Script OCR Systems View project

Grid computing View project

Prediction of Stock Market Index Using Genetic Algorithm

R. Lakshman Naik[1*], D. Ramesh[1], B. Manjula[1], Dr. A. Govardhan[2]

1. Department of Informatics, Kakatiya University, Warangal-506 009, Andhra Pradesh, India

2. Department of CSE, JNT University (H), Hyderabad-500 085, Andhra Pradesh, India

    * E-mail of the corresponding author: lakshman.ramavathu@gmail.com

Abstract

The generation of profitable trading rules for stock market investments is a difficult task but admired problem. First stage is classifying the prone direction of the price for BSE index (India cements stock price index (ICSPI)) futures with several technical indicators using artificial intelligence techniques. And second stage is mining the trading rules to determined conflict among the outputs of the first stage using the evolve learning. We have found trading rule which would have yield the highest return over a certain time period using historical data. These groundwork results suggest that genetic algorithms are promising model yields highest profit than other comparable models and buy-and-sell strategy. Experimental results of buying and selling of trading rules were outstanding.

Key words: Data mining, Trading rule, Genetic algorithm, ANN, ICSPI prediction

1. Introduction

For a long time, stock market prediction is long esteemed desire of investors, speculators, and industries. Although several studies investigated to predict price movements in stock market, financial time series too complex and noisy to forecast. Many researchers predicted the price movements in stock market using artificial intelligence (AI) techniques during past decades

Recent research tends to include novel factors and to hybridize several AI techniques. In (Y. Hiemstra, 1995) proposed fuzzy expert systems to predict stock market returns. He suggested that ANNs and fuzzy logic could capture the complexities of the functional mapping because they do not require the functional specification of the function to approximate. A more recent study of (K. Kohara, 1997) incorporated prior knowledge to improve the performance of stock market prediction. In (R. Tsaih, 1998) integrated the rule-based technique and the ANNs to predict the direction of the S&P 500 stock index futures on a daily basis.

Previous research using AI techniques almost predicted the price of every trading day, week, and month. It is more important, however, to determine stock market timing, when to buy and sell stocks, than to predict the price movement for everyday because investors in stock market generally do not trade everyday. If investors trade their stocks everyday, they are charged to tremendous amount of fee for trade. Market timing is an investment strategy which is used for the purpose of obtaining the excess return. Traditionally excess return is achieved by switching between asset classes in anticipation of major turning points in stock market (G. Waksman, 1997).

While rule-based technologies improved dramatically, many of stock market applications were less than successful.

For this reason, the trend toward automatic learning systems is particularly evident in the financial services sector. Advances in chaos theory provide the theoretical justification for constructing nonlinear models, which is typically the goal in machine learning. Previous studies on this issue suggest that artificial intelligence techniques such as artificial neural networks (ANN) have more frequent chances to detect nonlinear patterns in stock market (H. Ahmadi, 1990; K.Kamijo, 1990; T. Kimoto, 1990). ANN, however, has a drawback that the users of the model can not readily comprehend the final rules.

We propose data mining approach using genetic algorithms (GA) to solve the knowledge acquisition problems that are inherent in constructing and maintaining rule-based applications for stock market. Although there are an infinite number of possible rules by which we could trade, but only a few of them would have made us a profit if we had been following them. This study intends to find good sets of rules which would have made the most money over a certain historical period. This paper tends to mine reasonable trading rules using genetic algorithms for India Cements stock price index (ICSPI) future. We have found trading rule which would have yield the highest return over a certain time period using historical data. Experimental results of buying and selling of trading rules were outstanding.

2. Related Work

In refer. (T. Kimoto, 1990) Used several learning algorithm and prediction method for the Tokyo stock exchange prices index (TOPIX) prediction system. This system used modular neural network that learned the relationships between various factors. The output of this system was the best timing for when to buy and sell stocks. They executed simulation of buy and sell stocks to evaluate the effect of system. In this study, vector curve, turn-over ratio, foreign exchange rate and interest rate were used as input variables. Trading profit using this system revealed more than that of "Buy and hold strategy".

In refer. (K.Kamijo, 1990) Classified the changing pattern of TOPIX to triangle pattern by use of candlestick chart. They have learned these patterns using the recurrent neural network. The test set of triangle was accurately classified in 15 out of 16 experiments. (H. Ahmadi, 1990) Tried to test the 'Arbitrage pricing theory (APT)' by ANN. This study used backpropagation neural network with generalized delta rule to learn relationship between the return of individual stocks and market factors.

