

Emerging Issues of Credit Card Frauds and their Detection Techniques using Genetic Algorithm

Sudan Jha,
PhD Scholar

Associate Professor, Department of Graduate Studies
Nepal College of Information Technology
Kathmandu, Nepal

Abstract— CREDIT CARD has been widely used and simultaneously its malfunctioning as well as most importantly FRAUD cases have become an emerging issues in the recent trends in technologies. This paper tries to provide the work that helps in detecting the fraudulent card during transactions and alerts the customer regarding the fraud. This paper also aims in minimizing the number of false alerts. The concept of genetic algorithm is a novel one in this application domain. The algorithm begins with multi-population of randomly generated chromosomes. These chromosomes undergo the operations of selection, crossover and mutation. Crossover combines the information from two parent chromosomes to produce new individuals, exploiting the best of the current generation, while mutation or randomly changing some of the parameters allows exploration into other regions of the solution space. Natural selection via a problem specific cost function assures that only the best fit chromosomes remain in the population to mate and produce the next generation. Upon iteration, the genetic algorithm converges to a global solution.

Keywords— Fraud; Fraud types, Card Systems, Functional tools

I. INTRODUCTION

In recent years, the prevailing data mining concerns people with credit card fraud detection model based on data mining. Since our problem is approached as a classification problem, classical data mining algorithms are not directly applicable. So an alternative approach is made by using general purpose meta heuristic approaches like genetic algorithms.

This project is to propose a credit card fraud detection system using genetic algorithm. Genetic algorithms are evolutionary algorithms which aim at obtaining better solutions as time progresses. When a card is copied or stolen or lost and captured by fraudsters it is usually used until its available limit is depleted. Thus, rather than the number of correctly classified transactions, a solution which minimizes the total available limit on cards subject to fraud is more prominent. It aims in minimizing the false alerts using genetic algorithm where a set of interval valued parameters are optimized.

To develop a credit card fraud detection system using genetic algorithm, during the credit card transaction, the fraud is detected and the number of false alert is being minimized by using genetic algorithm. Instead of maximizing the numbers of

correctly classified transactions we defined an objective function where the misclassification costs are variable and thus, correct classification of some transactions are more important than correctly classifying the others.

The algorithm begins with multi-population of randomly generated chromosomes. These chromosomes undergo the operations of selection, crossover and mutation. Crossover combines the information from two parent chromosomes to produce new individuals, exploiting the best of the current generation, while mutation or randomly changing some of the parameters allows exploration into other regions of the solution space. Natural selection via a problem specific cost function assures that only the best fit chromosomes remain in the population to mate and produce the next generation. Upon iteration, the genetic algorithm converges to a global solution.

II. LITERATURE REVIEW

A. The Framework

The main problem lies behind data mining where the objective is to correctly classify the transactions as legitimate or fraudulent. For classification problems many performance measures are defined most of which are related with correct number of cases classified correctly. A more appropriate measure is needed due to the inherent structure of credit card transactions. When a card is copied or stolen or lost and captured by fraudsters it is usually used until its available limit is depleted. Thus, rather than the number of correctly classified transactions, a solution which minimizes the total available limit on cards subject to fraud is more prominent.

Since the fraud detection problem has mostly been defined as a classification problem, in addition to some statistical approaches many data mining algorithms have been proposed to solve it. Among these, decision trees and artificial neural networks are the most popular ones. The study of Bolton and Hand [1] provides a good summary of literature on fraud detection problems.

A.1 the Genetic Algorithm

Genetic algorithms in this paper aim at obtaining better solutions as time progresses. Since their first introduction by Holland [2], they have been successfully applied to many problem domains from astronomy to sports, from optimization to computer science, etc. They have also been used in data

mining mainly for variable selection and are mostly coupled with other data mining algorithms. In this study, we try to solve our classification problem by using only a genetic algorithm solution.

Pseudo code of genetic algorithm

Initialize the population

Evaluate initial population

Repeat

Perform competitive selection

Apply genetic operators to generate new solutions

Evaluate solutions in the population

Until some convergence criteria is satisfied.

