Ensemble Machine Learning Framework for Big Data Analytics

*A **Thesis***

submitted in partial fulfillment of the requirements for the award of degree of

DOCTOR OF PHILOSOPHY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

Nishtha Hooda

(Registration No: 901403021)

under the guidance of

Dr. Seema Bawa, Professor,

Dr. Prashant Singh Rana, Assistant Professor

Computer Science and Engineering Department,

TIET, Patiala-147004, INDIA

Computer Science and Engineering Department

TIET, Patiala -147004, INDIA

MAY 2018

Certificate

I, *Nishtha Hooda, Regn No. 901403021*, hereby declare that the work which is being presented in this thesis entitled, **"Ensemble Machine Learning Framework for Big Data Analytics"** in partial fulfillment of the requirement for the award of "**Doctor of Philosophy**" submitted in Computer Science and Engineering Department of TIET, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Seema Bawa and Dr. Prashant Singh Rana, refers other research works which have been duly listed in the reference section. The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

Nishtha 6/June/2019

(Nishtha Hooda)

Regn. No. 901403021

This is to certify that the above statements made by the candidate is correct and true to the best of my knowledge.

Verified by:

Seema Bawa June/06/2019

(Dr. Seema Bawa)

Computer Science and Engineering Department, TIET, Patiala-147001, India

6/6/2019

(Dr. Prashant Singh Rana)

Computer Science and Engineering Department, TIET, Patiala-147001, India

Acknowledgement

I owe my deepest gratitude to my supervisors *Dr. Seema Bawa* and *Dr. Prashant Sign Rana* for their invaluable advice and encouragement at every step of my research work. Without their unfailingly support and belief in me, this thesis would not have been possible. I would like to thank *Dr. Seema Bawa*, who suggested me this research topic. Without her continuous optimism concerning this work, encouragement and blessings, this research would hardly have been completed. I also express my warmest gratitude to my other supervisor, *Dr. Prashant Sign Rana*, for introducing me the interesting world of machine learning and data analytics. His guidance and supervision in the research laboratory analysis have been a true support during this work.

The contribution of my supervisors in this thesis is beyond their role as an academic supervisor and includes constant support on a personal level without which this journey of my research have never been completed. And for this I am truly grateful. They are great mentors for my life as well.

I am thoroughly grateful and highly indebted to *Mr O.P Pandey* (Dean of Research) for his constant support. I am grateful to *Mr. Maninder Singh* (Head of CSE department) for being a source of motivation. I also acknowledge the *Ministry of Electronics and Information Technology (MEITY)*, Govt. of India for supporting this research work. I would also like to thank my classmates *Miss Manvi Sharma*, *Mrs. Sujata Singla*, *Mrs. Prabhjot Kaur*, and *Mrs. Harmanjeet Kaur* for helping and encouraging me.

Finally, I would like to express my sincere and deep gratitude to my father *Mr. Ram Narain Hooda* and mother Mrs. *Sushila Hooda* for their blessings and having faith in me. I would like to mention my deepest gratitude to my father-in-law *Mr. Layak Singh* and my mother-in-law *Mrs. Rajesh Devi* for their love, encouragement, care, and support. I do not think I can ever repay the dept I owe them.

Additionally, I would like to give a special acknowledgement to my husband, *Mr. Kapil Chaudhary* for supporting me, and for having faith in me at every step. Words can never be enough to thank his constant encouragement.

Dedicated to my Spritual Masters of Past and Present

Abstract

Data is growing tremendously. Every domain is becoming data rich and hence, are more excited to use the concept of big data. Every business organization requires business insights. Lately, researchers are extensively embracing machine learning in diverse areas of research like health-care, astronomy, computational biology, finance, etc. The problem is that the big data concepts should be understood well. There is no threshold value that defines the size of big data. Big data Analytics is not only about the size of data but it is an opportunity to get valuable insights from the massive available data. Machine Learning (ML) applies scientific algorithms to the collected data with the goal of creating automated environment for making predictions or important business decisions.

Researchers around the globe are working on improving the machine learning algorithms for modeling prediction and analytics problems. No single best machine learning algorithm is present which is applicable for all the possible cases of problems. So, numerous research attempts have been made for improving the performance of machine learning models by developing an ensemble-classifier which is created by combining multiple machine learning models.

An ensemble learning serves as a powerful tool in machine learning as it employs multiple classifiers and works on optimizing the performance of base classifiers separately. Although it cannot always guarantees a success, but generally it offers better performance than a single classifier solution. By choosing a developing a special aggregation technique, an ensemble classifier can aid to scrutinize the risk of obtaining poor results from a single classifier system.

In this thesis, a modified variant of an ensemble builder, Multi Criteria based TOPSIS Ensemble (MCTOPE) is proposed. In the proposed method, three new modifications are introduced. Firstly, ensemble builder is developed as an automated process. One need not think about combining multiple classifiers manually. Secondly, the user is relaxed from defining the number of candidate classifiers. The MCTOPE automatically tries the combinations of classifiers and chooses the best performers. Thirdly, unlike other ensemble building techniques, candidate classifiers for building an ensemble are not chosen on the basis of

accuracy after the performance evaluation phase. MCTOPE employs multiple-criteria decision making (MCDM) based TOPSIS algorithm during the ensemble building process. The TOPSIS performance score is evaluated using multiple-performance criteria of classifier like accuracy, sensitivity, specificity, F score, area under ROC curve, etc.

The work presented in this thesis mainly focuses on utilizing the ensemble machine learning technique for predicting the target in two different case studies.

In the first case study, drug toxicity prediction problem is solved using MCTOPE framework. Two V's of Big data i.e. variety and value are focused. Complex, unstructured, and high dimensional drug molecular data is collected with an objective of finding valuable insights in order to predict the toxic/non-toxic class of a drug molecule.

In the second case study, Three V's of Big data i.e. variety, veracity, and value are focused. An unstructured audit data is collected with an objective of finding fraudulent/non-fraudulent class of a public firm. A web-application is offered to the auditors using R script and Django Python Web framework for prediction of fraudulent firm on the basis of input features. This web-application will help the auditors in automating a part of work before auditing the firm.

The results obtained from the experiments have proved the usefulness of ensemble machine learning models for fraud prediction during audit planning, and toxicity prediction during drug design and development. Hence, contributing the research area of an external auditing and biological computing.

Journal Publications

1. Nishtha Hooda, Seema Bawa, Prashant Singh Rana, "T-Ensemble Approach for Drug Toxicity Prediction," *International Journal of Computer Science and Information Security*, vol. 15, no. 1, pp. 545-548, Jan. 2017. (ESCI Indexed, Impact Factor 0.66)

2. Nishtha Hooda, Seema Bawa, Prashant Singh Rana,"B2FSE Framework for High Dimensional Imbalanced Data: A Case Study for Drug Toxicity Prediction", Special Issue on Computational Biology and Bio computing in Biological Complex and Big Data, *Neurocomputing*, vol. 276, pp. 31-41, doi 10.1016/j.neucom.2017.04.081, Feb. 2018. (**SCIE Indexed, Impact Factor 3.317**)

3. Nishtha Hooda, Seema Bawa, Prashant Singh Rana, "Fraudulent Firm Audit Risk Assessment Classification: A Case Study of an External Audit, vol. 32(1), pp. 48-64, doi 10.1080/08839514.2018.1451032, April 2018 *Applied Artificial Intelligence Journal*. (**SCIE Indexed, Impact Factor 0.67**)

4. Nishtha Hooda, Seema Bawa, Prashant Singh Rana, "MCTOPE Ensemble Machine Learning Framework for Fraudulent Firm Prediction: A Case Study, *Communicated in The Computer Journal*. (SCIE Indexed, Impact Factor-0.77)

5. Nishtha Hooda, Seema Bawa, Prashant Singh Rana, "Emerging Machine Learning Challenges and Practices in Big Data Analysis: A Review, *Communicated in Applied Computing and Informatics Journal*.

List of Abbreviations

AUC	Area under ROC Curve
BDA	Big Data Analytics
BI	Business Intelligence
BN	Bayes Net
CRISP-DM	Cross Industry Standard for Data Mining
DSM	Decision Stump
DT	Decision Tree
EML	Ensemble Machine Learning
HTS	High Throughput Screening
MCTOPE	Multi-Criteria based TOPsis Ensemble
ML	Machine Learning
NB	Naive Bayes
NN	Neural Network
PLM	Probit Linear Model
RF	Random Forest
ROC	Receiver Operating Characteristic
RS	Random Sample
SMOTE	Synthetic Minority Over sampling Technique
SVM	Support Vector Machine
TOPSIS	Technique for Order Preference by Similarity to an Ideal Solution
Tox21	Toxicology in the 21st Century
UML	Unified Modeling Language

Contents

Certificate **i**

Acknowledgement **iii**

Abstract **vii**

Publications **ix**

List of Figures xix

List of Tables xxii

1 Introduction **1**

1.1 Background 1

1.2 Big Data Analytics 2

1.2.1 Big Data 2

1.2.2 Big Data Analytics (BDA) 5

1.2.3 Business Intelligence 6

1.2.4 Big Data Analytics and Data Mining 7

1.3 Machine Learning (ML) 9

1.3.1 Supervised Machine Learning 9

1.3.2 Unsupervised Machine Learning 10

1.3.3 Machine Learning Approaches 11

1.3.4 Machine Learning in Big Data Analytics 13

1.4 Ensemble Machine Learning 16

1.4.1 Need of Ensemble Machine Learning 17

1.4.2 Techniques of Ensemble Machine Learning 17
1.5 Research Gaps 18
1.6 Problem Statement 22
1.7 Thesis Objectives 23
1.8 Thesis Contribution 24
1.9 Thesis Organization 25

2 Literature Review **27**
2.1 Big Data Analytics 27
2.1.1 Challenges in Big Data Analytics 28
2.1.1.1 Data Deluge 29
2.1.1.2 Big Data Capture, Transmission and Storage 32
2.1.1.3 Big Data Curation 33
2.1.1.4 Accelerating Data Analysis Algorithms 34
2.1.1.5 Big Data Visualization 34
2.1.2 Research Disciplines in Big Data 34
2.2 Machine Learning 38
2.2.1 Research Trends 39
2.2.2 Feature Selection 41
2.2.3 Classification and Prediction 43
2.2.4 Clustering and Outlier Detection 48
2.2.5 Association Rule Learning 52
2.3 Ensemble Machine Learning 53
2.3.1 Homogeneous Ensemble 58
2.3.2 Heterogeneous Ensemble 61
2.4 Multi Criteria Decision Making 62

3 Proposed Framework: Multi Criteria based TOPSIS Ensemble (MCTOPE) **67**
3.1 The Architecture of MCTOPE Framework 67
3.1.1 Layered View 67
3.1.2 Detailed Architecture 68

3.1.2.1 Data Preparator 70
3.1.2.2 Prediction Engine 71
3.1.2.3 Model Pool 74
3.1.2.4 Performance Evaluation 75

4 Design and Implementation of MCTOPE 79

4.1 Design 79
4.1.1 Structural Modeling 79
4.1.1.1 Class Diagram 79
4.1.1.2 Component Diagram 80
4.1.2 Behavioural Modeling 81
4.1.2.1 Usecase Diagram 82
4.1.2.2 Sequence Diagram 83
4.1.2.3 State-chart Diagram 83
4.1.2.4 Activity Diagram 84
4.2 Implementation 86
4.2.1 Experimental Setup 86
4.2.2 Class Balancing 86
4.2.3 Feature Importance and Feature Selection 88
4.2.4 Prediction Engine 89
4.2.4.1 Classification Models 89
4.2.4.2 Ensemble Model 99
4.2.5 TOPSIS Performance Evaluation 100
4.3 Drug Toxicity Prediction: Case study I 102
4.3.1 Problem Description 102
4.3.2 Need of Drug Toxicity Prediction 103
4.3.3 Dataset 104
4.3.4 Proposed Solution 104
4.3.4.1 Feature Extraction 104
4.3.4.2 Prediction Engine 104

4.4 Audit Fraudulent Firm Prediction: Case study II 105
4.4.1 Problem Description 106
4.4.2 Need of Fraudulent Firm Prediction 106
4.4.3 Dataset 108
4.4.4 Proposed Solution 109
4.4.4.1 Feature Extraction 109
4.4.4.2 Prediction Engine 114

5 Test and Comparative Analysis **115**
5.1 UCI Datasets 115
5.2 Testing Drug Toxicity Predictor 120
5.2.1 K-Fold Cross Validation 120
5.2.2 Experimental Results 122
5.2.3 Comparison with State-of-the-Art Methods 126
5.2.4 Testing AIDS Therapy drug molecules 127
5.3 Testing Audit Fraudulent Firm Predictor 129
5.3.1 K-Fold Cross Validation 129
5.3.2 Comparison with State-of-the-Art Methods 131
5.3.3 Testing Web Application 132
5.3.4 Test Case Execution 133

6 Conclusions and Future scope **149**
6.1 Conclusion 149
6.2 Future Work 151

References **153**

List of Figures

1.1 Big Data and Machine Learning Google Trends [3] 2
1.2 7 Vs of Big Data [132] 4
1.3 Big Data Analytics Opportunities [50] 5
1.4 Business Intelligence and Big Data Analytics Evolution 6
1.5 CRISP-DM Mining and Analytics Model [145] 7
1.6 Plotting Cholera cases using Clustering 11
1.7 Decision Tree 12
1.8 Bagging or Bootstrap Aggregation Technique of Ensemble Machine Learning 19
1.9 Stacking technique of Ensemble Machine Learning 19

2.1 Ten Fold Global Mobile Data Growth 30
2.2 Big Data Issues and Solutions [156] 30
2.3 Lambda Architecture [156] 31
2.4 Disciplines connected with Big Data Techniques [50] 35
2.5 Machine Learning Techniques 39
2.6 Research trends in machine learning and data analytics 42
2.7 Research of TOPSIS in different application areas 62
2.8 Combination of TOPSIS with different methods 64

3.1 MCTOPE Architecture 69
3.2 Architecture of Data Preparator 69
3.3 Architecture of Prediction Engine 72
3.4 Multi-Criteria Analysis Flow 76

4.1 Class Diagram 80
4.2 Component Diagram 81
4.3 Usecase Diagram 82
4.4 Sequence Diagram 83
4.5 State chart Diagram 84
4.6 Activity Diagram 85
4.7 Prediction Method 105
4.8 Proposed method as decision support system 106
4.9 Audit Work-Flow 108
4.10 Proposed framework for an Audit Field Work Decision Making 114

5.1 Topsis Score Analysis of Australian Dataset 116
5.2 Topsis Score Analysis of German Dataset 116
5.3 Topsis Score Analysis of Banknote Dataset 117
5.4 Topsis Score Analysis of Popfailure Dataset 118
5.5 Topsis Score Analysis of Wholesale Dataset 118
5.6 Topsis Score Analysis of Sonar Dataset 119
5.7 K Fold Cross Validation results of accuracy 120
5.8 K Fold Cross Validation results of sensitivity 121
5.9 K Fold Cross Validation results of specificity 121
5.10 K Fold Cross Validation results of specificity 122
5.11 Accuracy 123
5.12 Sensitivity 123
5.13 Specificity 125
5.14 AUC 125
5.15 TOPSIS Performance Comparison Analysis 127
5.16 Toxicity prediction of AIDS therapy drug molecules 128
5.17 K Fold Cross Validation results of accuracy 129
5.18 K Fold Cross Validation results of sensitivity 130
5.19 K Fold Cross Validation results of sensitivity 130
5.20 K Fold Cross Validation results of AUC 131

5.21 TOPSIS Performance Comparison Analysis 133
5.22 Home page of Fraudulent Firm Application 136
5.23 Fraud class testing of Fraudulent Firm Application 137
5.24 No fraud class testing of Fraudulent Firm Application 138
5.25 Positive field validation testing of Fraudulent Firm Application 139
5.26 Field validation testing of Fraudulent Firm Application 140
5.27 Field validation testing of Fraudulent Firm Application 141
5.28 Field validation testing of Fraudulent Firm Application 142
5.29 Field validation testing of Fraudulent Firm Application 143
5.30 Field validation testing of Fraudulent Firm Application 144
5.31 Field validation testing of Fraudulent Firm Application 145
5.32 Field validation testing of Fraudulent Firm Application 146
5.33 Field validation testing of Fraudulent Firm Application 147

List of Tables

1.1 Business Analytics and Mining Applications [219] 8

2.1 Types of Big Data Analytics [220] 28
2.2 Key technologies in Big Data 32
2.3 Free and proprietary softwares 36
2.4 Open research problems in Big Data Analytics 41
2.5 Classification problem in Big Data Analytics 44
2.6 Clustering problem in Big Data Analytics 50
2.7 Research Contribution in Ensemble Building 55

3.1 Layers of MCTOPE Framework 68
3.2 Training data for classification 74
3.3 Confusion matrix 76
3.4 Performance evaluation metrics 77

4.1 Machine learning classification methods 87
4.2 Target Sectors 109
4.3 Risk factors classification and other features in model 111
4.4 Sample data of the corporate sector unit 113

5.1 Description of validation dataset for validation for proposed framework . . 116
5.2 Experimental Results 124
5.3 Experimental Results 127
5.4 Toxicity prediction results 129
5.5 Experimental Results 133

5.6 Test Cases 135

Chapter 1

Introduction

In this Chapter, the background of research is presented. The concept of big data, evolving definitions of big data, big data analytics, and machine learning are introduced. The various types of data analytics, the challenges in the big data analytics are also discussed. Numerous machine learning techniques in data analytics and introduction to ensemble machine learning are covered in detail.

1.1 Background

Big Data has already drawn a remarkable attention for current researchers frontiers. More attention is given to the big data because data collection has become much cheaper now. Data is growing tremendously as it is generated by low-priced several information-sensors like mobile devices, wireless sensor networks, cameras, etc [180]. It has been reported that more than 2.5 exabytes of data is being generated everyday and technological per-capita capacity of the world to store information gets doubled after every forty months [89]. Every domain is becoming data rich and hence, are more excited to utilize the complex and massive data. The problem is that big data concepts should be understood well. There is no threshold value that defines the size of the big data. It is not only about the size but big data is an opportunity to get valuable insights from the massive available data [224]. Due to massive growth of data in last ten years, every business organization requires business insights. It helps them to improve their strategic as well as operational decisions. Latterly, researchers

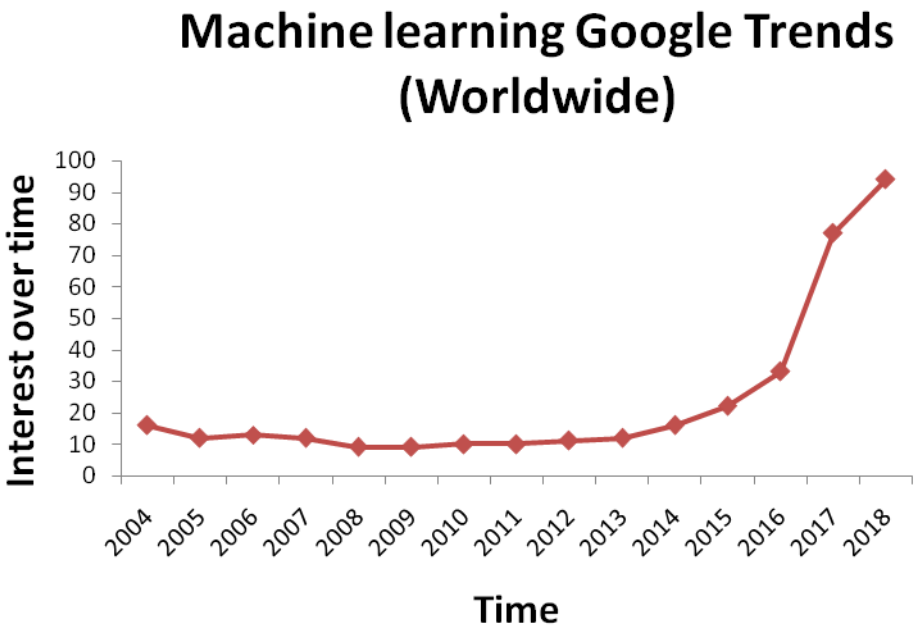

Figure 1.1: Big Data and Machine Learning Google Trends [3]

are embracing machine learning in various domains of research like medicine, astronomy, finance, etc. Figure 1.1 shows Google search trend of machine learning, depicting the interest of researchers in the field between 2004-2018. In last five years, it can be easily observed that the researchers are very excited about machine learning because of the availability of new tools and technologies.