In refer. (R.R. Trippi, 1992) Executed daily prediction of up and down direction of S&P 500 Index Futures using ANN. Generating a composite recommendation for the current day's position. Input variables in this study were technical variables for the two-week period to the trading day, open, high, low, close price, open price and the price fifteen minutes after the market opening of the current trading day. The output variable was long or short recommendation. They performed composite rule generation procedure to generate rules for combining outputs of networks. They reported prediction accuracy was 45.3% - 52.8%.

In refer. (L. S. Duke 1993) also executed daily prediction of German Government Bond Futures using feed-forward backpropagation neural network. In this study, they used opening range (obtained from the highest and lowest bids at the open), highest, lowest price, closing price, volume of traders, open Interest, industrial production, consumer prices, current account balance, unemployment rates, short and long term interest rates, wholesale price index, M3 combined supply and benchmark bond yield as input variables. The following day's closing price was output. A result

of the network's predictions as compared to the actual movement was 53.94%.

In refer. (David de la Fuent, 2006) have presented a work which also attempts to optimize the timing of an automated trader. They use GA to develop trading rules for short time periods, using Technical Indexes, such as RSI, as GA's chromosome. They proposed the use of the developed rules on stocks of Spanish company. However, the description in that work was too preliminary to allow for a comparison with our system to be made.

And also, in refer. (Cyril Schoreels, 2004) have presented a work which attempts to generate the buying and selling signals against 30 companies' stocks in Germany (DAX30). They use combination of Technical Indexes applied to GA as well as (David de la Fuent, 2006), and then rank the stocks according with the strength of signals to restructure the portfolio. However, they don't devise any criterion of profit cashing and loss cutting.

3.   Genetic Algorithm (GA): Proposed Model

Application of GA in the context of data mining is generally for the task of hypothesis testing and refinement, where the user poses some hypothesis and the system first evaluate the hypothesis and seek to refine it. Hypothesis refinement, where the user poses some hypothesis and the system first evaluates the hypothesis and then seeks to refine it.   Hypothesis refinement is achieved by "seeding" the system with the hypothesis and then allowing some or all parts of it to vary.   One can use a variety of evaluation functions to determine the fitness of a candidate refinement.   The important aspect of the GA application is the encoding of the hypothesis and evaluation function for fitness. Another way to use data mining is to design a hybrid techniques by blending one of the know techniques with GA (Arun K. Punjari, 2001).

GA is search algorithm based on the mechanics of natural selection and genetics and they combine survival of the fittest among string structures to form a search algorithm (L. Davis, 1991). GA is particularly suitable for multi-parameter optimization problems with an objective function subject to numerous hard and soft constraints. The main idea of GA is to start with a population of solutions to a problem, and attempt to produce new generations of solutions which are better than the previous ones. GA operates through a simple cycle consisting of the following four stages: initialization, selection, crossover, and mutation. Figure 1 shows the basic steps of proposed genetic algorithms model.

In the initialization stage, a population of genetic structures (called chromosomes) that are randomly distributed in the solution space is selected as the starting point of the search. These chromosomes can be encoded using a variety of schemes including binary strings, real numbers or rules. After the initialization stage, each chromosome is evaluated using a user-defined fitness function. The goal of the fitness function is to numerically encode the performance of the chromosome. For real- world applications of optimization methods such as GA, the choice of the fitness function is the most critical step.
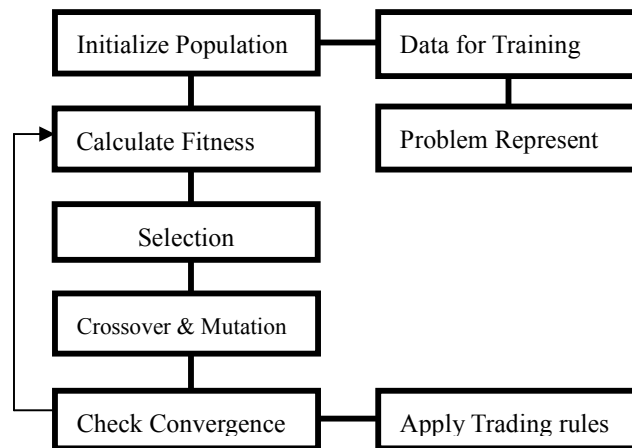
Fig.1. Basic steps of proposed model

The mating convention for reproduction is such that only the high scoring members will preserve and propagate their worthy characteristics from generations to generation and thereby help in continuing the search for an optimal solution. The chromosomes with high performance may be chosen for replication several times whereas poor-performing structures may not be chosen at all. Such a selective process causes the best-performing chromosomes in the population to occupy an increasingly larger proportion of the population over time.