A.2 the Selection Process

Selection is used for choosing the best individuals, that is, for selecting those chromosomes with higher fitness values. The selection operation takes the current population and produces a 'mating pool' which contains the individuals which are going to reproduce. There are several selection methods, like biased selection, random selection, roulette wheel selection, tournament selection. In this work the following selection mechanisms are used.

A.3 Tournament Selection

Tournament selection has been used in this as it selects optimal individuals from diverse groups. It selects individuals from the current population uniformly at random, forms a tournament and the best individual of a group wins the tournament and is put into the mating pool for recombination. This process is repeated the number of times necessary to achieve the desired size of intermediate population. The tournament size controls the selection strength. The larger the tournament size, the stronger is the selection process.

A.4 Elitist Selection

In order to make sure that the best individuals of the solution are passed to further generations, and should not be lost in random selection, this selection operator is used. So we used a few best chromosomes from each generation, based on the higher fitness value and are passed to the next generation of population.

A.5 Reproduction

To generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation. For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously. By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each new child, and the process continues until a new population of solutions of appropriate size is generated. Although reproduction methods that are based on the use of two parents are more "biology inspired", some research suggests more than two "parents" are better to be used to reproduce a good quality chromosome.

These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best organisms from the first generation are selected for breeding, along with a small proportion of less fit solutions, for reasons already mentioned above. Although Crossover and Mutation are known as the main genetic operators, it is possible to use other operators such as regrouping, colonization-extinction, or migration in genetic algorithms.

A.6 termination

This generational process is repeated until a termination condition has been reached. Common terminating conditions are: (1) A solution is found that satisfies minimum criteria; (2) Fixed number of generations reached; (3) Allocated budget (computation time/money) reached; (4) The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results; (5) Manual inspection; last but not the least (6) Combinations of the above

B. Feasibility

B.1 Operational Feasibility

The proposed paper will have high and enough operational reach, which ensures the security of the information among the general user friendly windows environment. Graphical user Interface, being today de facto standard.

B.2 Technical Feasibility

This paper emphasizes the possibilities necessary hardware and software resources to that are readily available. The technical requirements are thoroughly checked. Information regarding the upgrades in the technical aspects is gathered and is estimated with the technical features of the existing system. If the technical features that are available in the existing system are suited to accommodate the proposed system, then the system that has been developed is said to be technically feasible. As all the technology for this project is available in the latest Browsers, this project is technically feasible.

B.3 Echnomical Feasibility

Economic analysis, the most frequently used method, most commonly referred as cost/benefit analysis. Since the tolls and utilities as well as software used in this work are either freeware or open source so the cost is minimal.

III. SCOPE & ATTRIBUTES

The prime requirement is that no error condition causes our work to exit abruptly. Any kind of error occurred in any process should return an understandable error message. The response should be fairly fast, the action participants should not be confused at any point of time about action that is happening. The system performance is adequate.

C 1. Non Functional Attributes

a. SECURITY: - The proposed solution provides a security to different kind of customers by means of authentication

level. The authorization mechanism of the system will block the unwanted attempts to the server.

b. **RELIABILITY:** - The paper guarantees to provide reliable results for the entire user. The system shall operate 95% of the time. The number of defect should not exceed 10 per function. In addition, before the submission of the final release the calendar must be tested in case of the defects over 10 per function.

c. **USABILITY:** - Since use of GUI interface and online Interfacing is used, the system detects the fraud and reports to the user.

d. **SCALABILITY:** - The need for scalability has been a driver for much of the technology innovations of the past few years. The industry has developed new software languages, new design strategies, and new communication and data transfer protocols, in part to allow web sites to grow as needed.

e. **MAINTAINABILITY:** - Maintainability is our ability to make changes to the product over time.