1.2 Big Data Analytics

This section presents the concept of big data, data analytics and the business intelligence.

1.2.1 Big Data

"Big Data: it's not the data" [224]

In mid 1990s, the word "big data" is emanated in the lunch-table conversation at Silicon Graphics Inc. It became widespread in 2011 [26]. Over the years, the definitions of big data is evolving leading to the confusion in the mind of researchers. The various definitions of big data are:

"Big data is high-volume, high-velocity and high-variety information assets that demand

cost-effective, innovative forms of information processing for enhanced insight and decision making" [217].

"Big data is a term that describes large volumes of high velocity, complex and variable data that require advanced techniques and technologies to enable the capture, storage, distribution, management, and analysis of the information" [50]. The evolution of the Web and the prevalence of online information sharing in the society has led to an overabundance of information. Web 2.0 is the term used to describe the trend of World Wide Web technology whose aim is to promote the information sharing and collaboration among billions of users on the web [223]. YouTube, Flicker, Facebook, LinkedIn, Instagram are connecting billions of users around the world. So, the Web 2.0 is commonly called "Participation Web" because the rationale of Web 2.0 is to design the system which results in connecting more and more people together. This results in the information deluge, i.e. the problem of information overload. Size is the only dimension which is highlighted while defining big data. We cannot define big data by one single line as there are numerous explanations defining different characteristics of Big Data from 3Vs, 4Vs to 7Vs as shown in Figure 1.2 and are explained below [34]:

i Volume: It defines the big size of data-set.

ii Velocity: It defines the dynamic aspects of data i.e. the speed with which data in coming from different sources and going out after processing.

iii Variety: It defines multimodal nature of big data i.e. the diverse nature or types of data set i.e. structured, unstructured and semi- structured. Example: images, text, chats, tables, posts, tweets, videos, etc.

Although 3Vs defines the basic characteristics of big data, more dimensions are contributed by researchers in upcoming years to better understand and define the big data characteristics as explained below [132]:

iv Veracity: It specifies the truthfulness of the data. It includes the certainty in the facts defined in the data as well as the meaningfulness of results. In the absence of veracity, incorrect inferences will be drawn.

v Validity: It is similar concept to veracity but validity focuses more on accuracy and

correctness of data with regards to its usage and requirements.

vi Volatility: It defines that the specific data has some kind of retention period and removing such data after expiry is also needed in real time data storage. Example: An electrical business company destroys the history of a customer after 1 year warranty period is completed.

vii Value: It defines the required outcome for big data processing. It indicates the worth of data for those who are consuming it.

Big Data = Data + Value

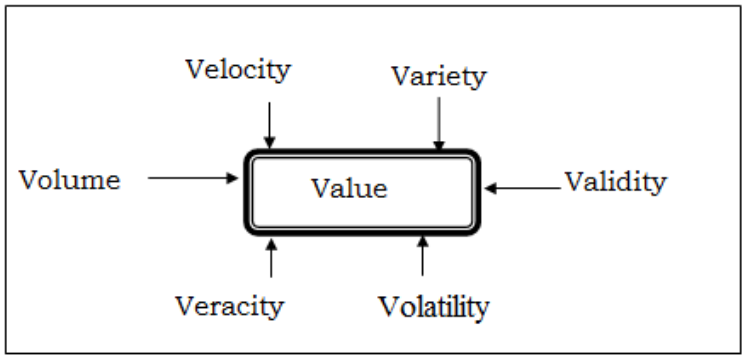

Figure 1.2: 7 Vs of Big Data [132]

All seven dimensions of big data are related to each other. There are several myths about big data. The realities behind the concept of big data are [224]:

i Big data is not only about the big size: The volume of big data gets undue attention. Big data is generally measured by the number of bytes it contains. It is because the size of data is easily measurable. When researchers have hundred gigabytes of data, they are generally confused whether the data is a big data problem or not. The problem is that big data concepts should be understood well. There is no threshold value that defines the size of big data. It is not only about the size but Vs like value of data and variety are also important.

ii Big data and data science are not the same: Some people call big data as data science only and use the terms interchangeably. The intent of data science is to focus on everything from data cleaning, preparation, explanation to data analysis. It is possible to do data science without big data and one can also involve big data with data science.

iii Big data is not a hype: Researchers are already working on data analysis and database system. Nothing have changed much. But, more attention is given to big data because data collection has become cheaper now. Every domain is becoming data rich and hence, are more excited to use the concept of big data.

1.2.2 Big Data Analytics (BDA)

Big data is a massive information assets that demand cost-effective, innovative forms of information processing techniques for enhanced insight and decision making [34]. Big data was considered as an extremely profound problem in early 2000. As the data was skyrocketing, the organizations were facing scalability issues. Now a days, business organizations are exploring big data as an opportunity to discover unknown facts about their data. It is about examining huge datasets to identify hidden patterns, real time insights, unknown correlations and hidden novel patterns [149]. BDA is an emerging area of research which offers systematic and advanced methods. Hence, we can consider BDA as a combination of big data and modern analytics. The rationale is to promote more profound business intelligence (BI) trends today [149, 257]. There are numerous benefits of big data analytics for business organizations like drastic improvement in their operational efficiency, better support for customer services, identifying new and wide range of products in the competitive market, etc. as presented in the Figure 1.3 [50].

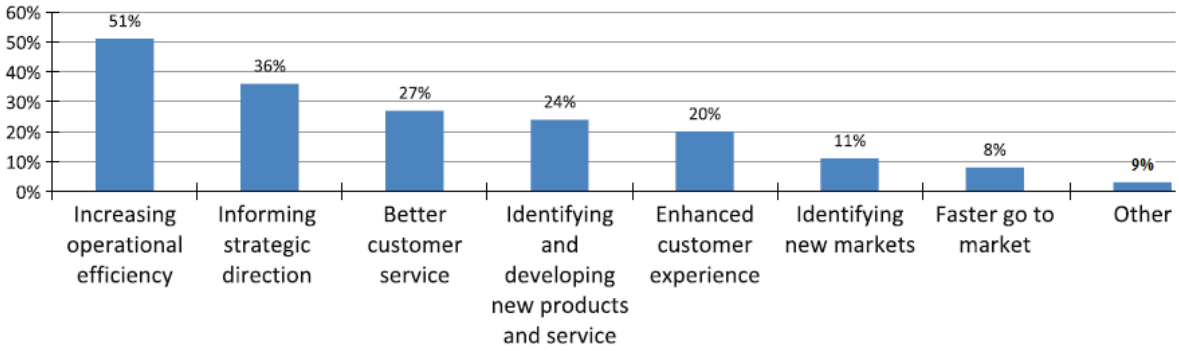

Figure 1.3: Big Data Analytics Opportunities [50]

1.2.3 Business Intelligence

Business Intelligence is an umbrella term that can be considered as the combinations of all the required advanced skills, technology, techniques, methods, people and practices that are helpful for business organization in their decision making process. Business Intelligence converts raw data into useful knowledge that becomes the asset for every business organization. Now a days, companies don't go for gut instinct for decision making. Highly competitive market demands spending huge amount of money on business intelligence (BI) tools and business analytics. As every business organization is data driven so the need to shift them to business analytics is highly desirable to survive in the today's market.

The relation of big data and business intelligence is depicted in the Figure 1.4. Business organization collects useful information for interpreting it for better decision making which will ultimately optimize their business process. Traditional Business Intelligence used to focus on descriptive analytics by analyzing their historical data but now due to the competitive business environment, organizations are working more on predictive analytics which can help them to avoid making costly errors in future [257].

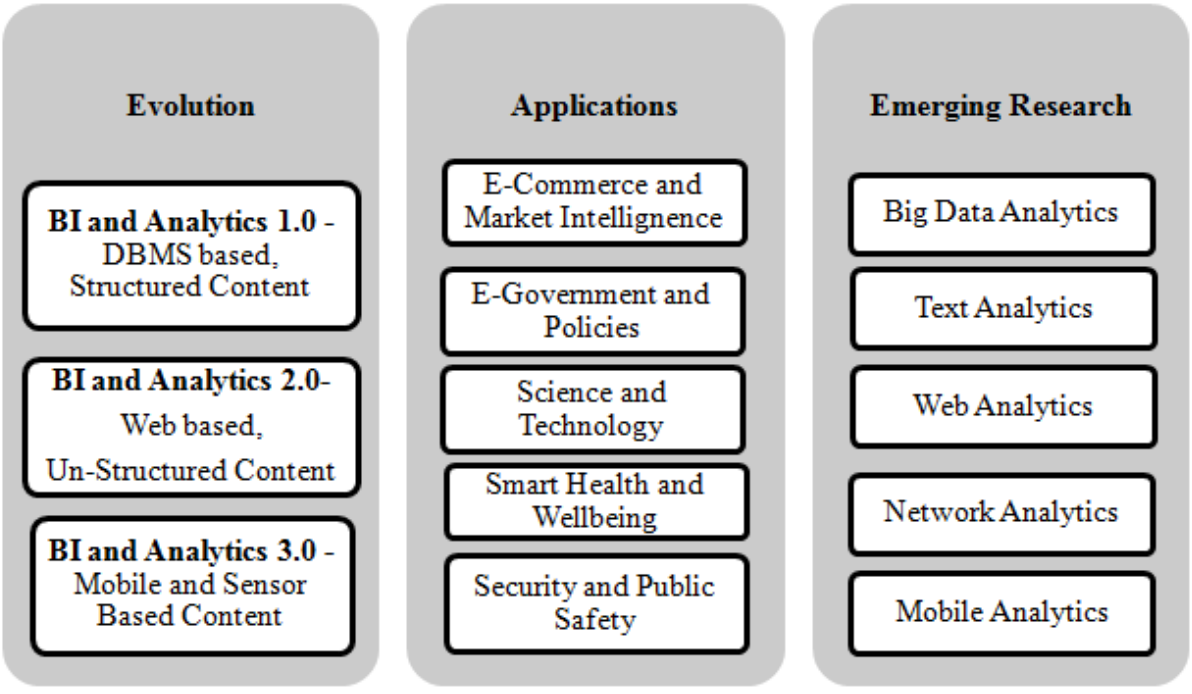

Figure 1.4: Business Intelligence, Big Data Analytics Evolution and Emerging Research [257]

1.2.4 Big Data Analytics and Data Mining

The most prevalent problem involves mining of the huge and complex data. Big data analysis requires numerous innovative techniques for developing an automated environment with the goal of discovering hidden patterns in the data. The newly discovered patterns help an enterprise in decision making process which is best explained by an industry standard CRISP Cycle as shown in the Figure 1.5.

CRISP is a cross industry process. It is usually known as CRISP-DM i.e. Cross Industry Standard for Data Mining. This offers the systematic approach for planning any data mining business project. It is a flexible methodology while using business analytics to solve big business problems as explained below [145]:

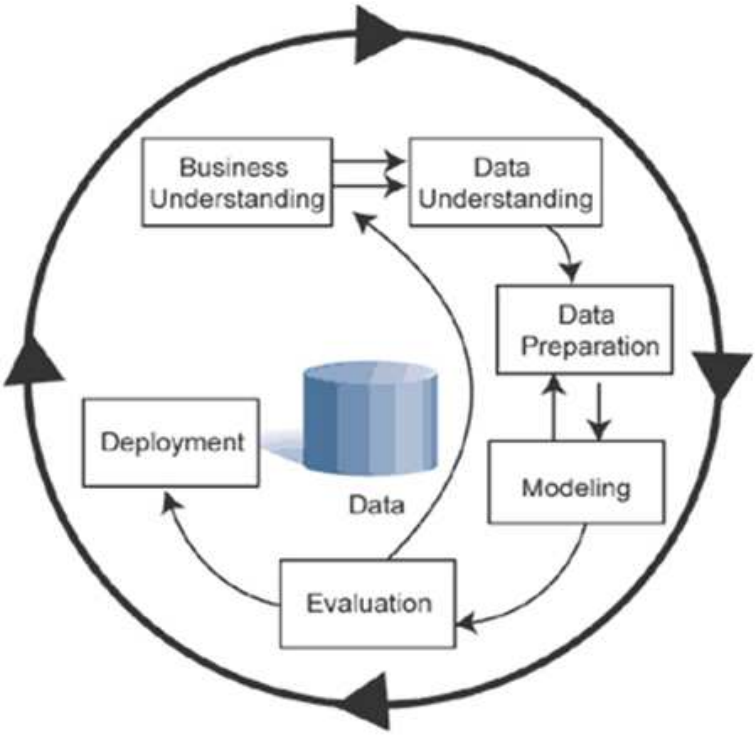

Figure 1.5: CRISP-DM Mining and Analytics Model [145]

i Business Understanding: It emphasize on assessing and deeply understanding the current situation of business and then setting the business objective. This will become the problem statement and also includes the goals of mining and analytics.

ii Data Understanding: Once the goal of project is established, the next goal should be the data understanding. Data is collected at this stage, explored and data quality is verified.

iii Data Preparation: This is an important step of preparing the data for analysis. It includes selecting the data for analysis that is relevant to the mining, and analytics goals . Useful data is included and reasons are mentioned for excluding the data. Data cleaning occurs at this phase to raise the quality of data.

iv Modeling: At this phase, best modeling technique is finalized. Example decision tree with C4.5 algorithm, clustering analysis with K-Means, etc.

v Evaluation: Although the accuracy and generality of model is already checked in the last step, next goal is to evaluate the level to which the selected model meets the business objectives. It includes testing the model on real applications.

vi Deployment: Using the evaluation results, this phase discovers the best strategy for deployment. This is the phase where predictive analytics are applied and results are shown to the customers. Errors and pitfalls are analyzed to improve the process in future.

The above model is used for the various applications of data mining and business analytics projects which can be classified into four groups and presented in the Table 1.1.

Table 1.1: Business Analytics and Mining Applications [219]

Prediction and Description	i. What customer would like to have? ii. Sales forecasting and analysis.
Relationship and Marketing	i. Discover sales triggers. ii. Identify critical issues.
Customer Profiling	i. Identify the need of most valuable customers. ii. Facilitate loans and promote offers.
Segmentation, Outlier Identification	i. Assembling similar objects. ii. Finding dissimilar objects as outliers. iii. Cluster analysis. iv. Fraud detection.

1.3 Machine Learning (ML)

Instead of explicit programming instructions only, Machine Learning (ML) applies scientific algorithms to the collected data with the goal of creating automated environment for making predictions or important business decisions [103, 116]. ML is considered as the branch of artificial intelligence and can only be considered as a good approach if the practitioner has very little idea about what they are looking for in the data. The most prevalent problem involves mining huge data sets. Researchers accept data mining as an algorithmic problem. There are several modelling approaches that are used for data analysis [103, 116].

Major ML algorithms taxonomy is divided three major parts, which will be discussed in detail in the subsections [68]. (i) Supervised Machine Learning, (ii) Unsupervised Machine Learning, and (iii) Semi-Supervised Machine Learning. Supervised Learning technique is applied when the input data collected has some kind of known labels or results. In Unsupervised Learning technique, input data does not have known results. Here, the model is prepared by detecting any kind of structure in the input data. In Semi-Supervised Learning technique, input data is the mixture of both labeled as well as unlabelled data. This method utilizes both supervised and unsupervised algorithms to model the problem with goal of both structuring the data and making predictions.

1.3.1 Supervised Machine Learning

Supervised Learning technique is applied when the collected input data has some kind of known labels or results. Input data is called training data and model is prepared by using this data. Eventually, the trained model is employed to do the future predictions. Prediction accuracy plays a vital role while training process. It is required to repeatedly train the model using training data until it achieves the desired level of prediction accuracy. Example: classification and regression problems.

The explosive growth of data demands to explore innovative research to extract and use information more tactfully. Different data mining techniques support different ML ap-

proaches to solve different kind of problems. There is no one approach that fits for all the applications. Different algorithms are used to model the problem. Selecting appropriate machine learning model that fits the data is important to get the highest prediction accuracy.

Ensemble technique is applied to revamp the prediction accuracy of model by combining many weaker algorithms together. Researchers around the globe are working on efficiently applying ensemble algorithms for modeling the prediction and data analysis problems.

1.3.2 Unsupervised Machine Learning

In Unsupervised Learning technique, input data does not have known results. Here the model is prepared by detecting any kind of structure in the input data. Example: clustering and association rule mining. The main intent of association rule learning approach is to find hidden and profitable relations between different features the data [44]. The technique is famous for discovering purchasing trends in the market. For instance, there is a relation between pizza, potato chips, and coke like pizza, potato => coke. The people who are purchasing pizza and potato chips are likely to have coke with it. In practical applications, millions of market transactions are observed and association rules are discovered. There are various algorithms that are used to generate association mining rules like Apriori algorithm, FP-Growth algorithm, etc [264]. During data analysis, summarization technique is also done in which the complex structure of huge data is summarized to some simple idea. Example: Google summarize complex web pages by single number i.e. page rank. Another famous example of summarization is clustering. Here, data is assumed as the collection of points of objects in the multi dimensional space. The objects which are close to each other are assigned in the same cluster. The distance is measured by various metrics like euclidean distance, manhattan distance, etc.

Clusters are summarized by giving centroid to the various compact cluster. Example of one case of clustering is that one physician plotted the cholera cases on the map of London as shown in Figure 1.6. After clustering process, it was found that most of the cases are around the intersection of roads where contaminated wells were present. Thus, the cause of cholera was discovered with clustering [103].

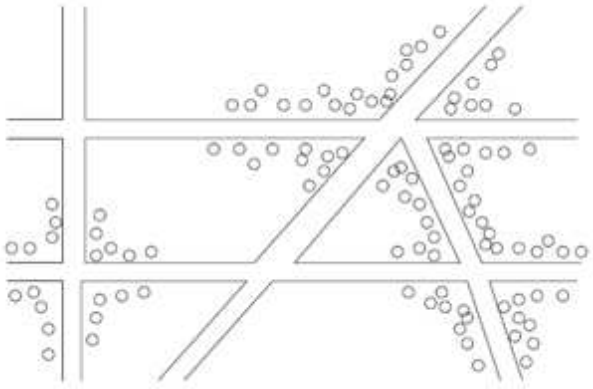

Figure 1.6: Plotting Cholera cases on London Map using Clustering [103]

1.3.3 Machine Learning Approaches

Depending upon the nature of input data, different ML algorithms are used to model a problem. Earlier, statisticians use Statistical Modeling (SM) in which they consider data mining as the construction of efficient statistical model from which underlying distribution can be drawn. Example: We have set of data and statistician found that the data comes from Gaussian distribution that will be called as the model of data. Thus, the researcher can use different Gaussian parameters like mean, standard deviation to depict the nature of data.

In ML modeling approach, researchers use training data set to train the system. Most of data mining algorithms comes from machine learning. Example decision trees, hidden markov model, etc. As ML addresses bigger and complex problems, the need of focusing the most relevant information to train the system and rejecting the irrelevant one is potentially an important problem. It is also called feature selection. Feature selection picks-up the relevant samples for training data and discarding the irrelevant one with the desired level of accuracy. Example: Rating of a movie in the Netflix challenge is predicted by using the responses of different users on it [93].

i. Decision tree learning is an admired learner which works on constructing a decision tree during learning process [179]. The target function should be a discrete function. If the target function is a finite set of values in decision tree analysis, it is classification tree. But, if the target function is a set of continuous values like real numbers, it is called regression trees. Decision tree represents samples as the branches of the tree and the target function is

represented as the leaves. Decision trees are used in machine learning to represent data so that it can be used for predictions and decision making. The goal is to predict the target value based on several features of data samples as represented in the Figure 1.7.