Crossover causes to form a new offspring between two randomly selected 'good parents'. Crossover operates by swapping corresponding segments of a string representation of the parents and extends the search for new solution in far-reaching direction. The crossover occurs only with some probability (the crossover rate). There are many different types of crossover that can be performed: the one-point, the two-point, and the uniform type.

Mutation is a GA mechanism where we randomly choose a member of the population and change one randomly chosen bit in its bit string representation. Although the reproduction and crossover produce many new strings, they do not introduce any new information into the population at the bit level. If the mutant member is feasible, it replaces the member which was mutated in the population. The presence of mutation ensures that the probability of reaching any point in the search space is never zero.

4. Trading Rule

Although there are an infinite number of possible rules by which we could trade, it seems that only a few of them would have made a profit. To find the rule that would have yielded the most profit had it been used to trade stocks on a given set of historical data, firstly, we develop trading rules of this general form is as shown below.

4.1. General form of trading rules

If today's value of the indicator 1 is greater than or equal to (less than) A1,

And change since the last day's value of the indicator 2 is Greater than or equal to (less than) A2,

165

And last day's value of the indicator 3 is greater than or equal to (less than) A3,

And last day's value of the indicator 4 is greater than or equal to (less than) A4,

And last day's value of the indicator 5 is greater than or equal to (less than) A5,

And today's value of the indicator 6 is greater than or equal to (less than) A6,

Then buy, else sell

There are five conditions that are evaluated for each trading day. If the all of five conditions are satisfied, then the model will produce 'buy' signal on that day, otherwise it will suggest 'sell.' A_1 to A6 denotes the cutoff values. The cutoff values range from 0 to 1, and represent the percentage of the data source's range. For example, if RSI (relative strength index) ranges from 0 to 100, then a cutoff value of 0.0 would match a RSI of 0, a cutoff value of 1.0 would match a RSI of 100, and a cutoff value of 0.5 would match a RSI of 50. This allows the rules to refer to any data source, regardless of the values it takes on. We consider additional flexibility regarding the indicator component of the rule structure such as 'today's value,' 'last day's value,' and 'change since the last day's value.' Translating this in its full form, for example, would yield the following statement:

4.2. Trading rules

If today's value of ROC is greater than or equal to 30.0,

And change since the last day's value of RSI is less than 60.0,

And last day's value of stochastic %D is less than 51.0,

And last day's value of A/D oscillator is less than 12.5,

And last day's value of MACD is less than 14.9,

And today's value of stochastic %K is less than 75.9,

Then buy else sell

Above rule structure is summarized in Table 1. In Table 1, 'which data' means data source the rule refers to, and 'modifier' means a modifier value that determines if the value itself should be examined, or if the last day's value or the change since the last day should be examined. There has been much debate regarding the development of trading system using historical data. We agree that the future is never exactly like the past; however, a common investment approach is to employ systems that would probably have worked well in the past and that seem to have a reasonable chance of doing well in the future. So, we define a goal of the system as finding a rule which would have yielded the highest return over a certain time period.

In setting up the genetic optimization problem, we need the parameters that have to be coded for the problem and an objective or fitness function to evaluate the performance of each string. The parameters that are coded are the cell values of Table 1. The varying parameters generate a number of combinations of our general rules. The task of defining a fitness function is always application specific. In this case, the objective of the system is to find a trading rule which would have yielded the highest return over a certain time period. We apply the trading profit to the fitness function for this study.

Table 1: Trading rules

Rule number	Which data	Less than/ greater than or equal to	Cutoff value	Modifier
1	INT_{i1}	1 or 2	A1	1,2 or 3
2	INT_{i2}	1 or 2	A2	1,2 or 3
3	INT_{i3}	1 or 2	A3	1,2 or 3
4	INT_{i4}	1 or 2	A4	1,2 or 3
5	INT_{i5}	1 or 2	A5	1,2 or 3
6	INT_{i6}	1 or 2	A6	1,2 or 3

Description: INT_{ij} (i=1,2,..n, j=condition number)

1=Less than/ 2= greater than or equal to

Cutoff Aj (j=condition number)

1=today's value, 2=last day's, 3=change since the last day

4.3. Data and Variables

The research data used in this study is India cements stock price index (ICSPI) from September, 2011 through April, 2012. Futures are the standard forms that decide the quantity and price in the certified market (trading place) at certain future point of time (delivery date). General functions of futures market are supplying information about future price of commodities, function of speculation and hedging. Being different from the spot market, futures market does not have continuity of price data. That is because futures market has price data by contract. So, in futures market analysis, nearest contract data method is mainly used and incorporated in this research. We collected a sample of 160 trading days. Data has been collected from Yahoo finance and Money control web site.