C.2. The Current System and the Proposed System

a. **EXISTING SYSTEM:** -The existing system i.e. the Traditional detection method mainly depends on database system and the customers base knowledge, which usually are delayed, inaccurate and not in-time as a result discriminate analysis and regression analysis are widely used which can detect fraud by credit rate for cardholders and credit card transaction. The data itself is not presumed to be efficient. The high amount of losses due to fraud and the awareness of the relation between loss and the available limit has to be reduced. The fraud has to be deducted in real time and the number of false alert has to be minimized.

b. **PROPOSED SYSTEM:** - This paper overcomes the above mentioned issue in an efficient way. Using genetic algorithm the fraud is detected and the false alert is minimized and it produces an optimized result. The fraud is detected based on the customer's behavior. A new classification problem which has a variable misclassification cost is introduced. Here the genetic algorithms is made where a set of interval valued parameters are optimized.

c. **SYSTEM DESIGN:** - The process of design involves "conceiving and planning out in mind and making a drawing, pattern or a sketch". The system design transforms a logical representation of what a given system is required to do into the physical reality during development. Important design factors such as reliability, response time, throughput of the system, maintainability, expandability etc., should be taken into account. The task of system design is to take the description and associate with it a specific set of facilities-men, machines (computing and other), accommodation, etc., to provide complete specifications of a workable system.

This approach / system must provide for all of the essential data processing and it also possess additional option that the tasks may be identified during the work of analysis; thus

proving improvement over the existing system. At the outset of design, a choice must be made between the main approaches. Talks of 'preliminary design' concerned with identification analysis and selections of the major design options are available for development and implementation of a system. These options are most readily distinguished in terms of the physical facilities to be used for the processing who or what does the work.

IV. DESIGNS

D.1 Architectural Design: - During architectural design, the various web pages and their interconnections are identified and designed. The major software components are identified and decomposed into processing modules and conceptual data structures and the interconnections among the modules are identified. The following modules are identified in the proposed system.

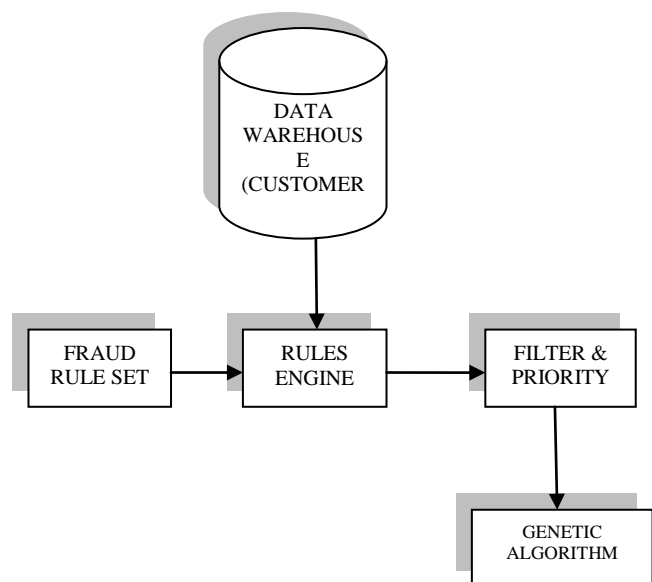


Figure 1:- OVERALL SYSTEM DESIGN

The overall system architecture describes work structure of the system in the following way: - (1) The customer data in the data warehouse is subjected to the rules engine which consists of the fraud rule set; and (2) The filter and priority module sets the priority for the data and then sends it to the genetic algorithm which performs its functions and generates the output.

D.2 Detailed System Design: -

Detailed design deals with the various modules in detail explaining them with appropriate Diagrams and notations. The Use case diagram is designed to see the working logic of the proposed system. The sequence diagram is designed to describe, how the client and the server interacts with each other when processing a content. The flow of the proposed system is described with the activity diagram. We know where the application starts and when it ends after processing the keywords and the current URL link. This will help the

programmers to implement the internal logic for the module in the given specification.

In this part of design phase, the design is carried out using the top-down strategy. First the major modules are identified. Then they are divided into sub modules so that each module at the lowest level would address a single function of the whole system. Each module design is explained detail. The input module is design in getting the users requirements. The detailed input design provides as information regarding what are tools used in getting inputs and send to the server.