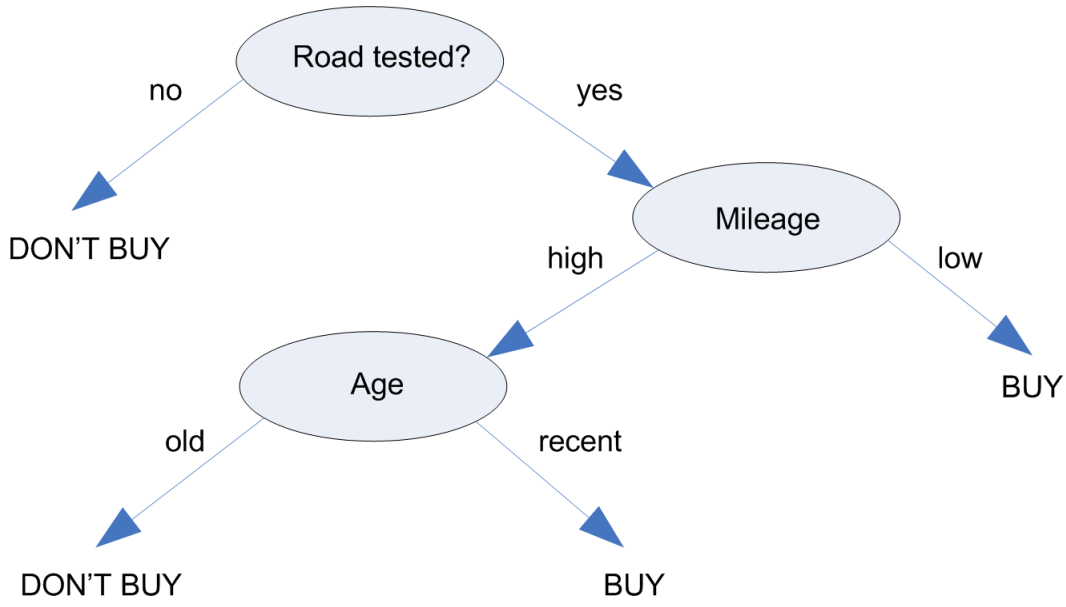

Figure 1.7: Decision Tree

ii. Artificial neural network algorithms are famously studies as as neural networks [203]. Neural network based algorithms are inspired from the structure of biological neurons of human brain. Neural network based model works as computing model having several processing units which are inter-connected with each other. These processing units are known as neurons which work together to provide the required output. The idea is similar to the working of trillions of nerve-cells working inside the human brain. Neural network based machine learning algorithms are generally preferred when there is a complex relationship between the input features and output variables. Such data are available in the domains like speech recognition, image processing, biology, etc.

iii. Deep learning is an extension of neural networks but deep learning algorithms are much powerful than neural networks [200]. It is powerful because it consists of many hidden layers of non-linear processing layers of artificial neural networks and propositional formulas between input and output layers. Deep neural networks are capable of solving much powerful data problems related to audio recognition, natural language processing. Deep learners are useful in other domains like computer vision, bio-informatics.

iii. Support vector machine (SVM) can be used for both regression and classification problems. Support vector machine algorithms can efficiently perform classification for linear and non-linear data. It maps features into high dimensional feature space by employing kernel [14]. SVM machine learning models represent samples as points in the space and map them in a way so that clear gap is identified into different category of classes as much as possible. The same representation helps in prediction the new testing samples for prediction. SVMs are popularly used by researchers for text categorization and image classification problems.

iv. Bayesian belief networks or bayesian networks represent data as directed acyclic graph. Hence, they are known as probabilistic graphical machine learning models [160]. For instance, to model the relationship between diseases and symptoms, bayesian classifier can be applied. If we know the symptoms, probability of having a disease can be calculated with the help of bayesian networks. Bayesian networks are used in every field by researchers like computational biology, bio-informatics, information retrieval, semantic search, etc.

1.3.4 Machine Learning in Big Data Analytics

There are different applications of big data analytics using machine learning techniques which are discussed in this section.

i Targeting customers and optimizing business processes:
This is the most advertised area where data analytics is used by business organizations to understand the interests and preferences of their customers. The data is generally collected through browser logs, sensor data, social media like facebook, twitter, etc. For instance, recommender engine targets customers by building a predictive model of their respective customers based on their past purchases and browsing history. On the basis of machine learning prediction models, business organizations optimize their business processes like optimizing stock based on data collected from social media. Mostly business organizations apply business analytics for optimizing their supply chain management [1, 2, 50].

ii Health care data anaytics:

Now a days, health care sectors maintain huge volume of patients data. Advanced analytical tools are used to harness such data in order to detect health insurance claims, finding errors, recurrent losses thus, enhancing health services. For instance, a framework is proposed for improving the information retrieval of massive medical records [8]. Because of technological advancement, decoding DNA string takes minutes now which is helping to the doctor to better cure the diseases. Eighty percent of the data is unstructured. The data used to treat the patients these days is limited to only basic information but social media and sensor data helps to better analyze the details using data analytics. It is also used to analyze the data of millions of pre mature babies. In fact, machine learning predictive algorithms are designed to predict the diseases before hand by recording the heart beat and other useful information [2, 159, 184]. To effectively utilize health resources, length of stay of patient and other factors are used to propose an intelligent capacity management system [118]

iii Improving winning rate in sports:

Popular sports now use data analytics. Arsenal, Premier League Soccer team has invested millions for using big data analytics for improving their winning rate. Sensor data is collected through cameras installed in the stadium to track every player. It not only includes collecting information on the ground but also the smart data is collected to track the important things like social behaviour, emotional balance, health, and sleep information. IBM Slam Tracker tool is used for tennis to analyze the performance of players by applying video analytics. Automated algorithms are used to improve the accuracy of analysis [1, 2].

iv Fraud detection in banking and insurance:

Banking and insurance companies use data analytics to grasp the opportunity of knowing the demands and preferences of their customers. Sixty percent financial organizations consider that data analytics is playing a significant role for competitive advantage. The target is to resolve the client problem beforehand. This deliberately helps the companies to improve the customers satisfaction in the competitive market. Most needful efforts are made by data analytics using machine learning techniques is to detect frauds and security breaches. Like IBM provides big data analytics solutions to their clients

to quickly detect and mitigate different types of fraud [1, 2, 15].

v Government security and law enforcement:

Government conducts variety of programs to collect real time data from numerous sources like cell phones, sensors, social media, etc. This data is generally unstructured and is need to be used to fight against crime in defence, national security, revenue, etc. Data analytics technology provides a platform to efficiently manage this data. For instance, denial of Service attack is detected by researchers by employing neural network model [133]. Police these days also apply analytics using machine learning and data mining tools to predict criminal activities. Example IBM Terra Echos utilizes sensor data for surveillance system to locate threats beforehand. National Security Agency (NASA), U.S. successfully used big data analytics to detect terrorist attacks [1, 2, 256].

vi Competitive advantage in telecommunication market:

Due the advancement in social media, unstructured data of wide variety is coming at an enormous speed. But, this data is used by service providers to gain market share in the competitive market. By delivering smarter services for telecommunication and achieving excellence in the service, network data analytics tools are used globally. Due to the ease in availability of smart phones, network operator markets has now more competitive. This leads to the arrival of excellent call centre services which resolves customers issues in seconds. To sustain in such market, gaining insights through data anaytics tools and machine learning approaches are extremely needed [1, 2, 46].

vii Understanding customers in travel and transportation:

In order to improve the customer experience, transport and travelling organizations use data analytics tools. The infrastructure is enhanced according the needs of their customers. As customers usually compare the fare rates which are changed daily by competitive organizations, so applying fare analytics to sustain the market is another critical problem for such travelling organizations. Lucrative services are offered by collecting the customer's data through social media, sensor data and feedbacks. Tactfully analyzing and harnessing the terabytes of fare data across the globe is challenging yet accomplished through data analytics tools [1, 2, 24].

viii Harnessing smart grid and smart meters data:

The data collected from smart grid and smart meters is highly unstructured and is being generated with enormous speed. Managing such data to solve critical business problems is a challenging task. Data analytics help in integrating such data and gaining insights in order to dig out the useful, unknown, and profitable patterns. Prediction analysis and machine learning techniques help the organizations to improve the forecasting and scheduling of distributed, renewable and precious assets. IBM Vestas makes use of supercomputer and big data analytics tools to harness huge data for accurately placing the turbine [1, 2, 99].

ix Optimizing cities infrastructure:

Now a days, data analytics tools are widely helping in optimizing city traffic based data collected on real time traffic information, social media and weather reports. In fact, many countries are successfully applying data analytics to improve their transport infrastructure so that can be counted as smarter cities. Like decision would be taken using analytics that which bus would wait if the train is delayed or weather is bad to minimize jams in the cities [1, 2, 221].

x Improving science and research:

Experiments in the research labs generate a huge amount of unstructured data that is need to managed efficiently. Like at CERN physics lab in Swiss, the world's largest particle accelerator called Hadron collider is used and experiments are carried out of big projects generating enormous amount of data with high velocity. Data centres are used by Swiss to analyze data using advanced data analytics tools [1, 2, 229].

1.4 Ensemble Machine Learning

Ensemble learning revamps the prediction performance of a learning model by combining different algorithms together. Researchers around the globe are working on efficiently applying ensemble machine learning algorithms for modeling prediction and analytics problems. Dietterich reviewed various ensemble methods like adaboost, random forest, etc. in his studies [79]. It is uncovered in the experiments that an ensemble works more efficiently when

compared to a single learner in terms of their accuracy in classification and prediction [79].

1.4.1 Need of Ensemble Machine Learning

As stated by Wolpert, there is no single best algorithm which is applicable for all the possible cases of problems [64]. Furthermore, many researchers strived to improve the performance of machine learning models by developing an ensemble-classifier which is constructed from diverse machine learning models [79, 247]. From a practical point of view, multiple-opinions are unfailingly better than a single opinion in any decision making process. For instance, suppose there are 25 base classifiers. Each classifier has error rate, $\varepsilon = 0.35$. Assume classifiers are independent. Probability that an ensemble classifier makes a wrong prediction can be calculated as:

$$\sum_{i=13}^{25} \varepsilon^i (1-\varepsilon)^{(25-i)} = 0.06 \tag{1.1}$$

Ensemble learning serves as a powerful tool in machine learning as it employs multiple classifiers and works on optimizing the performance of base classifiers separately. Although it cannot always guarantees a success, but generally it reduces variance and offers better performance than a single classifier solution [79, 212, 247]. By choosing a specific aggregation technique like majority voting, boosting, bagging, etc., an ensemble classifier aids to scrutinize the risk of obtaining poor results from a single classifier system.

1.4.2 Techniques of Ensemble Machine Learning

The three most well-liked techniques for building an ensemble which are known as "meta-algorithms" approaches for combining different ML models into one efficient predictive model. They are explained below:

i Bagging:

It stands for bootstrap aggregation. In this technique, an ensemble model is constructed by utilizing the random sub-samples of the training data and by combining multiple homogenous models i.e. the models typically of the same type. The sub-samples are randomly generated and sampling with replacement from the original dataset and known as bootstrap samples as shown in the Figure 1.8. The advantage of decreasing the size

of training data helps in reducing the variance of the prediction. It is generally used when the base classifiers are unstable like neural network, decision tree, etc. and small changes in the training dataset can cause large changes in the learning classifier. Example of ensemble implementing this technique is random forest.

ii Boosting:

In this technique, an ensemble model is constructed using subsets of the training data and combining multiple homogenous models i.e. the models typically of the same type. But, unlike bagging, the sub-samples generated from the training data are not random. The generation of every new sub-sample depends upon the performance of the previous models i.e. it contains the records that were misclassified by the previous models. The sub-samples of the training data are used to produce a series of averagely performing models and then their performance is boosted by combining them using a specific cost function like majority voting. The advantage of boosting is to improve the predictive force. Example of ensemble implementing this technique is adaboost.

iii Stacking:

In this technique, multiple models typically of different types are used for combination. The combining mechanism is different in the sense that the output of the classifiers of *Level (N-1)* will be used as training data for another classifier of *Level N* to approximate the very same target function as shown in the Figure 1.9.

1.5 Research Gaps

The various research gaps while reviewing various techniques of machine learning and big data analytics are presented as follows:

i Need for handling Data Drifts Issues

In big data streams, hidden patterns also evolve with time. This leads to data drifts issues. In predictive big data analytics, when the statistical properties of the target feature changes with time out of the blue, prediction accuracy also brings down in unforeseen ways. Growing use of spatio-temporal trajectory data like information of moving objects using GPS, virtual globes, images captured by remote sensors etc demands more

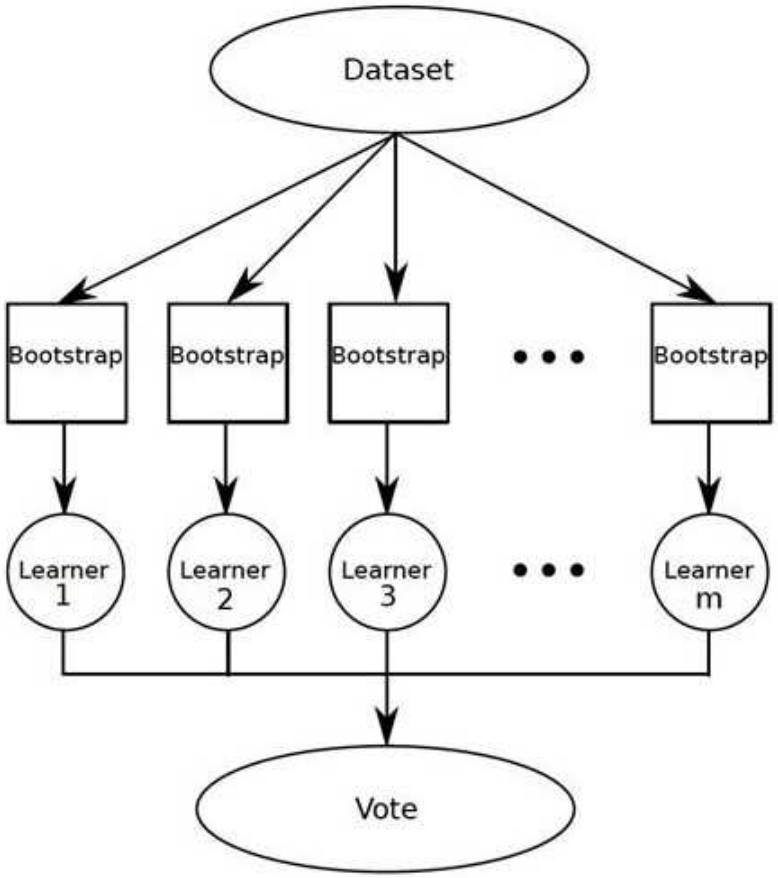

Figure 1.8: Bagging or Bootstrap Aggregation Technique of Ensemble Machine Learning

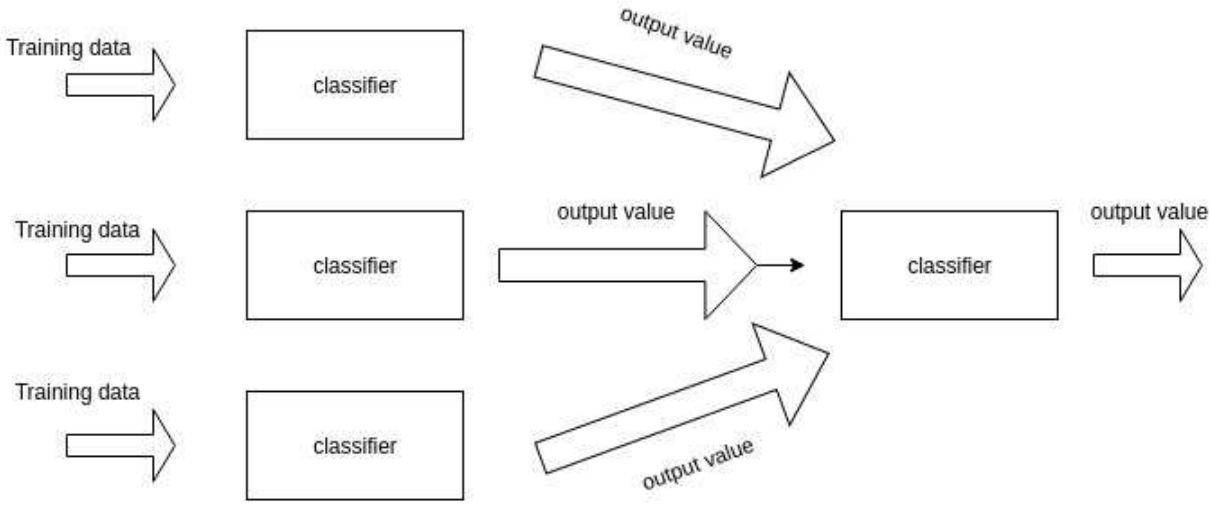

Figure 1.9: Stacking technique of Ensemble Machine Learning

efficient framework. More concern is required to introduce efficient algorithms adaptive to data drifts while processing evolving big data streams for analytics. It demands more innovative research to apply the results in different domains like for public safety to detect crime spots, to monitor diseases for health care, etc [50, 83, 192].

ii Handling High Dimensionality Issues

Several researchers contributed different efficient algorithms for handling big data. But, for such huge and massive data, it is always suspected to contain redundant information. As performance of algorithm scales down while analyzing data in high dimensional space, it demands trimming useless information using tools like feature extraction, subset selection etc. during pre-processing. More efficient framework contributions are needed to mark the relevancy of features to build efficient frameworks for big data analytics [50, 201].

iii Ensemble Classifier Analytics Framework for Hybridization

Hybridization of many classifiers to improve the prediction accuracy gives new direction to the performance of predictive analytics. Although such combinatory classifiers gives good results to small data but the performance of predictive models lowers down for big data. Researchers around the globe are proposing ensemble classifiers for big data but much more innovative research inventions are demanded to apply such results on big data applications of different domains like neuro-imaging, bio-informatics, finance, etc [47, 50, 83, 94, 192, 198, 201] .

iv Efficient Classification Framework for handling Imbalanced Data

Imbalanced data-streams is very interesting problem resulting in uneven classification distribution eg. 95 percent of negative class and 5 percent of the positive class. When such problem arises, classical machine learning classifiers (considering it as balanced training set) get biased usually towards majority class. As the mining is becoming more pervasive, it is important to handle such challenges for getting accurate predictions in large data analytics. Applying analytics to such datasets is another promising area of research and needs more efficient work [66, 158, 195].

v In-situ Information Visualization Framework

The boom in big data makes researchers to drive their attention on using information visualization techniques for big data analytics. But scalability is the major issue while dealing with the visualization of big data for analysis. The goal of in-situ visualization is to dynamically change the representation whenever a new data arrives. Innovative research contributions are demanded to visualize the big data information incrementally. Numerous research efforts in this field are favoured for a specific application or for any special aspect of visualization technique. Visualization developers are in dire need for efficient frameworks generic enough for wide range of domains [50, 175].

vi Anomaly Detection Framework for Streams

Different anomaly detection algorithms like outlier based cluster analysis, nearest neighbour etc have been proposed over the time for detecting outliers in big datasets. Although much work is contributed to find solutions using serial elision, innovative research is required to implement parallelism like map reduce to revamp the overall performance of proposed frameworks. Such results are need to implemented for real time big data applications in various domains like finding frauds, medical diagnosis, insurance, etc [104, 193, 194].

vii Efficient Frameworks for Curation

The results of analyzing large scale complex data can be potentially fruitful only when the quality of raw data is tested before analytics. Real world data that are collected from heterogeneous sources like social media, sensors, etc. mostly consists of noise, missing and inconsistent values. For this reasons, researchers are now more concerned about finding new techniques for testing and validating datasets. More contribution is required to find best solutions for various issues like defining test strategies for functional and non functional testing of big datasets, setting up optimal test environment while dealing with non relational databases. Very less work is contributed for performance and fail-over testing to handle performance related issues for big datasets testing [50, 100].

viii Diversity Impact in Ensemble

Researchers are studying the impact of diversity for building ensemble to improve their performance [121]. It is not intuitive but weak classifiers can be used to build stronger

ensemble[183]. Very less work has been done on studying the impact of diversity while building ensemble [49, 52].

ix Multi-Criteria Evaluation for Ensemble

In the research work carried out so far in the domain of machine learning for analyzing data, an ensemble machine learning aids in revamping the accuracy of the machine learning models. Accuracy is the prime evaluation metric in most of the research works [55, 56, 123, 146, 181, 205]. Some researchers also included area under the curve and error rate. However, limited work is carried out on multi-criteria evaluation of ensemble machine learning.