5. Experimental Result

To find the profitable trading rules, in this paper GA model was proposed in the previous section. We use 100 chromosomes in the population for this study. The crossover and mutation rates are changed to prevent the output from falling into the local optima. The crossover rate ranges 0.5 - 0.7 and the mutation rate ranges 0.05 - 0.12 for this experiment. We extract six trading rules by genetic search process. The derived rules are alternatively good trading rules although there is minor difference in simulated performance. Each of rules consists of conditions referring input factors. These processes and simulation are done by the genetic algorithms software package (evolver). Trading profit earned from simulation results are summarized in Table 2

Table 2: Summarized Results

Trading rules	Accumulated amount (Assume initial investment of 10,000)			
	Training (Sep 2011 –Dec 2011)	Profit (%)	Validation (Jan 2012 – April 2012)	Profit (%)
Buy-Sell	3856	-61.44	7364	-26.36
Rule 1	12178	21.17	13740	37.4
Rule 2	12716	27.16	12357	23.57
Rule 3	12730	27.3	12100	21.00
Rule 4	12176	21.76	14893	48.93
Rule 5	12273	22.73	14753	47.53
Rule 6	13042	30.42	15437	54.37

While the underlying index decreased more than 61% during the training period, we could find rules that would have yielded the high level of profit had it been used to trade stocks on a given set of historical data. The rules derived by learning training data using GA are applied validation (holdout) samples to verify the effectiveness of the proposed approach. While the underlying index decreased about 26% during the validation period, the trading strategies followed by the derived rules earn 21% to 54% of trading profit during the period. These preliminary results shows that GA is promising model yields highest profit than other comparable models and buy-and-sell strategy. However, trading rules generated by GA produce predictions only when the rules are fired. Each of the rules extracted produces predictions less than 18% of time

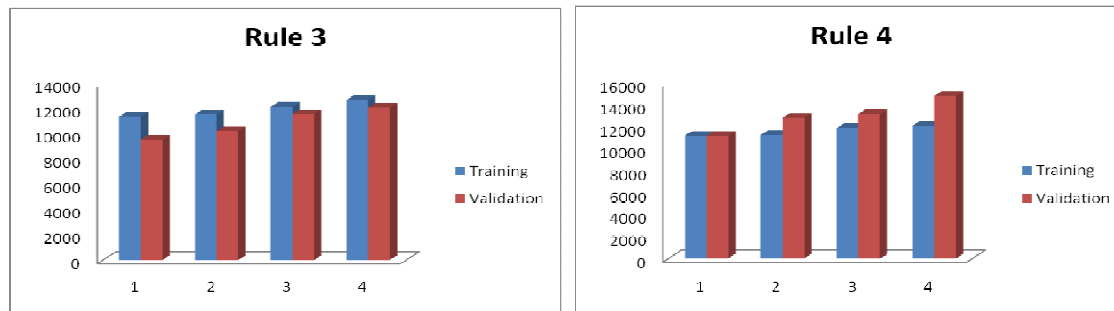## 5.1. *Performance of Simulation Results*