Output design is gives the user with good interacting option on the screen. The information delivered to the users through the information system. Useful output is essential to ensure the use and acceptance of the information system. Users often judge the merit of a system based upon its output. Productive output can only be achieved via close interaction with users. The output is designed in attractive and effective way that user can access them with a problem.

D.3 Use Case Diagrammatic Representation: -

A use case diagram is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The below diagram figure shows the overall use case diagram for credit card fraud detection.

The USE CASE diagram below describes the interaction between the customers and card issuers

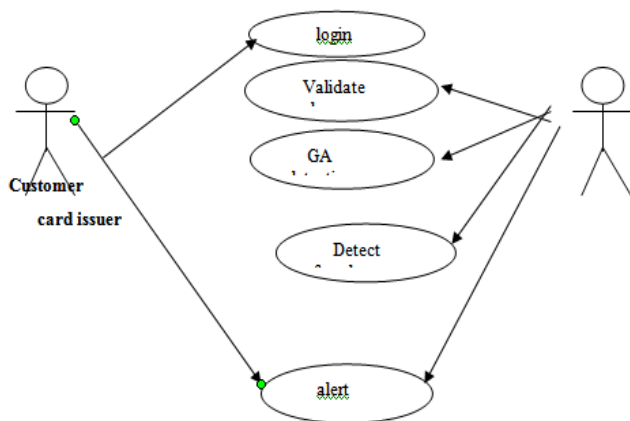


Figure 2:- Use Case Diagram of the overall structure

D.4 Flow of Genetic Algorithm: -

The diagram shown next states the process of genetic algorithm: Initially the initial population is selected randomly from the sample space which has many populations. The fitness value is calculated for each chromosome in each population and is sorted out.

In selection process two parent chromosomes are selected through tournament method. The Crossover forms new offspring (children) from the parent chromosomes using single

point probability. Mutation mutates the new offspring using uniform probability measure.

In elitism selection the best solution are passed to the further generation. The new population is generated and undergoes the same process it maximum number of generation is reached.