1.6 Problem Statement

Due to massive growth of data in last ten years, every business organization requires business insights. It helps them to improve their strategic as well as operational decisions. It has been realized that big data is not only about the size but other Vs like value and variety are also important. There is no threshold value that defines the size of big data. Big data is an opportunity to get valuable insights from the massive available data. Machine learning offers both computational statistics and automation-environment support for data analytics and is a promising field of research. Machine learning offers different techniques like classification, clustering, outlier detection, visualization etc. for getting valuable insights. In these different techniques, there is a need of vast research on special problems. For instance, if an imbalanced data arrived, how can we prevent our classifier from not getting biased. Such cases needs potential efforts. Similar is the issue of high dimensionality in big data. Most of the data collected in real time problems contains redundant information which requires trimming of unnecessary information during preprocessing. Apart from these problems, assuring the quality of data coming from different sources by removing noise, handling missing values and inconsistencies is also a serious problem. Traditional data mining techniques suffers from performance degradation issues. In recent years, an ensemble machine learning has become an established research field of machine learning. Hybridization method of ensemble is need to be employed. Feature selection techniques can be explored for handling high

dimensional problem as well for improving the performance of machine learning algorithm using subset of data. During ensemble building, other research issues like studying the impact of diversity during ensemble building is also important. However, very less research contributions are found in actual design and implementation of an Ensemble Framework on interdisciplinary problems of different domains like finance,wireless communications, bioinformatics, genetics, drug discovery, etc. For evaluating the machine learning algorithm performance, accuracy is used by researchers. Accuracy is misleading when data suffers from imbalance or other issues. Multi-criteria analysis of models is also required for complete validation. On considering these highlighted problems at various phases of data analytics, it appears that there is a dire requirement to review these machine learning techniques and algorithms for data analytics of massive data. Future efforts can be made in building an efficient and robust ensemble framework that could be useful for massive data applications of different domains like bioinformatics, finance, etc. The rationale of the research is to focus on further experimentation of the ensemble machine learning techniques and proposing a robust ensemble machine learning framework for extracting valuable insights from the data by offering efficient preprocessing features like efficient cleaning, feature selection, etc.

1.7 Thesis Objectives

This section presents the research objectives in this thesis research.

1. To study and review existing machine learning algorithms for data analytics.
2. To propose an ensemble machine learning framework which consists of features like preprocessing and classification etc.
3. To design and implement the proposed framework.
4. To test and validate the performance of proposed framework using various parameters like total time, accuracy, etc.

1.8 Thesis Contribution

In this thesis, an attempt has been made to solve the predictive-analytics problems using ensemble machine learning and multi-criteria decision making method. The main contribution of the thesis are done in several phases and they are as follows:

i A modified version of ensemble building machine learning framework with the name of Multi Criteria based TOPsis Ensemble (MCTOPE) is developed to solve the prediction problems with better predictive performance.

ii Data cleansing, class-balancing, feature selection, and feature importance of different features is carried out by "data preparator" in the MCTOPE using SMOTE, correlation coefficient, and random forest information measure, etc. Irrelevant and redundant features are removed during this process.

iii Different from the traditional methods of ensemble building, the proposed MCTOPE framework works on optimizing the overall performance of the system at the time of building an ensemble. The performance analysis is done using TOPSIS algorithm, a multi-criteria decision making technique on an entirely independent dataset called testing-dataset to avoid over-fitting. Random samples are generated, and the diverse machine learning models are employed to train the classifiers from the model-pool of ten different state-of-the-art classifiers. At the end of more than thousand iterations, and experimenting combinations of diverse machine learning models, a final ensemble with the highest performance score is employed for the prediction model.

iv To evaluate the performance of MCTOPE framework, the proposed framework is first validated on six different datasets from UCI machine learning data repository, and then we implement it on two case-studies of different domains i.e. drug toxicity prediction and fraudulent firm prediction.

v Different Vs of Big data like variety, veracity, and value are focused for predictive analytics in the thesis. Unstructured drug molecules data is collected for drug toxicity prediction case-study, validating the data using TOX21 database and then, prediction model is built to predict the toxic/no-toxic class of an unknown drug molecule. Simi-

larly, unstructured data of public firms is collected from an audit office for fraudulent firm prediction by validating the data by a team of auditors. The rationale is to built a web-application that can predict the fraudulent/non-fraudulent class of an unknown firm.

vi Drug Toxicity Predictor Testing: To validate the framework analytically, the molecular descriptors of three unknown drug molecules, namely nevirapine, delavirdine, and efavirenz, which play a key role in the AIDS therapy are used as testing drug molecules. The 100% correct prediction results serve as a proof of eligibility of the proposed framework to perform an efficient toxicity assessment task.

vii Fraudulent Firm Predictor Testing: To validate the framework analytically, data of new firms for next year audit is used for testing. A web-application is developed using R and Django Python framework that takes the input of important features and predict the probability of risk using ensemble model, working in the back-end. The test cases are developed and executed to check the fraudulent firm predictor for the firm.

The results obtained from the experiments have proved the usefulness of ensemble machine learning models for fraud prediction during audit planning, and toxicity prediction during drug design and development. Hence, contributing the research area of an external auditing and biological computing.

1.9 Thesis Organization

The rest of the thesis is structured as follows. Chapter 2 covers the literature review. The various techniques and challenges of big data analytics are reviewed thoroughly. As big data analytics is an emerging research area, different other research disciplines are also connected to the data anlaytics, which are also reviewed in the Chapter 2. The research contribution of machine learning techniques in the area of big data analytics is summarized. The role of machine learning techniques like classification, feature selection, clustering, outlier detection, and association rule mining are also discussed much in detail. The research contribution of researchers in the area of ensemble machine learning including, homogenous and heterogenous ensemble are also discussed. Besides, machine learning and big data analytics,

the review of multi criteria decision making methods are also presented in Chapter 2. The proposed framework is presented in detail in Chapter 3. The complete architecture of the proposed MCTOPE framework along with the layered view is also presented here. The detailed architecture of data preparator, prediction engine, machine learning model pool are also presented in much detail. The detail of the performance evaluation method for testing the performance of the proposed framework are discussed in the last section of the Chapter 3.

Chapter 4 discusses the design and implementation details including the detailed explanation of the case-studies. The design details are presented by UML diagrams. It also includes structural modelling and behaviour modelling of the proposed framework. The structural models are presented by class and component diagram. Behaviour models are presented by usecase diagram, sequence diagram, state-chart diagram and activity diagrams. The implementation detail includes the detail of experimental setup as well the implementation details of both the case studies. This chapter also presents the implementation detail of various optimization techniques like class balancing, feature importance, feature selection and ensemble modelling. The complete implementation detail of TOPSIS performance score is included much in detail. The implementation detail of both the case studies, namely, drug toxicity prediction and audit fraudulent firm prediction with complete explanation of the dataset is presented here.

Test and demonstration details of the proposed framework are presented in Chapter 5. The detail of the various datasets for the validation and testing of the proposed framework is summarized here. The complete detail of the K fold testing implemented for the two case studies namely, drug toxicity prediction and audit fraudulent firm prediction is also presented in this chapter. Test cases which are written and executed for testing the performance of proposed web-interface are presented in detail. The screen shots of positive and negative test-cases executed for testing the performance of the developed interface are also included in Chapter 5.

Finally, Chapter 6 concludes the thesis and points out the scope of further research.

Chapter 2

Literature Review

This Section reviews the research work of various researchers in big data analytics and machine learning. Various important research contributions in the field of machine learning, ensemble machine learning are presented in detail. Research gaps, problem formulation, and research objectives are also presented here.

2.1 Big Data Analytics

Data Analytics is a step by step procedure of converting the raw data into useful information for making important decisions [105]. At times, data analytics looks notably simple when viewed from this perspective, and comprehending the big picture of data analytics will make it more easier to understand. Data analytics involves all the mathematical techniques of interpreting data, procedures of applying mathematical and statistical approaches for analyzing the data and planning easier ways of interpretation [272]. The process of data analytics is generally described by a simple Equation [105]:

$$Data = Model + Error \tag{2.1}$$

The collected data is generally complex and hard to communicate. Data model represents a compact description of data. It is much easier to understand, to build theories, and to make predictions. The data model complexity increases proportionally to the number of

parameter to be predicted. The error is depicted by the amount to which the data model is unable to classify or predict the unknown parameter. Lastly, our goal of data anaytics is to build a model which makes an error as small as possible. Researchers have successfully applied data analytics in different domains. For instance, data scientist have implemented Earth Science Data Analytcs (ESDA) to glean more knowledge about earth [110], to analyze social networking data [4] and in academics to analyze student success within a course [7].

Big Data Analytics is about examining massive datasets to identify hidden patterns, real time insights, unknown correlations and hidden novel patterns. The process includes data acquisition from various sources, storing it temporarily or permanently, and then processing and applying analytics to get the desired results. For data analytics, various techniques and algorithms are applied. On the basis of its usage, the research in data analysis can be classified into four divisions as summarized in the Table 2.1. These days business organizations are focusing more on predictive and presciptive analytics. It is used for better business decision making whether it is strategic or operational decisions resulting in an effective marketing, customer satisfaction and ultimately increasing revenue [219].

Table 2.1: Types of Big Data Analytics [220]

SNo.	Types of Analytics	Description
1.	Descriptive Analytics	What is happening?
2.	Diagnostic Analytics	Why did it happen?
3.	Predictive Analytics	What is likely to happen?
4.	Prescriptive Analytics	What should I do about it?

2.1.1 Challenges in Big Data Analytics

This section presents the various challenges faced by researchers in the field of big data analytics research.

2.1.1.1 Data Deluge

It is the considered as the situation where massive volume of data is generated at high speed causing the flood of data which is challenging task to manage. We live in 21st century where everything is data driven growing exponentially and this data generally unstructured like videos, images, text, etc. This data is exponentially growing and due to the easy availability of smart phones, mobile data is increasing at tenfold rate.

According to one Cisco report, mobile data has grown around 30 ExaBytes in 2014 and it will shown tenfold rise by 2019 as shown in the Figure 2.1. So, it is challenging task for researchers to manage this big data flood tactfully using efficient tools so that it is available to the decision makers whenever required [71, 124]. The most common design paradigms for processing huge volume of data are:

i Batch Processing Model

ii Real Time Processing Model

iii Hybrid Processing Model

As shown in the Figure 2.2, batch processing model deals with the issue of processing large volume. Real time processing issue addresses the issue of processing dynamic nature of data, i.e. velocity. This is also called stream data processing.

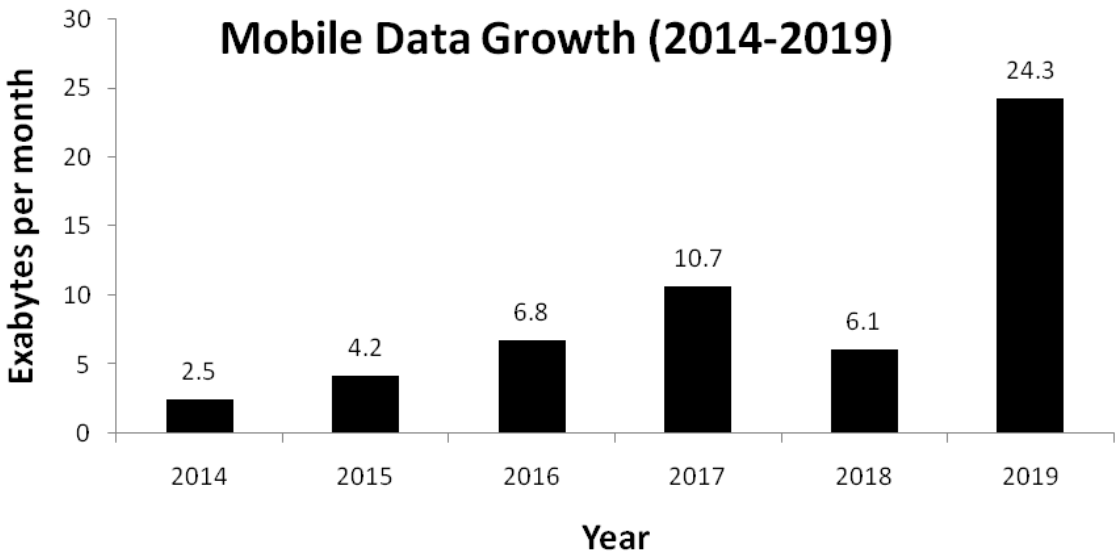

Figure 2.1: Ten Fold Global Mobile Data Growth from 2014-2019 [124]

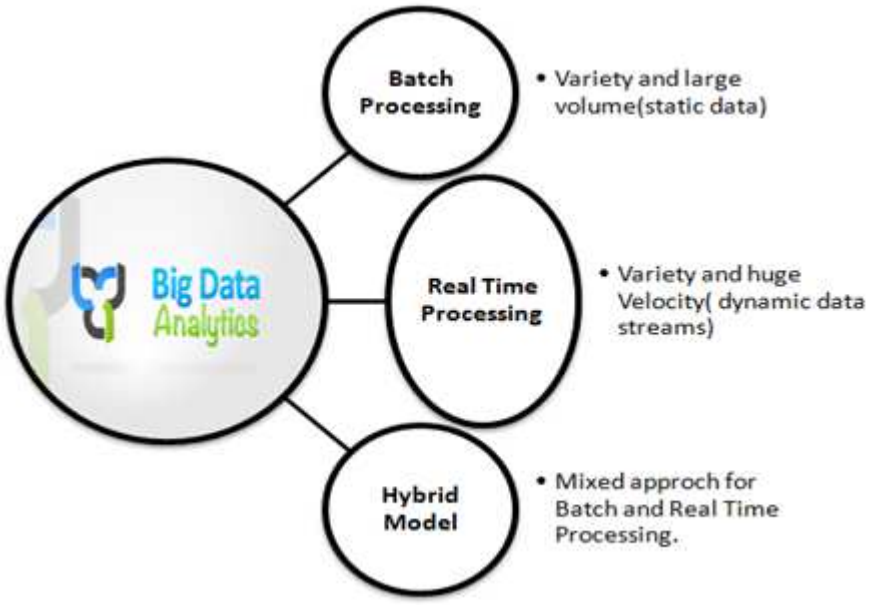

Figure 2.2: Big Data Issues and Solutions [156]

Hybrid Approach deals with both the issues of volume and velocity. Hence, it is responsible for combining the results of both batch processing and real time processing. Starting from 2003, when Google first published its white paper on Google File System and Map reduce framework [156].

In the old days of 2006, business organizations did not face much severe problems of big data. It was the time when big data problems were just newly discovered which led to the arrival of Hadoop and Map Reduce Framework solution. So, it is known as the first generation of Big Data where Hadoop gave well grounded solutions for batch processing data. With the advent of real-time processing in 2010, Yahoo came with S4 technology solutions. It is a general purpose and distributed platform for processing streaming data. Other big organizations like Google developed Millwheel for this purpose. Linkedin came with the development of Samza for processing data streams.

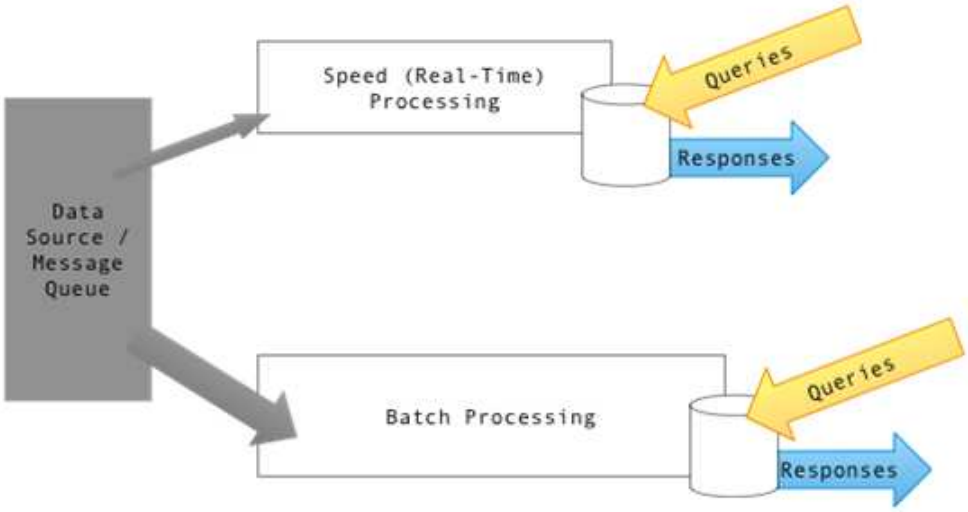

Figure 2.3: Lambda Architecture [156]

The Hybrid Model focuses on Lambda Architecture in 2012 as shown in Figure 2.3 which is implemented to handle massive data set by using both batch processing and stream processing model [156]. Database is an indispensable part of any business organization and big data is more demanding in terms of its requirements. As the data is not only huge but also not well structured hence it cannot be easily processed by traditional databases. Such critical demands led to the emergence of variety of databases known as NoSQL databases. Table 2.2 shows the various technologies that were arrived for various dimensions like processing paradigms, databases, etc.

Table 2.2: Key technologies of different dimensions in Big Data [220]

Dimensions	Technology
Batch Processing	Hadoop Sqoop Pig Hive Cascading Spark
Real Time Processing	Flume Kakfa S4 Storm Samza Spark Streaming
Hybrid Computation	Lambdoop Summingbird
NoSQL Systems	Hbase Redis MongoDB Neo4j Cassandra

2.1.1.2 Big Data Capture, Transmission and Storage

Big data volume is exponentially growing because of the increase of data gathering sources like mobile devices, sensor technology, camera, wireless network devices, etc. Quintillion bytes of data is being produced by these devices everyday. But, the challenge is to store the data flood. Many organizations destroy the valuable historical data because of the unavailability of required storage. Various underlying storage technologies like Solid State

drive (SSD), Phase Change Memory (PCM), Network Attached Storage (NAS), Direct Attached Storage, etc are supporting the requirement but much more innovative advancement is needed in this field [50].

Researchers are shifting towards cloud computing for online storage [155]. Cloud data storage is supporting on demand service model for solving the storage issues of high volume of data [54]. By conducting more than forty experiments while operating two case studies, researchers compared cloud and non-cloud systems for big data storage [276]. It is observed in the experiments that the execution time for storing big data on cloud systems are lower than the execution time on non-cloud systems. Additionally, cloud systems offer more improvements in efficiency as compare to non-cloud systems [276]. Although cloud systems support big data storage but they also suffer from several serious issues like network bandwidth capacity, data integrity, scalability, privacy, etc [54, 178]. To enhance the profit and qualify of service in a Cloud Federation Environment, policies are proposed in a research [41]. In recent years, nanophotonics enabled optical devices have also encouraged researchers to work on more disruptive methods to enhance the capacity of current optical memory for future big data storage [82].