While the underlying index increased about 37.4% during the validation period, the trading strategies followed by the derived rule 1 earn 21.17 % to 37.4% of trading profit during the period. The rule 1 extract produces predictions less than 12% of time
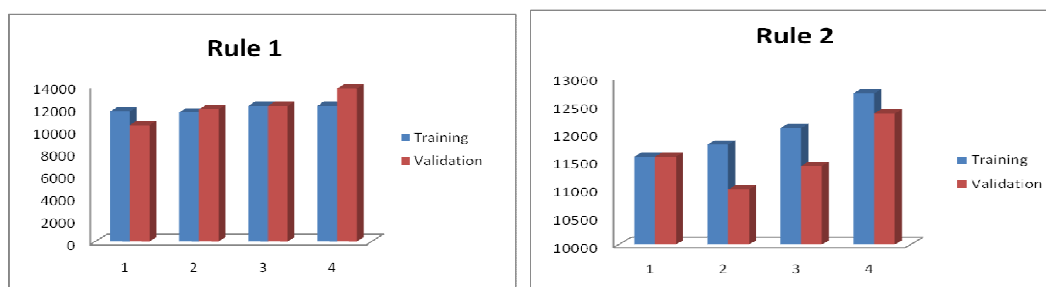
While the underlying index decreased about 23.57% during the validation period, the trading strategies followed by the derived rule 2 earn 27.16% to 23.57 %   of trading profit during the period. The rule 2 extract produces predictions less than 15% of time
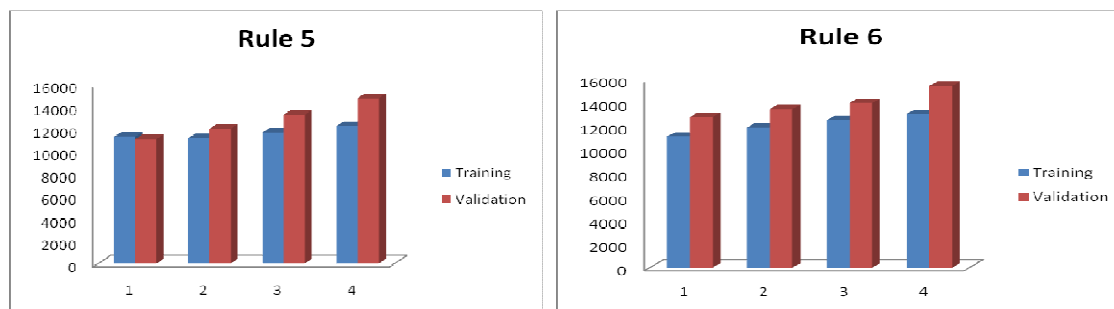
While the underlying index decreased about 21% during the validation period, the trading strategies followed by the

derived rule 3 earn 27.3% to 21.0 %   of trading profit during the period. The rule 3 extract produces predictions less than 15% of time

While the underlying index increased about 48.93% during the validation period, the trading strategies followed by the derived rule 4 earn 21.76 % to 48.93% of trading profit during the period. The rule 4 extract produces predictions less than 18% of time

While the underlying index increased about 47.53% during the validation period, the trading strategies followed by the derived rule 5 earn 22.73 % to 47.53% of trading profit during the period. The rule 5 extract produces predictions less than 17% of time

While the underlying index increased about 54.37% during the validation period, the trading strategies followed by the derived rule 6 earn 30.42 % to 54.37% of trading profit during the period. The rule 6 extract produces predictions less than 18% of time

While the underlying index decreased more than 61% during the training period, we could find rules that would have yielded the high level of profit had it been used to trade stocks on a given set of historical data. The rules derived by learning data using GA are applied validation (holdout) samples to verify the effectiveness of the proposed approach. While the underlying index decreased about 26% during the validation period,

6. Conclusion

In this paper we mine reasonable trading rules using GA for India Cements Stock Price Index (ICSPI). We have found six alternative good rules which would have yielded the high return over a certain time period. Simulated results of buying and selling of trading rules were outstanding. These preliminary results suggest that GA is promising methods for extracting profitable trading rules. Although the trading systems that have worked well in the past seem to have a reasonable chance of doing well in the future, we need a more extensive validation process.

We need a more extensive validation process because the future is never exactly as the past. We are working towards verifying and enhancing the trading rules using current data.

Reference

Arun K Punjari "Data mining techniques" universities press (India) Pvt. Ltd. 2001

Ahmadi, H., "Testability of the arbitrage pricing theory by neural networks," *Proceedings of the IEEE International Conference on Neural Networks*, 1990, pp.1385-1393.

L. Davis, *Handbook of genetic algorithms,* Van Nostrand Reinhold, NY, 1991.

Y. Hiemstra, "Modeling structured nonlinear knowledge to predict stock market returns", In Trippi, R. R. (Eds.), *Chaos & Nonlinear* Dynamics *in the financial Markets: Theory, Evidence and Applications*, Irwin, 1995, pp. 163-175.

K. Kohara, Ishikawa T., Fukuhara Y. and Nakamura Y., "Stock price prediction using prior knowledge and neural *networks*", *International Journal of Intelligent Systems in Accounting, Finance and Management*, vol. 6, 1997, pp. 11-22.

K. Kamijo and Tanigawa T., "Stock price pattern recognition: A recurrent neural network approach'" *Proceedings of the IEEE International Joint Conference on Neural Networks*, 1990, pp. 1215-1221.

T. Kimoto, Asakawa K., Yoda M. and Takeoka M., "Stock market prediction system with modular neural networks," *Proceedings of the IEEE International Joint Conference on Neural Networks*, 1990, pp. 11-16.