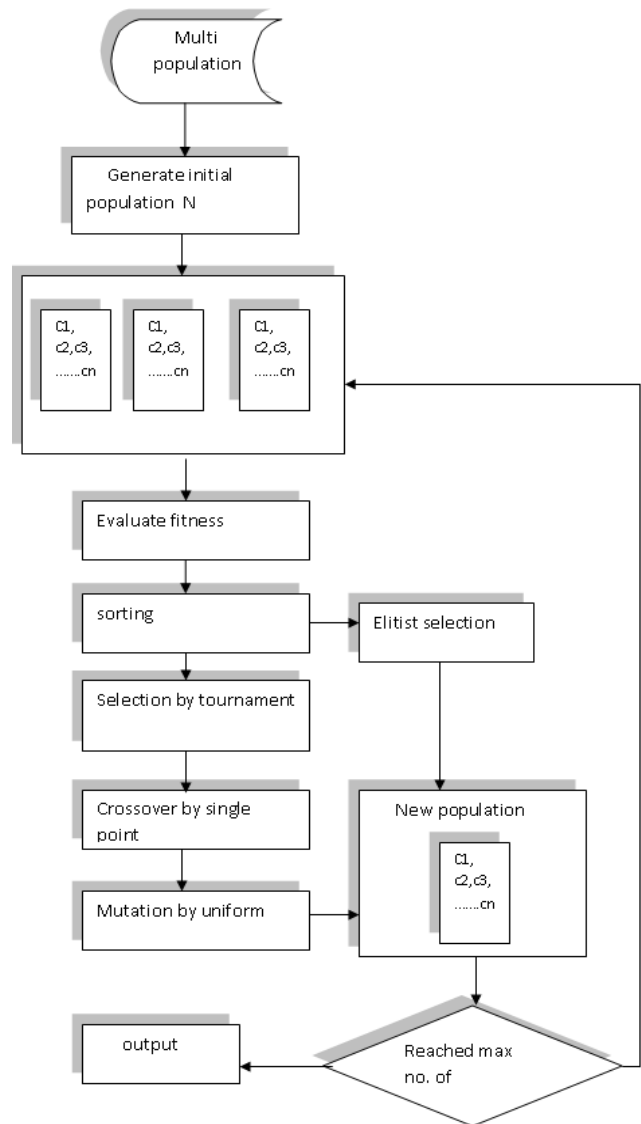


Figure 3:-Flow Chart of Genetic Algorithm

V. IMPLEMENTATION

It is the process of bringing developed system of revised system into operational use. If the implementation phase is not carefully planned and controlled, it can lead to many problems. Thus proper implementation is essential to provide a reliable system to meet managerial requirements. The

application is completely Java based. This enables the credit card issuers to use this application across wide variety of devices independent of the vendor of the devices. The paper work is based on the data cleansing and migration as well as back end by using Oracle 9i for storing database.

E.1 Coding

Standard coding practices are needed to ensure that the code is readable, understandable and easily modifiable. This approach has defined standards and guidelines to be followed while pseudo coding. These standards were followed during the development of the application to produce code that is more consistent and to make code maintenance.

Naming Conventions make programs more understandable by making them easier to read. They can also give information about the function of the identifier. All the controls used in the project were properly named according to their types.

E.2 Comments

The comments are used in the programs to improve the understanding of the code in a clear way. The complete code was properly commented. Appropriate comments were given for each function used, which described their functionality. Comments were also given for variable names to describe their purpose.

VI. TESTING

Testing is one step in the software/web engineering process that could be viewed as destructive rather than constructive. Testing requires that the developer discard preconceived notions of the "correctness" of the software just developed and overcome a conflict of interest. If testing is conducted successfully, it uncovers error in the software.

As a secondary benefit testing demonstrates that software functions appear to be working according to specification, that performance requirements appear to have been met. In addition data collected as testing is conducted provide a good indication of software reliability and some indication of software quality as whole. Testing cannot show the absence of defects, it can only show that software defects that are present.

The main objectives of testing are

- To ensure that during operation the system will perform as per specifications.
- To make sure that system meets the user requirements during operations.
- To make sure that during operation, incorrect input, processing and output will be detected.
- To see that when correct inputs are fed to the system the outputs are correct.
- Testing is a process of executing a program with the intent of finding errors.

F.1 Unit Testing

Developers write unit tests to check their own code. Unit testing differs from integration testing, which confirms that components work well together, and acceptance testing, which

confirms that an application does what the customer expects it to do. Unit tests are so named because they test a single unit of code. Unit testing focuses verification effort on the smallest unit of software design. Each of the modules in this project was verified individually for errors.

F.2 Intergration Testing

Integration testing is a systematic testing for constructing the program structure while at the same time conducting tests to uncover errors associated within the interface. This testing was done with sample data. The need for integrated test is to find the overall system performances. The Integration testing can be performed in the credit card fraud detection as follows.

In the Login Page User has not enter the Card Id and Pin number, but he/she clicks Sign In Button then the list of Errors should be displayed to the user as,

- Invalid Card ID
- Invalid Pin number

F.3 Validation Testing

Validation testing is where the requirements established as part of the software requirements analysis are validated against the software that has been constructed. It provides final assurance that the software meets all functional, behavioral and performance requirements. A deviation from the specification is uncovered and corrected. Each input field was tested with the validation rules specified integrity.