2.1.1.3 Big Data Curation

The rationale of data curation is to ensure quality assurance for knowledge discovery. It also supports other important factors of data management like authentication, retrieval, preservation, data representation, etc. Because of the rise in data complexity,the existing database management tools lack the efficient management of big datasets. Efficient database systems like data mart and data warehouses are available for managing and analysis of structured datasets and lack heterogeneous data curation activities [50]. But, pre-processing like cleaning, feature extraction, data reduction is necessary and challenging task for such volume of heterogeneous data before the data is sent for real time analytics [50, 186]. Testing the quality of data and compliance testing is inevitable phase before utilizing big data for data analysis [199].

2.1.1.4 Accelerating Data Analysis Algorithms

The major problem while performing the analysis tasks of big data is its large volume which leads to the scalability issues [50]. Researchers are now paying great attention to accelerate the performance of analysis algorithms when dealing with large volume of data coming with high velocity [54, 75]. This needs the development of various methods like sampling, multi-resolution analysis methods, etc. Machine learning, a subfield of artificial intelligence offers incremental and ensemble approaches to deal with the above challenge. Even the clock cycle frequency of processors doubles as it follows the Moore's Law but it still lags behind when compared with velocity with which big data is scaling. So, performing analysis of real time streams is still another challenge in this area [50].

2.1.1.5 Big Data Visualization

Data visualization focuses on representing the knowledge using graphs. There are many solutions proposed by researchers to overcome the challenges faced in visualizing large scale data using graphs [222]. Many companies like Ebay and Tableau use data visualization tools to understand their sales and the taste of billions of customers [50]. Researchers are working on advanced methods of visualization for solving the issues related to high dimensional heterogeneous data. For instance, input space histograms seem to be really helpful for visualization of hierarchical clusters in big data [111]. But, it is specifically very challenging task to represent large and complex data streams for visualization tasks. Even the advanced visualization tools suffers from low scalability and poor response time issues. Data Visualization demands new and innovative frameworks for visual analytical process of big data [175].

2.1.2 Research Disciplines in Big Data

Big data techniques involve innumerable disciplines like statistics, machine learning, etc [50]. Numerous techniques offered by the disciplines like data analytics, statistics and data mining overlap each other. Similarly machine learning utilizes countless algorithms of statistics and data mining. There is still some distinctions which can be observed and are presented

in the Figure 2.4.

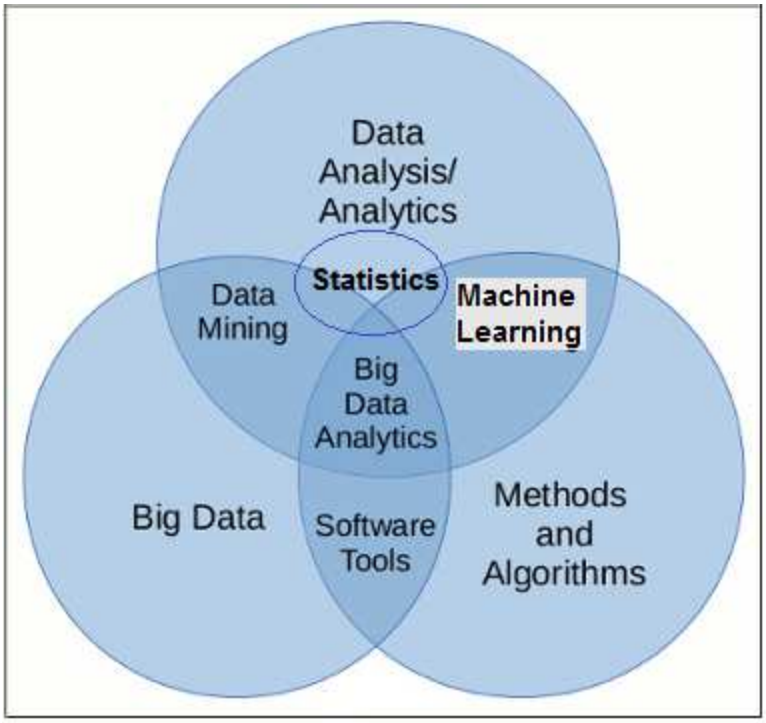

Figure 2.4: Disciplines connected with Big Data Techniques [50]

Statistics is the science works on the interpretation of complex data. Although, many statisticians around the globe are doing research in this field to solve different social and economic problems using innovative tools and techniques but there are certain pitfalls which cannot be ignored [255]. One major drawback is to move directly towards the target and not understanding the underlying phenomenons in the data. Most of the techniques work on summarizing the information and ignore the critical pre-processing issues like noise. The principles of statistics aims at obtaining much reliable information from the available data by finding relations between different features using numerical descriptors in the data [112, 255]. Although research is going in the field of big data for improving the performance of standard statistical methods but much more innovative research is needed to fulfil the complex demands in this area.

Data Mining aims at discovering unknown patterns in the data which ultimately helps any organization in their critical decision making process [219]. Data mining research moves much towards descriptive analytics to know the hidden patterns in the data.

Machine Learning is a branch of artificial intelligence which employs statistics and data mining algorithms to build models, focusing more on the predictive analytics. The aim is to set up an automated environment for predicting important factors related to business decision making tasks [177]. Unlike statistics, researchers don't work much on understanding data and writing programs. The research goal of machine learning is to train the machine learning models from training examples for making future predictions.

Instead of explicit programming instructions only, machine learning (ML) applies scientific algorithms to the collected data with the goal of creating automated environment for making predictions or important business decisions . The rationale of machine learning is to solve and automate more complex task by training the machine. Unlike statistics, the goal is not the understanding of problem more deeply to write a program but machine learning runs machine learning model algorithms in order to train the algorithm from the training examples.

Although, most of the learning algorithms are statistical in nature, but the aim is to work on the prediction performance and not to dig the statistical insight. Different free and proprietary softwares are present in the market to explore data-analytics using machine learning as presented in the Table 2.3.

Table 2.3: Free and proprietary softwares in the market to explore data-analytics using machine learning

S No.	Tool	Description	License	Ref.
1	Carrot2	Clustering framework	Free and open-source	[258]
2	ELKI	Outlier detection tool	Free and open-source	[28]
3	GATE	Natural language processing tool	Free and open-source	[218]
4	KNIME	Data analytics framework	Free and open-source	[225]

Continued on next page

Table 2.3 – continued from previous page

S No.	Tool	Description	License	Ref.
5	Massive Online Analysis	Big data stream mining tool	Free and open-source	[5]
6	ML-Flex	Allow user to intergrate third party machine learning packages	Free and open-source	[216]
7	MLPACK	Offers ready to use machine learning algorithms	Free and open-source	[232]
8	MEPX	Regression and classification problem tool	Free and open-source	[230]
9	NLTK	Statistical natural language processing tool	Free and open-source	[240]
10	OpenNN	Neural network library	Free open-source	[239]
11	Orange	Data mining and machine learning software	Free and open-source	[243]
12	R studio	Programming based tool for statistics, data mining and machine learning	Free and open-source	[269]
13	Scikit-learn	Library for machine learning for programs in Python	Free and open-source	[226]
14	Torch	Deep machine learning library	Free and open-source	[270]
15	UIMA	Unstructured information management component framework	Free and open-source	[274]
16	Weka	Data mining, machine learning algorithms suit	Free and open-source	[279]

Continued on next page

Table 2.3 – continued from previous page

S No.	Tool	Description	License	Ref.
17	Angoss Studio	Data mining tool	Proprietary	[6]
18	SPSS modeler	IBM data mining software	Proprietary	[266]
19	Micosoft Analysis Services	Data mining software by Microsoft	Proprietary	[263]
20	Netowl	For multilingual text mining	Proprietary	[238]
21	Oracle Data Mining	Data mining software	Proprietary	[18]
22	Rapid Miner	Data mining and machine learning software	Proprietary	[231]
23	SAS enterprise miner	Data mining software	Proprietary	[261]
24	Statistica	Data mining software.	Proprietary	[267]

2.2 Machine Learning

Machine Learning techniques are applied depending upon their ability to find solutions of numerous problems. On the basis of problem, literature review is divided further as shown in the Figure 2.5. Classification is a supervised learning technique in which classes are defined first and then classifier is trained by utilizing the training samples. The goal to get the ability of predicting the objects according to classes of sample data. Prediction involves finding future values based on the patterns found in the data sets. Traditional supervised machine learning focuses on implementation of only single machine learning model.

Ensemble machine learning is introduced to integrate multiple models of machine learning. Homogeneous ensemble combines same models multiple times with different training

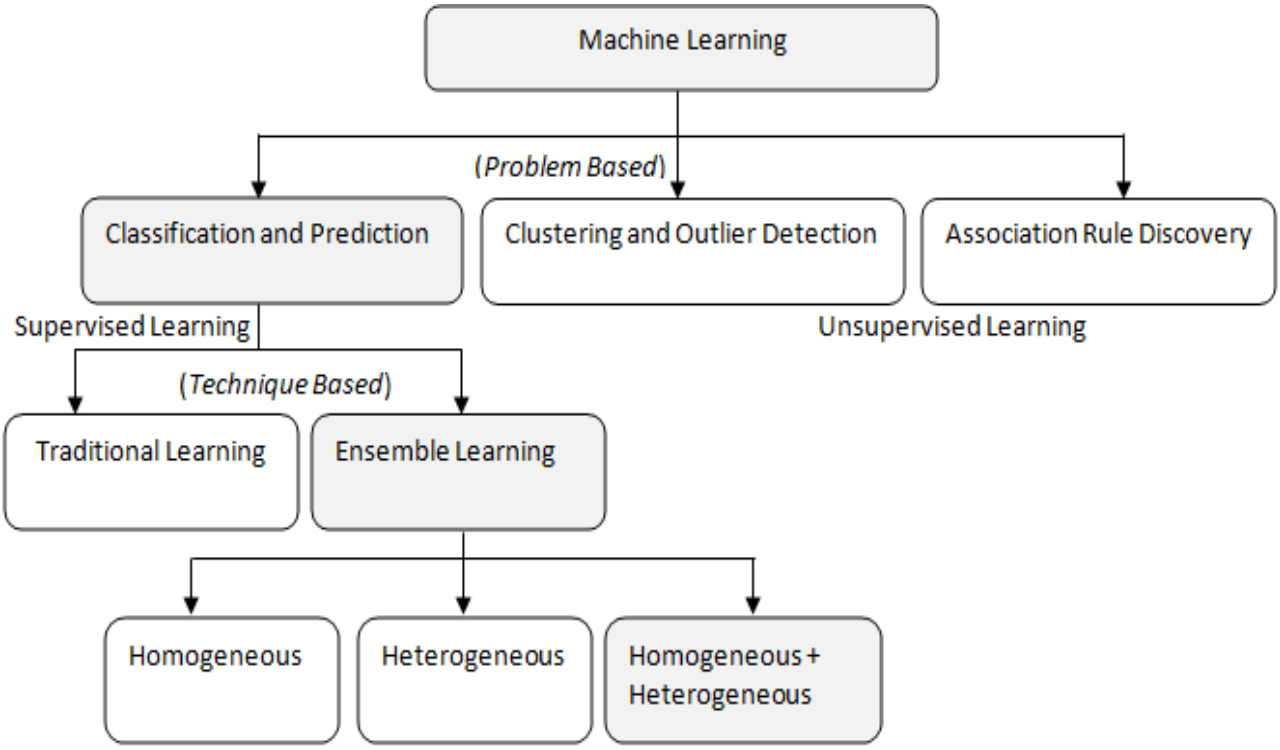

Figure 2.5: Machine Learning Techniques

datasets. Heterogeneous ensemble prefers diverse models integration. Clustering and association rule discovery is an unsupervised learning technique. In clustering, input data does not have known results. Here the model is prepared by detecting any kind of structure in the input data. Association rule learning approach is a technique for discovering hidden and interesting relationship among different features in the data [44]. This section reviews the work of researchers to solve different problems using machine learning techniques for mining valuable data sets in last few years.

2.2.1 Research Trends

Researchers around the globe are working significantly to achieve the big data objectives using machine learning. There are different areas where researchers are focusing tremendously. For instance, the every V of big data including volume, velocity, variety, veracity, and value have research issues which are need to be worked upon as presented in the Table 2.4. Additionally, the research trends from other perspectives like whether they are related to data mining or technique integration are depicted in the Figure 2.6 and explained below

[101]:

i Data mining:

There is heterogeneous data available for mining and research and can be collected from diverse sources. Such data demands effective quality assurance techniques to get the fruitful results from the machine learning. Although number of technologies based on associative rule discovery, ontology, and semantic web offer good results for mining and analysis of big data but much more mature and scalable methods are needed in this field.

ii Pattern training:

Machine learning works on training the machine learning models by utilizing the training samples. Most of the times, as the number of training samples increase, accuracy of the model get improved. Although labeled training data greatly help in training the model but is it increases the computational cost specifically when the data is incremental and changing with high velocity. So, there is always a trade-off between cost and accuracy. Hot research is going on how to choose the training samples without causing over-fitting of the data.

iii Technique integration:

It is another critical research issue of how to integrate other fields of research like cloud computing, data mining, etc. with machine learning for extracting valuable information from large scale complex data.

iv Privacy and security:

It is another matter of concern of how the researchers are utilizing the private and personal data of business organizations. So, efficient methods of machine learning should be applied for preserving the privacy of business organization along with protecting the personal information of the individuals in the organizations. It is an important research issue of getting the valuable information from the massive data using machine learning with the guarantee of privacy and security. For instance, multi-party privacy preserving protocols have been designed for social network analysis [187]

v Realization and application:

The ultimate objective of research in machine learning and big data is to support and help the people in the society. Although theoretical research has achieved many milestones but still there is great need to focus more on applying these techniques on actual problems in the environment.

Table 2.4: Open research problems in Big Data Analytics

No.	Big Data V	Open research issues
1	Volume	Storage, capture and transmission [50, 197] Cleaning massive data [50, 125, 151] Compressing massive data [197, 252] Feature selection for high dimensional data [51, 51, 125, 151, 252, 252]
2	Velocity	Flow management [51, 125, 151, 252] Learning of streaming data [50, 197]
3	Variety	Dealing with unstructured/semi-structured data [50, 197] Learning heterogeneous data [51, 125, 151, 197, 252]
4	Veracity	Assessing data varacity [51, 125, 151, 252] Learning with unreliable or contradicting data [51, 125, 151, 252]
5	Value	Machine learning module for decision support [50, 51, 125, 151, 252]

2.2.2 Feature Selection

The biggest challenge of big data analysis is processing distributed data of various data sources. Feature Selection is beneficial pre-processing step to solve the challenges related to the high dimensionality of data [186]. It focuses on harnessing multiple sources of data (in sequence or concurrently) and reduces the problem dimension. V. Boln-Canedo et al. (2015) highlighted the ongoing research contributions related to feature selection of high dimensional data.

Cheng et al. (2017) and reviewed open research issues in developing feature selection techniques for large scale data [98, 98]. In another research work, along with addressing the

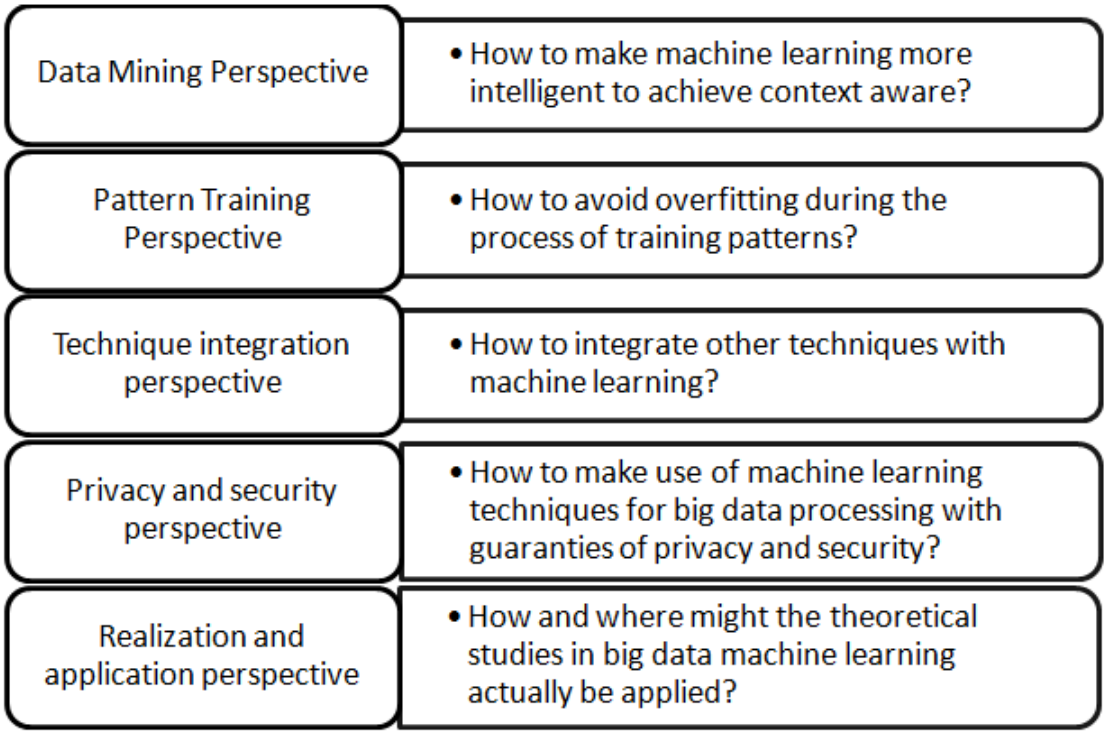

Figure 2.6: Research trends in machine learning and data analytics

challenges in feature selection, Huan Liu et al. (2017) presented open source repository for feature selection in detail. It describes scikit-feature repository of popular machine learning algorithms [97]. In their review, they observed that the most frequently used methods are found to be Chi-squared Univariate Ranker, F-score (Fisher score), Information gain, ReliefF, mRMR, SVM-RFE, CFS , FCBF, INTERACT, etc [186].

J. Paul et al. (2015) proposed kernel methods for heterogeneous feature selection. They implemented ontology based classifier framework for extracting ontological patterns in the pool of big sensory data sets [150]. Special efforts were made by John A.Lee et al. (2015) by not disturbing the global and local structure of data while reducing it's dimensionality. Refinement of Stochastic neighbour embedding (SNE) method is proposed for improving the quality of dimensionality reduction [126]. J. Fellus et al. (2015) proposed and implemented an asynchronous and decentralized PCA algorithm for reducing dimensionality of big data sets [76]. Zhou et al. (2015) implemented an iterative learning stacked-extreme machine learning algorithm that employs the PCA dimension reduction method for large unstructured data [213]. Y. Kima et al. (2015) developed an ensemble feature selection methods by

combining three regression model to select significant features in massive data of spectral fingerprints [113]. Few researchers employed feature selection techniques on the top of hadoop framework for big data analysis [60, 182].

2.2.3 Classification and Prediction

Classification is a supervised learning technique in which classes are defined first and then chosen classifier is trained by utilizing training samples. This classifier is used for classifying the objects according to classes of sample data for predictions in future. Prediction involves finding future values based on the patterns found in the data sets. Predictive analytics has drawn a remarkable interest of researchers in every domain to extract value from big data sets for decision making. Many researchers have applied classification and regression models for predictions in different domains like in networking for predicting number of zombies in DDoS attack [134]. Alberto et al. (2017) reviewed various challenges related to classification of big datasets [33]. Example: Predictive models are used for credit scoring to predict whether the loan application should be accepted or not.

Moeyersoms et al. (2014) reviewed a case study of energy company for churn prediction. The study proved that although the availability of socio demographic information of customers had increased the dimensionality of big dataset but they could be very useful when employed in predictive data modelling [100]. Hongfei et al. (2013) introduced a case study of predictive modelling using machine learning techniques by processing historical and real time datasets for railway company. The predictive analytics was applied to address the rail network velocity, maintenance and failure issues [120].