R. Tsaih, Hsu, Y. and Lai C. C., " Forecasting S&P 500 Stock index futures with a hybrid AI system", *Decision support Systems*, 1998, pp. 161-174.

G.. Waksman, Sandler M., Ward M. and Firer C., "Market timing on the Johannesburg Stock Exchange using derivative instruments", *Omega, International Journal of Management Science*, Vol.25, No. 1, 1997, pp. 81-91.

R. R. Trippi, and DeSieno D., "Trading equity index futures with a neural network," *The Journal of Portfolio Management*, 1992.

L.S. Duke and Long, J., "Neural network futures trading - A feasibility study," *Adaptive Intelligent Systems*, Elsevier Science Publishers, 1993.

David de la Fuente, Alejandro Garrido, Jaime Laviada, Alberto Gomez, "Genetic Algorithms to Optimize the Time to Make Stock Market Investment". *GECCO 2006*, Vol.2, pp.1857- 1858, 2006

Cyril Schoreels, Brian Logan, Jonathan M.Garibaldi, "Agent based Genetic Algorithm Employing Financial Technical Analysis for Making Trading Decisions Using Historical Equity Market Data", IAT04, 2004

R. Lakshman Naik received his B.Tech. (Electronics and Communication Engineering) from Sree Sarathi Institute of Engineering and Technology, Jawaharlal Nehru Technological University (JNTUH), Krishna, Andhra Pradesh, India and M.Tech.(Computer Science and Engineering) from Balaji Institute of Technology and Sciences, JNTUH, Warangal, A.P, India. He served as a Systems Engineer in Wipro Technologies. He is a Member of IAENG, IAEME, AIRCC and various organizations. He has Publications more than 10 papers in area of computer networks, Data mining and neural networks.

D. Ramesh received his B.Tech. (Computer Science and Engineering) from Srineedhi Institute of Science and Technology, JNTU, Hyderabad, A.P, India and M.Tech. (Computer Science) from School of Information Technology, JNTU, Hyderabad, A.P, India. He served as an Academic Consultant in JNTU, Hyderabad, A.P, India. Now he is working as Assistant Professor in Department of Informatics, Kakatiya University, Warangal, A.P, India. He has Publications more than 4 papers in area of Data mining and neural networks.

B. Manjula received his BCA (Computer Application) from Osmania University, Hyderabad, A.P, India and M.Sc. (Information System) from Osmania University, Hyderabad, A.P, India. She served as an Academic Consultant in Department of Computer Science, Osmania Univesrsity, Hyderabad, A.P., India. And now she is working as Assistant professor in Department of Informatics, Kakatiya University, Warangal, A.P, India. She is a Member of IAENG and IAEME. She has Publications more than 11 papers in area of computer networks, Data mining and neural networks.

Dr. A. Govardhan is a Professor, Department of computer science and Engineering at Jawaharlal Nehru Technological University (JNTUH) Hyderabad (A.P), India. His research interests are in the area of Mobile Computing, Data management in wireless mobile environments and Data Mining. He has served on several program committees of conferences in the area of mobile computing, data management and sensor networks. He is a Senior Member of IEEE, ACM and various organizations. He has organized several workshops, delivered numerous tutorials at major IEEE and ACM conferences, and serves as editor of several journals and magazines.

This academic article was published by The International Institute for Science, Technology and Education (IISTE). The IISTE is a pioneer in the Open Access Publishing service based in the U.S. and Europe. The aim of the institute is Accelerating Global Knowledge Sharing.

More information about the publisher can be found in the IISTE's homepage:
http://www.iiste.org

The IISTE is currently hosting more than 30 peer-reviewed academic journals and collaborating with academic institutions around the world. **Prospective authors of IISTE journals can find the submission instruction on the following page:** http://www.iiste.org/Journals/

The IISTE editorial team promises to the review and publish all the qualified submissions in a fast manner. All the journals articles are available online to the readers all over the world without financial, legal, or technical barriers other than those inseparable from gaining access to the internet itself. Printed version of the journals is also available upon request of readers and authors.

IISTE Knowledge Sharing Partners

EBSCO, Index Copernicus, Ulrich's Periodicals Directory, JournalTOCS, PKP Open Archives Harvester, Bielefeld Academic Search Engine, Elektronische Zeitschriftenbibliothek EZB, Open J-Gate, OCLC WorldCat, Universe Digtial Library , NewJour, Google Scholar