VII. CONCLUSION & FORESEEABLE ENHANCEMENTS

G.1 Conclusion: -

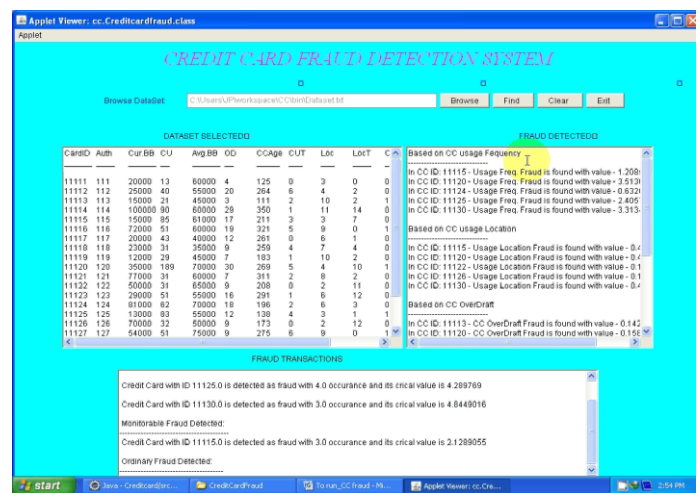
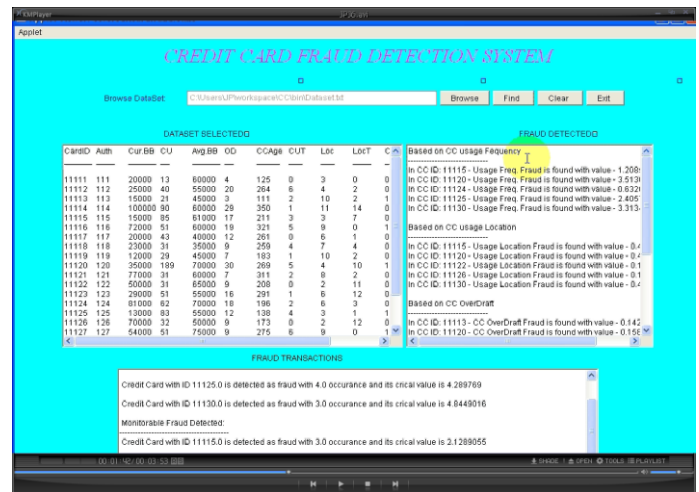
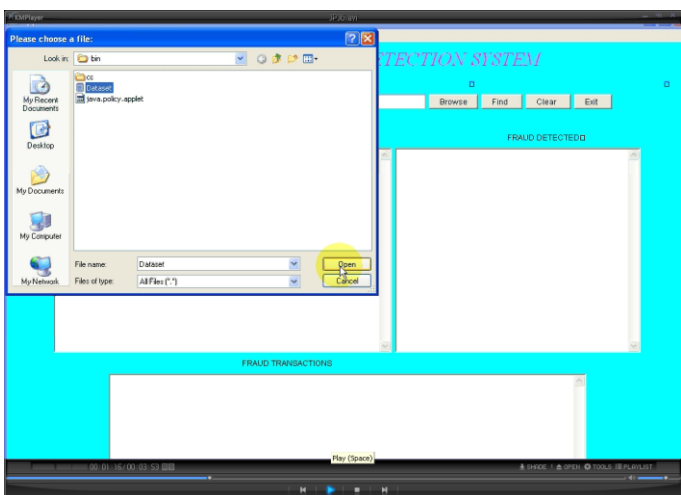
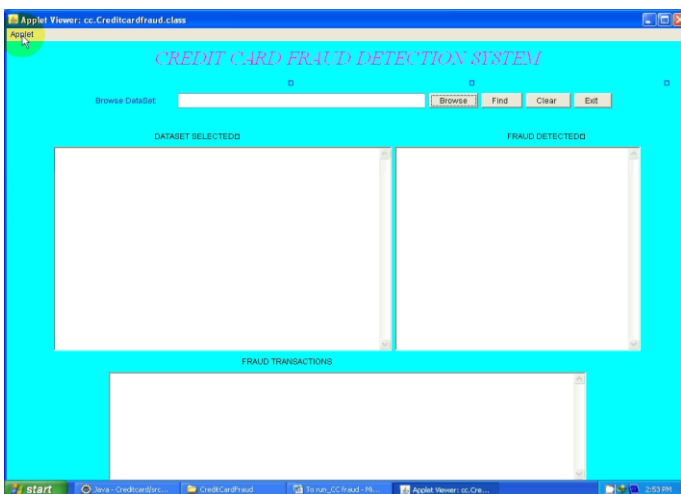
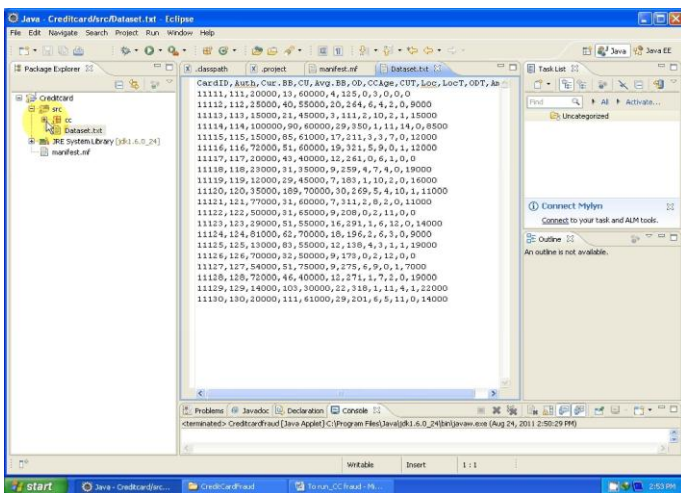
This method proves accurate in deducting fraudulent transaction and minimizing the number of false alert. Genetic algorithm is a novel one in this literature in terms of application domain. If this algorithm is applied into bank credit card fraud detection system, the probability of fraud transactions can be predicted soon after credit card transactions. And a series of anti-fraud strategies can be adopted to prevent banks from great losses and reduce risks.