McAfee et al. (2012) reviewed the success stories of PASSUR airlines and Sears Holding who were drifted towards predictive analytics of datasets to make their promotions more precise and quick [38]. Liu et al. (2013) presented a framework for sentimental mining. They implemented naive bayes efficient classifiers on the top of Hadoop framework [48]. Ayma et al. (2015) demonstrated the potential of ICP data mining package tool. Four different machine learning classifiers namely, SVM, naive bayes, decision tree and random forest were integrated on the top of Hadoop Map Reduce Framework for processing huge volume of data [185]. Cavallaro et al. (2015) implemented parallel SVM for the classification of remote

sensor data. Besides implementing machine learning techniques for improving prediction accuracy, need of parallelization for processing big data sets to speed up the total processing time was also demonstrated [77].

Lopez et al. (2015) introduced Linguistic Fuzzy Rule Method to deal with 7V's problems during classification of big data sets. The model was implemented with the goal of producing best and fast prediction results under distributed infrastructure of Hadoop Map Reduce framework [188]. By conducting a case study, Zamil et al. (2014) presented a smart irrigation system by proposing an Ontology Based Classifier framework . The idea was to capture the huge heterogeneous sensory dataset, integrating various knowledge domains and extracting ontological patterns in the pool of big sensory data sets [42]. Wang et al. (2015) presented a Sparse Online Learning Classification framework to handle the problem of high dimensionality in big data stream classification [62].

Bechini et al. (2016) proposed map-reduce solution for association and classification of big data by developing a variant of FP growth algorithm [29]. A variant of KNN algorithm (lazy learning) is proposed to handle scalability issues of large scale datasets [209]. A medical imaging data is used to implement the proposed method. Salehan et al. (2016) developed a scalable sentimental mining based automated system for sorting and classification of big data. The data used for research is large OCR dataset to help the consumers and vendors [135]. Recently (2017), fuzzy based methods on the top of Hadoop framework are proposed by researchers to solve big data classification problems [32, 171]. The review of some important contributions for classification problems in this field are summarized in the Table 2.6.

Table 2.5: Review of classification problems in Big Data Analytics using Machine Learning Methods

S No.	Domain	Model	Description	Ref.
Continued on next page				

Table 2.5 – continued from previous page

S No.	Domain	Model	Description	Ref.
1	Sentimental Analysis	Naive Bayes	A sentimental mining system for processing massive datasets is implemented on the top of Hadoop framework. Naive Bayes Classifier is employed for obtaining scalability and achieving 82 percent accuracy	B. Liu et al. [48]
2	Networking	Ensemble Extreme Learning Machine (ELM).	Ensemble ELM is proposed to revamp the training phase up to 4.6 times and reduces the test errors by 19 percent when compared with single ELM classifier performance.	X. Wang et al. [198]
3	Social Network Analysis	Parallel structural clustering algorithm (PSCAN)	Parallel clustering was implemented. Hubs and outlier of Twitter social network with billion of edges are detected.	W. Zhao et. al [194]
4	Cloud Computing	Adaboost ensembles	Developed a multi tiered ensemble based method HS Miner to handle huge volume and drifts while classifying the big data streams.	A. Haque et al. [35]

Continued on next page

Table 2.5 – continued from previous page

S No.	Domain	Model	Description	Ref.
5	Education	Neural network, decision trees and SVM	Sparse Online Learning Classification framework to handle the problem of high dimensionality in big data stream classification.	D. Wang et al. [62]
6	Cloud Computing	SVM	Classifying high dimensional data sets by distributing redundant features to several classifiers during ensemble construction. Prediction accuracy is improved by using ensembles.	Haque et al. [35]
7	Signal Processing	Window Adaptive Ensemble algorithm	Implemented Window Adaptive Ensemble algorithm on the basis of Online Accuracy Updated Ensemble (OAUE) for processing big data sets. Problem of sudden drifts in big data streams is solved.	Gu Xiao-Feng et al. [83]
8	Cybernetics	Nearest Neighbour Rule Based Ensemble	Proposed sub sampling based ensemble by majority voting method to speed up the computation of single classifiers.	B. Krawczyk et al. [47]

Continued on next page

Table 2.5 – continued from previous page

S No.	Domain	Model	Description	Ref.
9	Image Processing	Naive Bayes, Decision Tree, SVM and Random Forest	The potential of ICP data mining package tool for automatic image interpretation framework is implemented.	V. Ayma et. al [185]
10	Wireless Sensors	Parallel SVM	Classification framework of remote sensor data is implemented for improving prediction accuracy. Parallelization is achieved with parallel SVM to significantly reduce the processing time for big data sets	G. Cavallaro et al. [77]
11	Fuzzy Systems	Linguistic Fuzzy Rule Method	The model is implemented with the goal of producing best and fast prediction results under distributed infrastructure of Hadoop Map Reduce framework. All seven V's issues are handled by model.	V. Lopez et. al [188]
12	Face Recognition	Ensembling of linear classifiers	Gender classifier framework by using four million face images having more than 60,000 features. Prediction accuracy of facial patterns is improved.	S. Jia et al. [170]

Continued on next page

Table 2.5 – continued from previous page

S No.	Domain	Model	Description	Ref.
13	Ontology	Ontology Based Classifier framework	Smart Irrigation System is proposed to capture the huge heterogeneous sensory dataset, integrating various knowledge domains and extracting ontological patterns.	A. Zamil et al. [42]
14	Cloud Computing	Ensemble classifier framework	The framework is proposed for workload classification in cloud based big data applications and predicting the next workload in the virtual environment.	Cuzzocrea et al. [57]
15	Medical Imaging	KNN	K Means algorithm is used to cluster the large scale medical data and KNN is applied for classification.	Z. Deng et al. [209]
16	Performance Check	Gaussian Mixture, Logistic regression, Random forest	Comparing performance of classifiers with map reduce.	K. Nishchal et al. [108]
17	Sentimental Analysis	Naive Bayes Classifier	Improving the performance of classification when datasize increases.	B. Liu [48]

2.2.4 Clustering and Outlier Detection

Clustering technique is the part unsupervised learning in which classes are not pre-defined. The objects are conceptually divided into meaningful groups called clusters or data seg-

ments. Data samples that are much similar with each other form a cluster. Researchers also employed clustering for targeting advertisements for online social networks [244, 245]. Most common clustering technique is K-Means.

Govindarajan (2013) implemented K Means algorithm in distributed Hadoop Map Reduce environment for clustering the continuous data of students from online learning activities based on their competencies and learning habits [107]. Tsapanos et al. (2015) implemented kernel matrix based trimming algorithm for revamping the performance of kernel K means method. Distributed environment was set up with the help of map reduce framework to improve the clustering performance for big data sets [142]. Huag et al. (2014) reviewed the applications of extreme learning machine (ELM) algorithm in various domains like computer vision, robotics, biomedical engineering, etc. Empirical studies in various fields shows that apart from regression and classification, ELM is preferred over SVMs and other deep learning algorithms.

Clustering also helps in outlier detection, which is the technique to discover the data objects which have low similarity from the remaining data. Such objects are named as outliers. Zang et al. (2014) proposed an outlier detection to extract anomalies in big network data streams. Adaptive Stream Projected Outlier Detection method is implemented for KDD CUP anomaly detection application to revamp the efficiency and scalability of technique on large datasets [104]. Zhao et al. (2013) implemented parallel structural clustering algorithm (PSCAN) in the distributed Hadoop Map Reduce framework environment. The algorithm was implemented for big network like Twitter social network with billion of edges to detect hubs and outliers [194]. Wang et al. (2013) introduced an anomaly detection algorithm for big datasets using wavelet packet transform and control theory of statistics. In order to handle the storage complexity and save the computation cost for processing big datasets, wall human detection technique was applied using compressed data [193].

Current researchers are using it for clustering and feature selection of real time processing of big datasets [80]. But due the ease of availability of Global Positioning System (GPS) , remote sensors and other satellite technologies, the requirement of applying analytics on trajectory data comes into picture. Dang et al. (2014) introduced a scalable TRA-POPTICS algorithm. The clusters are constructed for trajectory data. The algorithm proved a perfor-

mance improvement of POPTICS algorithm for clustering real time data processing [208]. Recently (2017), most of the researchers are employing more efficient clustering techniques on the top of Hadoop framework in order to implement a scalable version of the clustering algorithm. For instance, Hdk algorithm is proposed by implementing K means algorithm on the top of hadoop framework [40, 164].

Although innumerable research have favoured K means clustering algorithm, but it requires to define the number of clusters initially. So, GHSOM (growing hierarchical self organizing maps) is developed by Chui Hui et al. (2017) to overcome the issues of K means algorithm for big data clustering [16]. Similarly, Bayesian hidden Markov model (HMM) is ensembled with Gaussian Mixture (GM) Clustering to get the scalable version of clustering big data [81].

Table 2.6: Review of Clustering problem in Big Data Analytics using Machine Learning Methods

S No.	**Domain**	**Model**	**Description**	**Ref.**
1	Education	K Means clustering	Clustering the continuous data of students from online learning activities based on their competencies and in distributed Hadoop Map Reduce environment. Learning habits of online students are detected.	K. Govindarajan et al. [107]
2	Big Data Computing	K means, PCA	Projective clustering is performed for huge volume of data by employing merge and reduce approach.	D. Feldman [59]

Continued on next page

Table 2.6 – continued from previous page

S No.	Domain	Model	Description	Ref.
3	Big Data Computing	Review	Comparing performance of clustering algorithms by measuring parameters like total time, stability, and scalability.	A. Fahad et al. [31]
4	Big Data Computing	K means	K means algorithm is employed on the top of big data using map reduce.	X. Cui [196]
5	Education	TRA-POPTICS	Introduced a scalable and fast algorithm for clustering trajectory big data for real time processing for performance improvement of POPTICS algorithm	Govindraj et al. [208]
6	Pattern Recognition	Kernel K-means	Distributed environment was set up with the help of map reduce framework to improve the clustering performance of kernel K means algorithm.	Tsapano et a. [142]
7	Medical Imaging	K-means	K Means algorithm is used to cluster the large scale medical data	Z. Deng et al. [209]

Continued on next page

Table 2.6 – continued from previous page

S No.	Domain	Model	Description	Ref.
8	Cloud Computing	C means	Weighted C means algorithm is employed using Taylor Theorem and it is combined with privacy preserving schemes for offering security of raw data on cloud.	Q. Zhang [152]
9	Health Care	Bayesian hidden Markov model (HMM), Gaussian Mixture (GM)	HMM and GM are combined with each other for clustering large scale genome data. Results are compared with effective segmentation methods like pruned linear method, binary clustering, etc.	G. Manogaran et al. [81]
10	Big Data Computing	Artificial Bee Colony	ABC algorithm is employed using Map Reduce program in Hadoop environment is order to optimize the time and revamp the classification accuracy for clustering large scale data.	S. Ilango et al. [168]

2.2.5 Association Rule Learning

Association rule learning approach is a technique to discovering hidden and interesting relationship among different features in the data [44]. Most of research utilized the Apriori algorithm for generating rules among different features in the data [44]. R. Dehkharghani et al. (2014) extracted valuable information from the large scale data of a social networking website of Twitter. The data is extracted for summarization of Kurdish political problem in

Turkey. In order to analyze the data, knowledge of sentimental analysis and algorithms of association rule mining are combined [157].

Czibula et al (2014) proposed a method of predicting software defects by employing association rules [78]. C. Tew et al. (2014) applied association rule mining for behavior analysis of 61 different datasets. Different interestingness measures are developed and implemented for validating the domain knowledge [53]. For the fulfil the demand of quality assurance during software maintenance phase, association rules are discovered for prediction of defect in software module. The study is implemented on NASA datasets. Soysal et al. (2015) addressed the challenge of extracting useful information from structured traffic incident data by employing heuristic method. Without the need of searching the entire lattice and pruning method, the proposed method extracted useful information with lesser computational cost when it is compared with other state-of-the art techniques [265].

F. Kargarfard et al. (2015) worked on generating association rules for implementing an expert system for influenza prediction with an accuracy of 99.58. A software is proposed for this purpose with the name "Prediction of Pandemic Influenza [73]. Similarly, association rule technique is used to develop a music recommendation system by Saifur et al. (2016) [136]. This expert system recommends the next song to the user on basis of their choices in the previous played music. X. Yuan (2017) presented a new variant of Apriori algorithm to remove it's major bottle for processing large datasets [280]. This is achieved by employing a mapping technique for database which would avoid recursive scans of database. Along with this, further improvement in efficiency is achieved implementing by overlapping and pruning methods.

2.3 Ensemble Machine Learning

Ensemble technique is employed to enhance the prediction accuracy of model by combining many algorithms together. Researchers around the globe are working on efficiently applying ensemble machine learning algorithms for modeling prediction and analytics problems. Dietterich et al. (2000) reviewed various ensemble methods like Adaboost, Random Forest etc in his studies. It is uncovered in the experiments that ensemble methods works superior

than any single classifier in terms of prediction accuracy [79]. Street et al. (2001) presented an ensemble solution for large scale streaming data classification. Besides parallelizing the algorithm, different blocks of data is used for different classifiers for effective ensembling [192]. Feng et al.(2014) implemented Window Adaptive Ensemble algorithm on the basis of Online Accuracy Updated Ensemble (OAUE) for processing big data sets. The solution was provided to solve sudden drifts issue in big data streams [83].

Krawczyk (2015) proposed sub sampling based Nearest Neighbour Rule Based Ensemble method for applying analytics on big datasets. The outputs of classifiers were integrated by majority voting method to speed up the computation [47] . Jia et al. (2015) presented a gender classifier framework by using four million face images having more than 60,000 features. Ensemble method of various classifiers were used to improve the prediction accuracy of facial patterns [170]. Haque et al. (2014) developed a multi tiered ensemble based method HS Miner to handle huge volume and drifts while classifying the big data streams. To achieve scalability and effectively speeding up the procedure, three large Adaboost ensembles were presented using Map Reduce based parallelism [35]. Piao et al. (2014) presented an ensemble method using Space Vector Machine algorithms (SVM) to handle high dimensionality problem while classifying big data sets. The result proved that prediction accuracy for high dimensional data can be improved further if the redundant features are distributed to several classifiers during ensemble construction [201]. Cuzzocrea et al. (2015) proposed an ensemble classifier framework to solve workload categorization problem in cloud based big data applications. The framework was implemented for workload classification and predicting the next workload in the virtual environment [57]. Wang et al. (2013) proposed an Ensemble Extreme Learning Machine (ELM) to speed up the training phase up to 4.6 times and reducing the test errors by 19 percent [198].

Diversity among candidate classifiers for ensemble is a crucial issue of ensemble building and now has become an overgrowing research area [52, 121, 161, 173, 174]. Although, many variants of ensemble integration have been proposed by different researchers but no technique is yet defined to be the best [162]. Different techniques contributed by researchers for maintaining the diversity of ensemble have been discussed here.

Ensemble method offers integration of multiple classifiers tactfully. Homogeneous en-

semble employs single learning method using different sub-samples of training data, whereas heterogeneous ensemble combines different classifiers on the same training sample. Different researchers revamp the performance of classification and prediction to obtain the variants of integration as summarized in the Table 2.7.

Table 2.7: Research Contribution in Homogeneous and Heterogeneous Ensemble Building

S No.	**Ensemble Technique**	**Base Model**	**Domain**	**Evaluation Parameters**	**Ref.**
1	Homogeneous	Weighed neural networks	Mixed	Relative error, bias , variance	I. Maqsood et al. [91]
2	Heterogeneous + Homogenous	Nave Bayes classifiers and SVM	Text categorization	F1 measure	Yan-Shi Dong et al. [202]
3	Homogeneous	Neural Network	Mixed	Accuracy	Hyun-Chul Kim et al. [90]
4	Homogeneous	Weighted neural network	Financial Prediction	Sensitivity, specificity	D. West et al. [63]
5	Homogeneous	Neural network	Financial Prediction	Accuracy, type I error, type II error	Chih-Fong Tsai et al. [55]
6	Homogeneous	Neural network	Financial Prediction	Accuracy, type I error, type II error, AUC	[123]

Continued on next page

Table 2.7 – continued from previous page

S No.	Ensemble Technique	Base model	Domain	Evaluation Parameters	Ref.
7	Homogeneous	Support vector machine	Traffic incident detection.	Detection rate (DR), false alarm rate (FAR), mean time to detection (MTTD)	L. Nanni et al. [165]
8	Homogeneous	C4.5 decision tree algorithm	Gene expression prediction	Accuracy	C. Wang [277]
9	Heterogeneous	Decision Stump, Random Tree, Reduced Error Pruning Tree	Banking	Coefficient of determination (R2), RMSE and relative absolute error(RAE)	H. Erdal et al. [84]
10	Homogeneous	Neural network	Pattern classification	Accuracy	N. Liu et al. [141]
11	Heterogeneous	IBL, KStar and SMOReg models	Crude oil price forecasting	Root mean square, correlation	L. Gabralla et al. [117]

Continued on next page

Table 2.7 – continued from previous page

S No.	Ensemble Technique	Base model	Domain	Evaluation Parameters	Ref.
12	Heterogeneous	Neural fuzzy (NF), k-nearest neighbor (KNN), quadratic classifier (QC),	Cancer prediction	Accuracy	Sheau-Ling Hsieh et al. [181]
13	Heterogeneous	Neural Network, Hidden Markov Models HMM1 and HMM2	Bio-informatics	Accuracy	P. Martelli et al. [146]
14	Homogeneous	C4.5	Finance	Accuracy, type-1 error, type-2 error, F measure	You Zhu et al. [205]
15	Homogeneous	Random Forest	Food Security	Area under Curve, True skill statitics (TSS)	CR Mi et al. [56]
16	Homogeneous+ Heterogeneous	Decision tree, bagging, boosting, random sub-space, RS-boosting, multiboosting	Finance	Accuracy, Area under Curve,	You Zhu et al. [206]

Continued on next page

Table 2.7 – continued from previous page

S No.	Ensemble Technique	Base model	Domain	Evaluation Parameters	Ref.
17	Heterogeneous	XGBoost, decision tree	Medical	Area under Curve	S. AlAref et al. [163]
18	Heterogeneous	ELM, MARS, M5 Tree, SVR	Ground water study	Correlation, accuracy	R. Barzegar et al. [153]
19	Heterogeneous	SVM and Frequency Ratio	Flood Management	Accuracy	H. Mojaddadi et al. [86]
20	Heterogeneous	C4.5 tree, NBS, IB3, SVM	Precision Oncology	Accuracy	L. Mirsadeghi et al. [122]

2.3.1 Homogeneous Ensemble

This employs single learning method using distinct sub-samples of training dataset. Many variants of homogeneous ensemble are explained below:

i Bagging:

It stands for Bootstrap Aggregation. In this technique, ensemble model is constructed by employing random sub-samples of the training samples using the combination of multiple homogenous models. Bagging is successfully employed by practitioners for revamping the classification accuracy of single classifier [12, 67, 144, 166].

The success of technique lies in reducing the variance of classifier without affecting the bias [95]. The technique is also applicable for extracting elongated structures in getting more reliable clustering solutions[45].

ii Attribute Bagging:

The variant of bagging for feature subsets called attribute bagging is also proposed by researcher for improving the stability and accuracy of classification [154]. AB focuses on randomly selecting the features for training subsect to introduce diversity. AB is also used for outlier detection for modeling high dimensional finance data [36]. Bias error is tested to calculate the average error of the classifier on different training set. Additionally, variance error is also important. It is an additional error caused by assumption of model. Hence, variance error values are changed on changing the different classifier and bias error varies on changing the different training samples [69].

iii Boosting:

It builds ensemble using subsets of the training data and combining multiple homogenous models i.e. the models typically of the same type. But unlike bagging, the subsamples generated from the training data are not random [27]. The technique of creating new training samples is based on the utilizing the samples which were misclassified in the last iteration. The sub-samples of the training data are utilized to produce a series of averagely performing models and then their performance is boosted by combining them using a specific cost function like majority voting[27].

iv Boosting by weights:

Boosting is performed by sampling as well as weights[102]. Adaboost model implements boosting by assigning weights to the original training data and then rearranging and tuning these weights as each classification model is trained by the base classifier algorithm [102]. In boosting by sampling, subsamples are extracted with replacement from training data. A research work compares the performance of bagging, boosting and random sampling in building ensemble on 33 datasets [27]. For improving classification performance, bagging and boosting are performed on unstable decision tree algorithm.

Bagging method is beneficial if the base learning algorithm is not stable (like C4.5 algorithm) and with high sensitivity i.e. the minute changes to the training sample result in considerable changes in the learned machine learning classifier [27]. It is also applied to the real time problem like for improving the accuracy of churn prediction in wireless telecommunication company [37]. OCR performance is also improved by

using bagging and boosting using neural network as a base model [228]. Performance of genetic algorithm is improved by manipulating the training data using bagging and boosting [228]. The nearest mean machine learning classifier is improved using both bagging and boosting in order to prove that when diversity is maintained, an ensemble performance is much superior than the homogeneous classifiers [137].

v Wagging:

It is another variant of bagging. Here, the classifier is trained using different of training samples [67]. Unlike traditional bagging that utilizes bootstrap samples of training dataset, wagging is implemented by assigning random weights to the training samples. Some researchers use gaussian noise or poisson distribution to generate instance weights [67] . The training samples which are assigned with zero weights can be removed from the training data.

vi Multiboosting:

It is another innovative method that combines efficient adaboost model with wagging[278]. The technique is superior because it handles bias (adaboost efficiently work on bias) and reduces variance (wagging works on variance). Generally, C4.5 algorithm is used as a base algorithm while implementing multiboosting. Multiboost package implemented in C++ is also available to employ multiboosting [58]. It has been proved on many UCI datasets problems that multiboosting produces better accuracy that adaboost and wagging [278].

vii DECORATE:

It is Diverse Ensemble Creation by Oppositional Relabeling of Artificial Training Examples (DECORATE) produces ensemble using synthetic data for training process [147]. To maintain diversity, artificial subsets of data are generated randomly from training set and they are assigned with the class which disagree the current decision. The classifier which gives highest accuracy on training data is used to build ensemble. By generating the synthetic training samples, decorate acts as a strong performing diverse ensemble. It is specifically used to improve the accuracy of classifier when the training samples are not limited[147].

2.3.2 Heterogeneous Ensemble

This employs many learning methods to generate ensemble. Different variants of heterogeneous ensemble are explained below:

i Majority voting:

In this ensemble is built using the majority rule of decision making i.e. final decision making of the classifier upon the decision of learners which is voted by more than half of the classifiers [61]. It is the straightforward solution of incorporating the predictions from multiple machine learning models [74]. The elementary form of majority voting counts single vote for each classifier by giving equal priority to each of the classifier [119]. Weighted Voting a variant of majority voting where each classifier has been provided different degree of weight to carry weighted influence in the final prediction [140]. The weights are generally associated with each classifier considering its performance on the dataset.

ii Stacking:

The goal of stacking is to improve the accuracy along with working on the scalability by employing the output of the Level (n-1) classifiers as training data for another classifier of Level n to approximate the target function [109]. Unlike majority voting, stacking integrates the classifiers in non-linear fashion. For constructing heterogenous ensemble, researchers przoved the performance of stacking is comparatively better than other techniques [167]. Stacking has been proved to be the best when classifiers used for ensemble are linear regression and probability distributions based [167]. When the research is extended with the set of meta-level features and multi response model trees, it has been demonstrated that the performance is better for the latter extension [167].

iii Cascade Generalization:

It is another variant of stacking. It sequentially employs the set of machine learning classification models [143]. At each step of the sequence, new attributes are added to extend the original data. The probability distribution from the base classifier method is favoured by researchers to derive the new attributes. The method helps in reducing the bias. To implement cascade generalization, generic algorithm of decision trees with

maximum depth is demonstrated [87]. A cascade implementation of support vector machine is also produced to very large dataset to reach the global optimum [85].

2.4 Multi Criteria Decision Making

Multi-criteria decision making is a sub-area of an operational research. The rationale is the explicit evaluation of different criteria influencing the decision making process. The field can be further studid into sub-branches like multi-objective decisions and multi-attribute decisions [281].

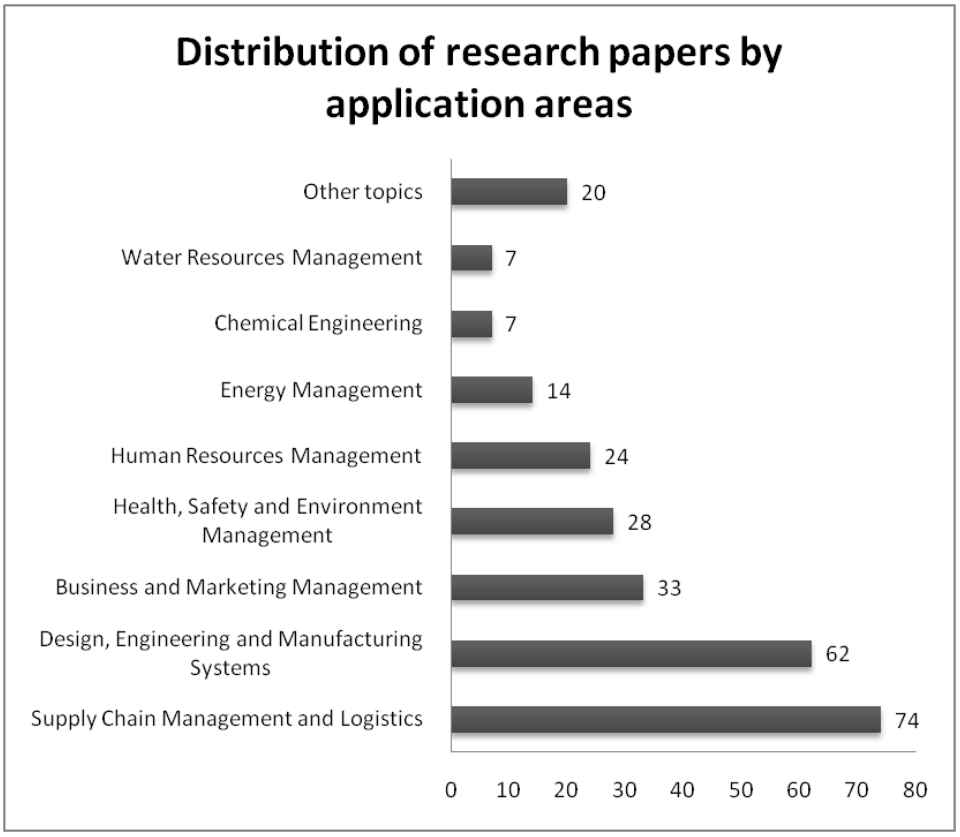

Figure 2.7: Research of TOPSIS in different application areas [130]

Although different types of methods are available for applying multi-criteria decision making, some common terminology is used for all of them as discussed below:

i Alternatives: It represents the multiple criteria or options, that can influence the process of decision making. The number of options/criteria should be finite in number.

ii Criteria: There are set of criteria that can influence the selection of alternatives.

iii Weight: It is important to explore the relative importance or ranking of these alternatives.

iv Decision Maker: These are the experts who are aware of the relative ranking of the criteria to be considered.

It is important to include all the criteria and alternatives. Redundant options and criteria can mislead the decision making. MCDM methods are implemented successfully in number of domains like engineering [227], supplier selection [189], integrated manufacturing [249], materia selection for construction industry [106], investment decisions [10], etc. Mark et al. reviewed and compared different multi-criteria decision making methods [271]. There are numerous MCDM methods available but only variants of VIKOR and TOPSIS methods work on the basis of an aggregation function [176]. Many researchers integrated TOPSIS technique in their case studies and working on proposing the variants of TOPSIS algorithm. There are extensive range of applications in diverse research fields where the TOPSIS multi-criteria methods is successfully applied by researchers. M. Behzadian et al. reviewed 266 research contributions of implementing TOPSIS method in nine different domains of research [130] as shown in the Figure 2.7.

The utmost usage of TOPSIS method for solving different research problems is found in supply chain management and engineering applications. 27.5 percent of the research is observed in solving supply chain and logistics problems whereas 23.04 percent of research contributions are observed in solving design, development, and engineering problems. Average number of research contributions are there in other fields like human resource, health care and energy management problems. Minimal research contributions are observed in the area of chemical, water resource management, and other topics of research.

Researchers integrated TOPSIS with fuzzy set methods to find the best suppliers for a buyers[70]. Wang et al. modified TOPSIS for risk assessment of bridge [204]. Osiro et al. compared AHP and TOPSIS methods when integrated with fuzzy theory for supplier selection problem [72]. Recently, Dikopoulou et al. (2017) also proposed a fuzzy integrated variant of TOPSIS algorithm in their research work [210]. Hence, instead of using TOPSIS method as a stand-alone approach for research, practitioners are integrating it with other

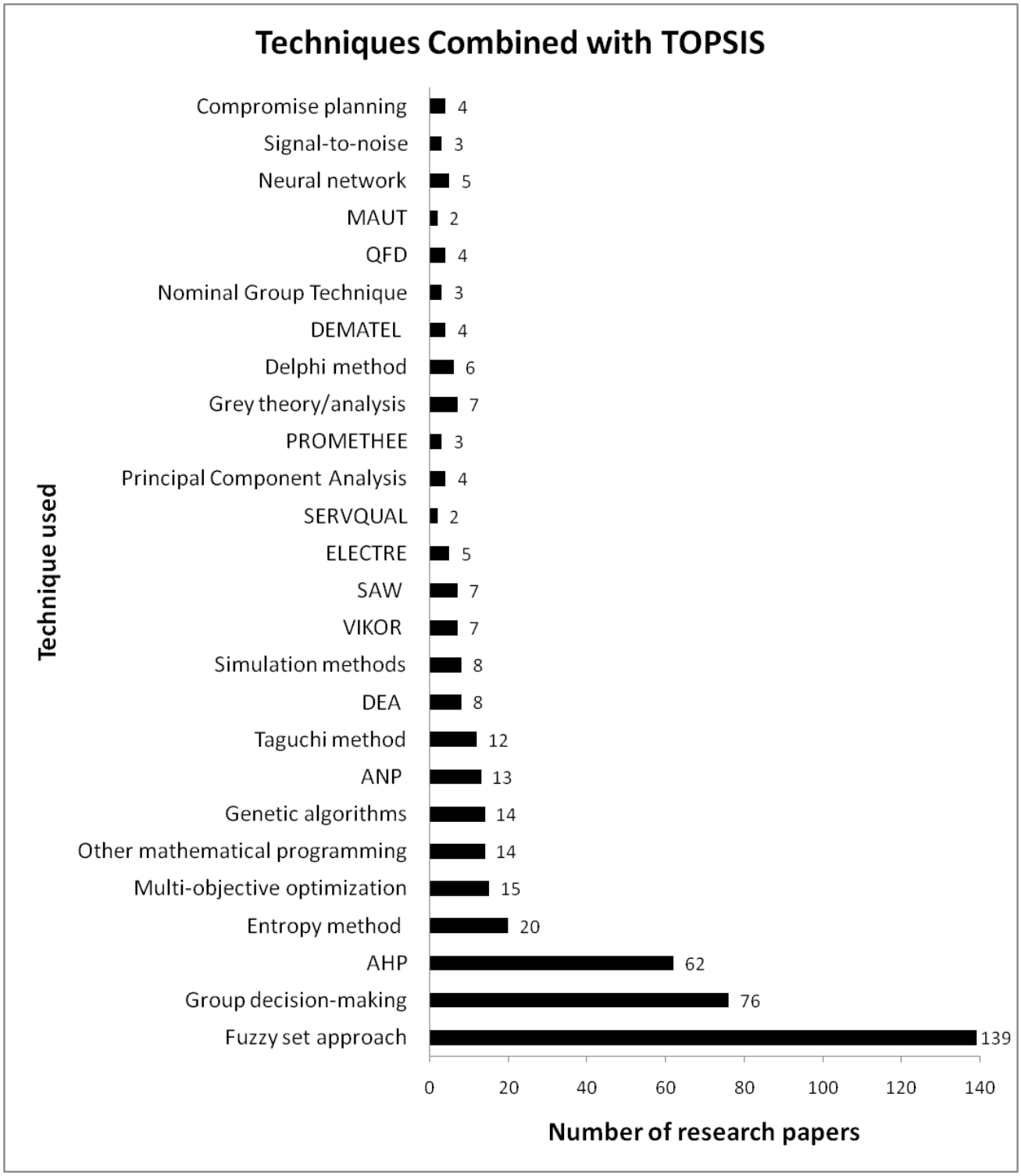

Figure 2.8: Combination of TOPSIS with different methods

methods as presented in the Figure 2.8. As we can observe in the graph, most of the researchers combined TOPSIS with fuzzy set approach to meet their research goals. Other than fuzzy set, techniques like AHP, genetic programming, multi-objective optimization are also integrated with TOPSIS in various case studies.

Chapter 3

Proposed Framework: Multi Criteria based TOPSIS Ensemble (MCTOPE)

This Chapter presents the architecture of the proposed Multi Criteria based TOPsis Ensemble (MCTOPE) framework where TOPSIS is Technique for Order of Preference by Similarity to Ideal Solution is a multi-criteria algorithm. The layered view, the abstract view, and the detailed view of the proposed framework are discussed in this Chapter.

3.1 The Architecture of MCTOPE Framework

In the research work carried out so far in the domain of machine learning and data analytics, ensemble machine learning approach is applied to revamp the accuracy of the prediction results. However, slightly different hybrid architecture integrating multi-criteria decision making for ensemble building is presented here. The classical methods test accuracy improvements in the prediction results after ensemble building, which is not always guaranteed. The proposed architecture is different because it considers six different evaluation metrics during ensemble building process using multi-criteria analysis algorithm and works on optimizing the results for more than thousand iterations to find the best candidate of the ensemble model.

3.1.1 Layered View

The architecture of MCTOPE framework can be divided into three layers:

Table 3.1: Layers of MCTOPE Framework

Layer	Modules	Sub-modules	Processes
Layer 1	Machine Learning Module (M1)	Data Preparator, Prediction Engine	Metrics Evaluation, Multi criteria TOPSIS Performance Analysis
Layer 2	Result Gathering Module (M2)	None	Prediction and Evaluation
Layer 3	Reporting Module (M3)	None	Reporting

i Layer 1: It consists of machine learning module (M1). The collected data is integrated, and is fed for preparation. Data preparator is responsible for pre-processing part like cleaning and refining of the data. After pre-processing, data is to used to train the prediction engine. The performance of the prediction is measured and optimization is performed iteratively.

ii Layer 2: It consists of result gathering module. The prediction results are gathered and analyzed at this stage. The performance of the prediction engine is optimized until the satisfactory results are achieved.

iii Layer 3: It consists of reporting module. This layer works on the analysis and visualization of results. The decision maker analyzes the results and works on decision making.

The detail of the modules and processes that are accumulated in each layer are summarized in the Table 3.1

3.1.2 Detailed Architecture

The novel architecture of Multi Criteria Technique of Order of Preference based Ensemble (MCTOPE) is presented in the Figure 3.1. To further elaborate, the architectures of the data preparator and prediction engine are also presented in the Figure 3.2 and Figure 3.3 respectively.

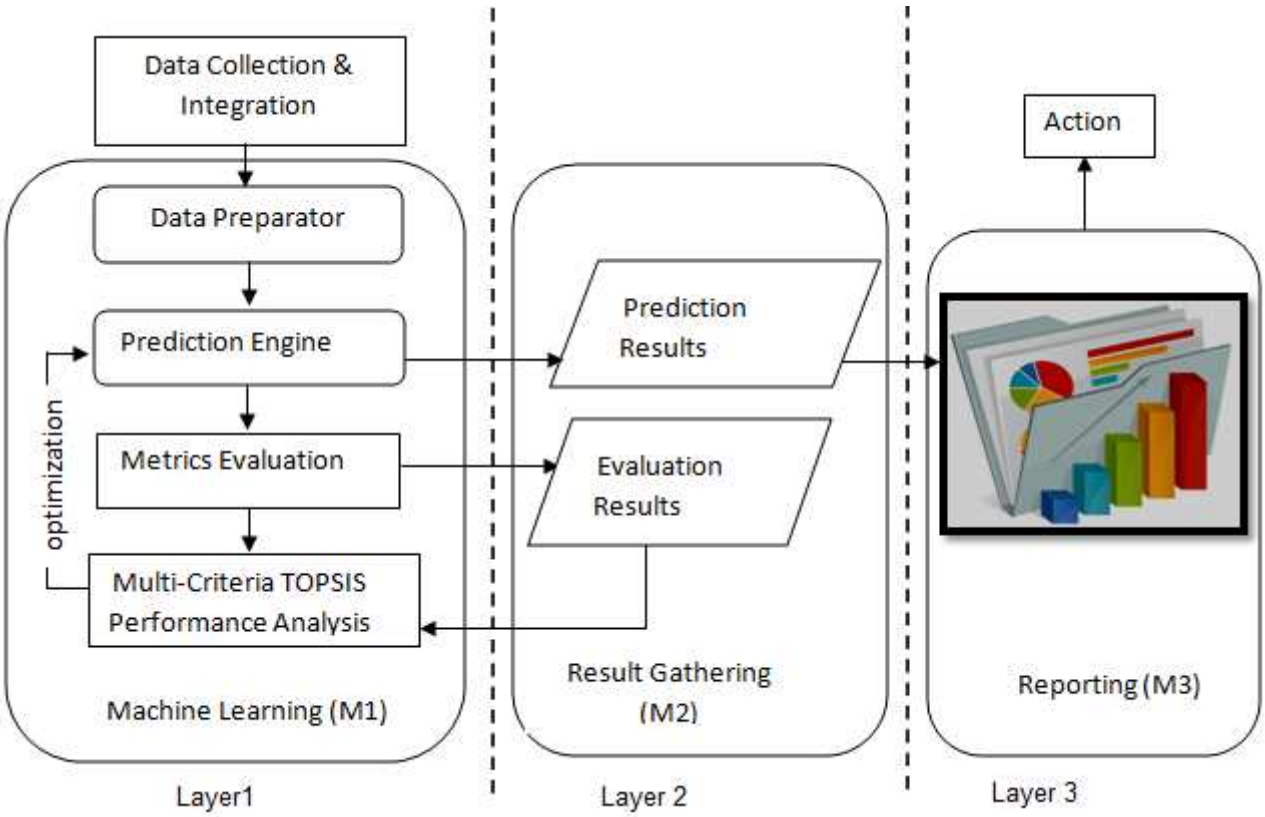

Figure 3.1: MCTOPE Architecture

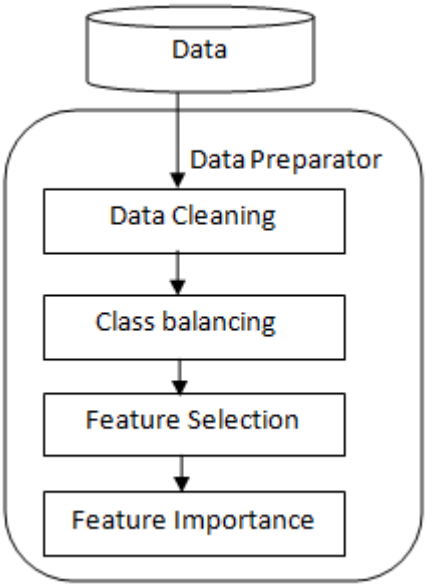

Figure 3.2: Architecture of Data Preparator in MCTOPE Architecture

3.1.2.1 Data Preparator

The collected data is generally messy and unstandardized. Sometimes unstructured data is collected from different sources. Mostly practitioners spend more than half of the time in cleaning and manipulating data before actually starting the data analysis. Various phases of data-preparator are discussed here.

i Data Cleaning:

Once the data is collected, it is need to be cleaned and transformed into useful and appropriate form by data preparator. Various steps are followed to clean the data:

(a) Firstly, the data is examined for errors and anamolies. Extreme values are carefully examined before going to the next step.