The objective of this research work was taken differently than the typical classification problems in that we had a variable misclassification cost. As the standard data mining algorithms does not fit well with this situation we decided to use multi population genetic algorithm to obtain an optimized parameter.

G.2 Future Enhancements: -

The findings obtained here may not be generalized to the global fraud detection problem. As future work, some effective algorithm which can perform well for the classification problem with variable misclassification costs could be developed.

VIII. SNAPSHOTS OF PROPOSED SOLUTION GUI



REFERENCES

- [1] Shailesh S. Dhok, Credit Card Fraud Detection Using Hidden Markov Model ISSN: 2231-2307, Volume-2, Issue-1, March, 2012
- [2] Ghosh, S., and Reilly, D.L., 1994. Credit Card Fraud Detection with a Neural-Network, 27th Hawaii International Conference on Information Systems, vol. 3 (2003), pp. 621- 630.
- [3] Syeda, M., Zhang, Y. Q., and Pan, Y., 2002 Parallel Granular Networks for Fast Credit Card Fraud Detection, Proceedings of IEEE International Conference on Fuzzy Systems, pp. 572-577(2002).
- [4] Stolfo, S. J., Fan, D. W., Lee, W., Prodromidis, A., and Chan, P. K., 2000. Cost-Based Modeling for Fraud and Intrusion Detection: Results from the JAM Project, Proceedings of DARPA Information Survivability Conference and Exposition, vol. 2 (2000), pp. 130-144.
- [5] Aleskerov, E., Freisleben, B., and Rao, B., 1997. CARDWATCH: A Neural Network Based Database Mining System for Credit Card Fraud Detection, Proceedings of IEEE/IAFE: Computational Intelligence for Financial Eng. (1997), pp.220-226
- [6] M.J. Kim and T.S. Kim, "A Neural Classifier with Fraud Density Map for Effective Credit Card Detection," Proc. Int'l Conf. Intelligent Data Eng. and Automated Learning, pp. 378-383, 2002.
- [7] W. Fan, A.L. Prodromidis, and S.J. Stolfo, "Distributed Data Mining in Credit Card Fraud Detection," IEEE Intelligent Systems, vol. 14, no. 6, pp. 67-74, 1999.
- [8] B. Brause, T. Langsdorf, and M. Hepp, "Neural Data Mining for Credit Card Fraud Detection," Proc. IEEE Int'l Conf. Tools with Artificial Intelligence, pp. 103-106, 1999.