(b) Secondly, missing values need multiple imputations. It is important to mark missing values without losing the collected information. Instead of deleting the data points containing missing data, mean substitution imputation is used in the proposed method.

(c) Redundant information is removed by deleting duplicate values in data.

(d) Data consistency is verified by checking the range of feature values.

(e) The validity of the data is also tested.

ii Class Balancing: Sometimes the count of samples in one class is far lesser than the count of samples in the other class. This triggers biasing. The problem is severe when the proportion of minor class is less than 10 percent. Sampling technique is much favoured by the researchers [211].

(a) Under-sampling technique targets the majority class and discards some of its samples to balance the class-proportion of the samples. But, useful data depletion is the major downside of this method.

(b) Over-sampling technique targets the minority samples. The idea is to replicate the minority samples for increasing the proportion of minority class.

(c) SMOTE is Synthetic Minority Over sampling Technique that generates a synthetic sample *syn* produced by the fusion of two randomly fetched nearest neighbour samples s1 and s2 from the minority class samples as follows:

$$syn = s_1 + \mu * (s_2 - s1), 0 \leq \mu \leq 1 \tag{3.1}$$

Lately, many researchers have employed SMOTE for handling the class imbalance issue in the diabetic and lung cancer prediction [39, 115].

iii Feature Selection: The proposed framework has adopted an ensemble feature selection method to extract the best features of the dataset that are contributing to the classifier's performance. Each data sample having properties or features p_1,p_2...,p_n. The pointless information is need to be removed. For this, firstly the redundant features with the correlation coefficient value (r greater than 0.75) are eliminated. The filtered features are then ranked by Mean Decrease Gini metric value which is defined in the equation:

$$MDG = \sum_{i=1}^{c_n} pi(1 - p_i) \tag{3.2}$$

Where c_n is the number of classes in the target variable and pi is the ratio of this class. Mean Decrease Gini is a variable importance measure works on the Gini impurity index. The intent is to calculate the node-split during the training process.

The node split is made on the variable p (parent node), where the gini index value for the parent node is greater than the respective child node. Summing up the gini decreases for each individual feature over all trees in the forest gives a quick feature importance that is often very consistent with the permutation importance measure as used by researchers latterly for classification. Let FF is the further filtered set obtained by ordering the features in the descending order of Gini importance. If G is the Ginni importance for F_i feature then, a binary tree is produced with nodes represented with descending order of their Ginni importance. The filter works on searching the optimal feature set by testing the performance of classifier.

3.1.2.2 Prediction Engine

It is not idiosyncratic to claim that the intent of building an ensemble is to revamp the classifier's performance to make it indistinguishable from the single classification system. The proposed framework automates the process the ensemble building. The models that are se-

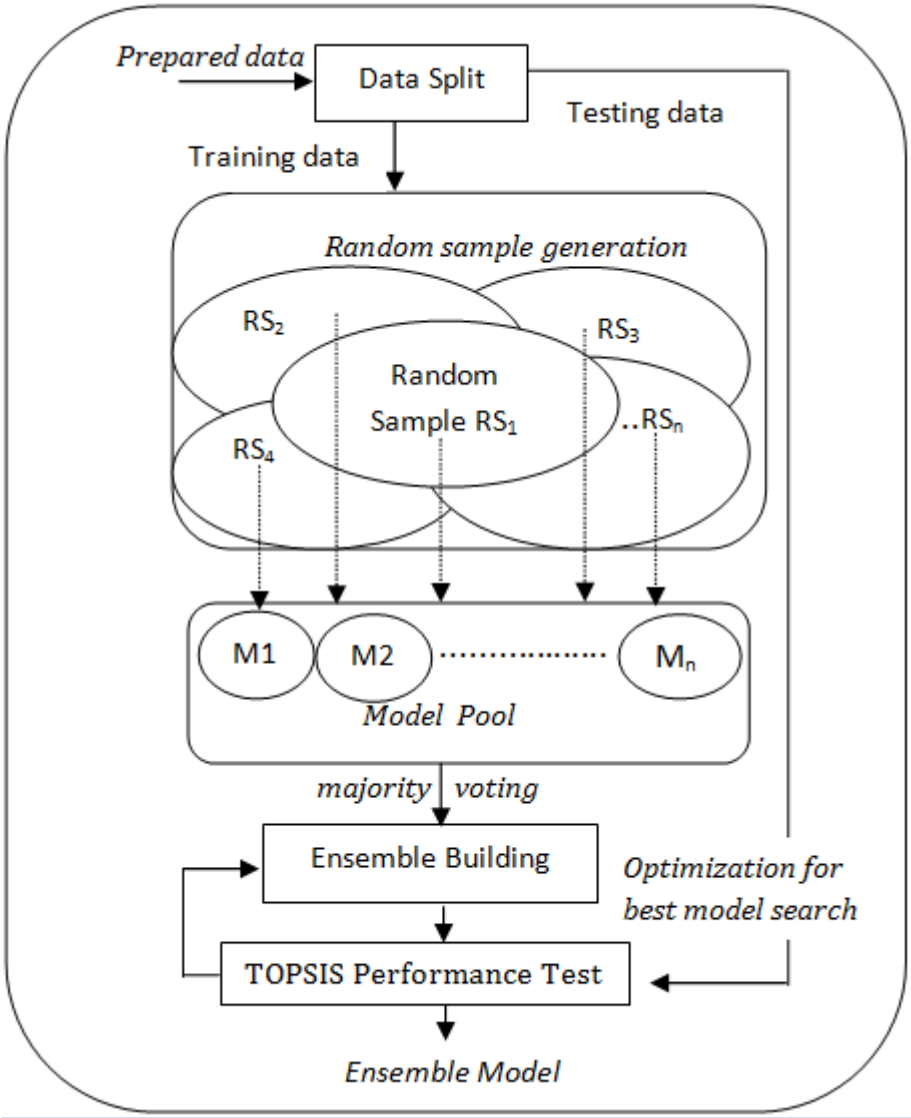

Figure 3.3: Architecture of Prediction Engine in MCTOPE Architecture; M: Machine learning model; RS: Random sample of training dataset

lected as the candidate of the model-pool are discussed here. In the traditional ensemble building framework, base classifiers are ranked according to their accuracy in predicting the target class. Some of the researchers combine the most diverse classifiers to improve the accuracy of the system [49, 127].

Mostly, researchers consider accuracy of an ensemble on the unseen dataset as the only performance-evaluation criteria. Accuracy calculates the number of correct predictions from all predictions. Accuracy is sometimes misleading and AUC (Area under the curve) is more preferred approach as compared to accuracy [11]. But, AUC alone is typically not enough to make the performance ranking of base-classifiers. The results of confusion matrix helps in summarizing the breakdown of errors, completeness, and exactness of a classifier during prediction [11, 268]. Based on these motivations in mind, we propose a novel ensemble builder based on multiple criteria analysis. Different from the traditional method, MCTOPE works on optimizing the overall performance of the system using TOPSIS algorithm, a multi-criteria decision making technique. It is presented in the Figure 3.3.

In the first step, the data is divided into two independent parts. One part is called training samples and other portion is called testing dataset. The rationale is to measure the performance analysis on entirely independent dataset called testing-dataset to avoid over-fitting. Training data T=$\{x_1, y_1 \ldots x_n, y_n\}$ which consists of n different training examples is constructed from the product of I (input space) $\times$ O (output space). Each training example x_i, y_i is represented by a set of condition attribute vector C (c_1, c_2, $\ldots$ c_k) and a target class d_i, which may belong to a set of classes D=$\{d_1, d_2, \ldots d_n\}$ as represented in the Table 3.2. The objective is to approximate an unknown function $f: I \rightarrow$ O by generating a function known as hypothesis. The hypothesis function t defined into some hypothesis space H. To meet this objective, machine learning algorithms utilize the training data to search for the best hypothesis function h. Intuitively, ensemble classifier generates the hypothesis function t from the combination of individual hypothesis $\{t_1, t_2, t_3, \ldots t_e\}$ of e different classifiers using a specific aggregation method. In the second step, random samples are generated to train the classifiers in the model-pool.

In the third step, initially, a preliminary-ensemble is generated by combining m ($1 \leq$ m $\leq$ 10) different models by random using majority voting aggregation technique. TOPSIS multi-

Table 3.2: A classification training data set with n=3 objects, k=5 conditional attributes and i=2 classes.

U			**C**			**D**
	c_1	c_2	c_3	c_4	c_5	d
x_1	xp	by	cu	di	ek	d_1
x_2	fc	gt	h	ih	jj	d_2
x_3	ke	lf	m	nm	ob	d_1

critera decision score of preliminary-ensemble is evaluated using six different performance measures. In the next iteration, a new ensemble is produced using different training samples, and with a new set of models in model-pool. The performance of this ensemble is compared with the preliminary ensemble, and the model with the higher TOPSIS score is saved. This phase is called optimization for the best model search. At the end n iterations (say n= 5000 for small dataset), a final ensemble with the highest TOPSIS score is declared as the winning-ensemble.

3.1.2.3 Model Pool

Ten state-of-the art machine learning classifiers are employed to make the pool of classifiers called model-pool.

i Decision Trees (DT): The model is an extended variant of C4.5 classifier model. It works by classifying samples by sorting them down the tree [250].

ii AdaBoost (AB): It is a successful ensemble classifier by Schapire and Freund. It employs multiple learners to finally make a more powerful learning algorithm [262].

iii Random Forest (RF): It is an ensemble based learning classifier. It generates a forest of decision trees by employing random inputs with goal of improving the classification accuracy rate [128].

iv Support Vector Machine (SVM): SVMs searches for data points that are present at the boundary between two classes and refer them as support vectors. It is a preferred technique for classification [172].

v Probit Linear Models (PLM): Linear Model is a traditional regression method for fitting the data. For binary classification, it is transformed using a logistic or probit function and offers similar results to the logistic regression [131, 215].

vi Neural Network (NN): It is inspired from biological neural networks and used to model complex relationships, and useful patterns in statistical data [169].

vii Decision Stump Model (DSM): It is a one-level decision tree. It is also used as base learners in ensemble models [190].

viii J48: It builds decision tree based on the theory of information entropy. J48 is a open source java implementation of C4.5 algorithm [251].

ix Naive Bayesian (NB): It computes the conditional a-posterior probabilities of a categorical class variable of a given independent predictor variables using the Bayes Rule [254].

x Bayesian Network (BN) This model is based on probabilistic and directed acyclic graph theory. It works by constructing a acyclic graphical model to represent the features and their conditional dependencies [13, 160].

3.1.2.4 Performance Evaluation

For comprehensive evaluation of ensemble model performance, multiple performance metrics are considered. Among the several solutions available, Multi-Criteria Decision making is the most prevalent approach. The complete work-flow of the multi-criteria decision analysis is shown in the Figure 4.9.

TOPSIS is a popular multi-criteria analysis technique. It works on the principle of finding a solution which should be closest to the defined ideal solution. Also, this solution should be farthest from the defined negative-ideal solution. Ideal solution i is the set of evaluation-criteria solution with maximum benefit and can be described as

$$i = \{min(errorrate), max(sensitivity), max(specificity), max(MCC), \\ max(Fscore), max(AUC)\} \tag{3.3}$$

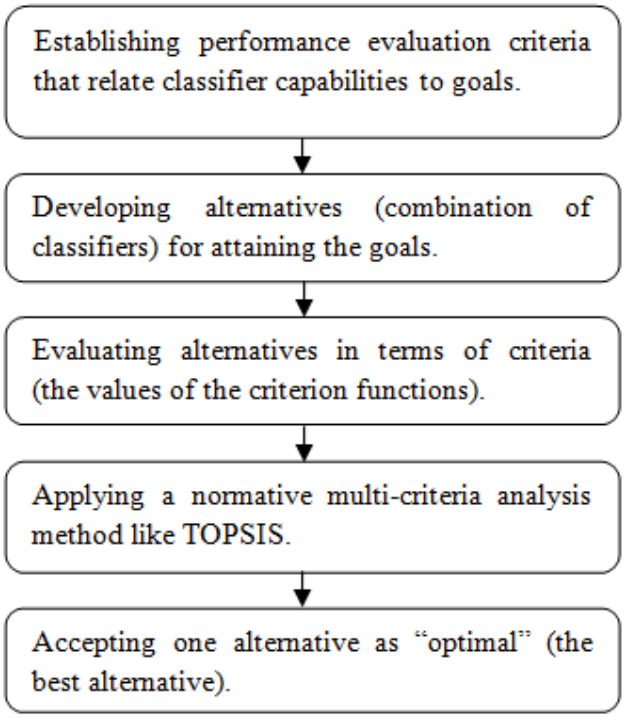

Figure 3.4: Multi-Criteria Analysis Flow

Table 3.3: Confusion matrix

	True Reference	
Predicted Condition	Positive Class	Negative Class
Positive Class	True positive X	False Negative Z
Negative Class	False Positive Q	True Negative Y

Negative ideal solution j is the solution with the maximum loss and can be described as

$$j = \{max(errorrate), min(sensitivity), min(specificity), min(MCC), min(Fscore), min(AUC)\} \tag{3.4}$$

Choosing the evaluation-criteria that suits the goal of improving the classification performance is an important step of this process. To check the performance of the classifiers, K fold (K=10) validation is implemented and six performance metrics namely error rate, sensitivity, specificity, F score, MCC, and AUC as described in the Table 3.4 are calculated using the results of Confusion Matrix presented in the Table 3.3.

Sensitivity is the true positive rate (TPR), measures the hit rate of a classifier in prediction. Specificity measures the true negative rate (TNR) of a model. Area under the curve

Table 3.4: Performance evaluation metrics

Performance Metric	**Formula**
Type-I error	Q
Type-II error	Z
Sensitivity	$X/(X+Z)$
Specificity	$Y/(Q+Y)$
Accuracy	$(X+Y)/(X+Z+Q+Y)$
F Score	$(2*X)/((2*X)+(Q+z))$
MCC	$(X*Y)-(Q*Z)/\sqrt{2*(X+Y+Z+Q)}$

(AUC) is equal to the probability that a classification model will rank a randomly chosen positive samples higher than a randomly chosen negative samples. A graph is generated by plotting true positive rate (TPR) against the false positive rate (FPR), depicting the relative trade-offs between true positive and false positives [214]. F measure (F1 score) is a balanced score of sensitivity and specificity [248]. F2 score weights sensitivity value higher than specificity. Matthew's correlation coefficient (MCC) is another balanced measure that focuses on true and false positives, and negatives [248].

Chapter 4

Design and Implementation of MCTOPE

This Chapter reviews the detail and implementation of MCTOPE framework. The detail of designing UML digrams, experimental setup, and implemented case-studies are also presented well in detail.

4.1 Design

Unified Modeling Language (UML) is a general purpose standard modeling language to visualize the proposed framework.Generally, we classify UML modeling into structural and behavioural modeling to present both the static and behavioural views of the proposed framework, as presented and discussed in detail in this Section.

4.1.1 Structural Modeling

Structural models like class diagram is presented in the Figure 4.1 and the component diagram is presented in Figure 4.2 capture the static features of the system. Both the class diagram and the component diagram give an idea of the different elements of proposed system and the mechanism to assemble them.

4.1.1.1 Class Diagram

Figure 4.1 shows the class diagram. It depicts the static view of the proposed framework. It describes the various attributes. As presented in the Figure, any type of dataset like spread-

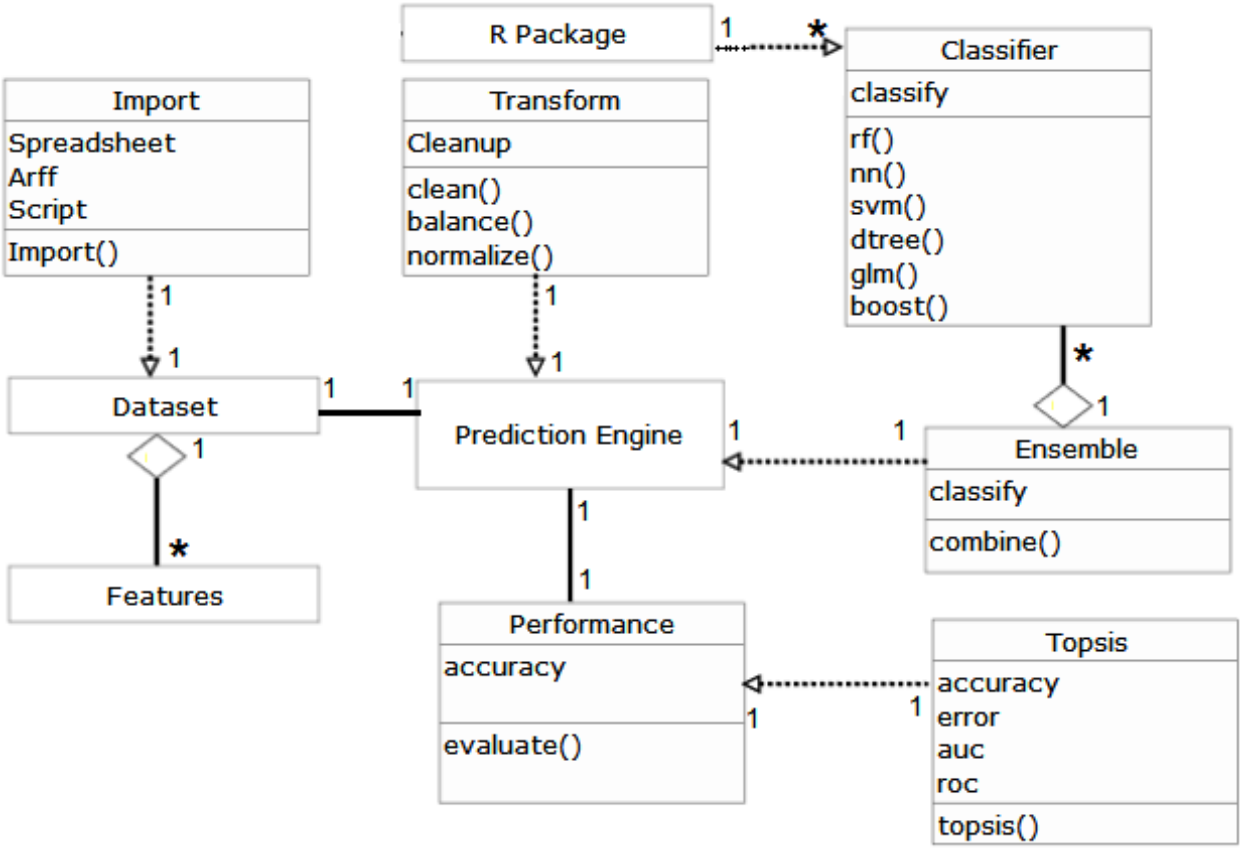

Figure 4.1: Class Diagram

sheet, arff file, script, etc. can be imported. Only one dataset of any type can be imported at one time. Each dataset can have many features associated with it. The dataset is fed to the prediction engine where different operations are performed like cleanup, balancing, normalization, etc. For modeling the data, the prediction engine calls different machine learning models like random-forest, SVM, etc. At prediction engine, ensemble function calls the combination of diverse classifiers. Later on, the performance of the built ensemble are checked using topsis performance calculator by utilizing different metrics like accuracy, auc, error rate, etc. Each ensemble is associated with number of classifiers. To implement the classifiers, R package is implemented.

4.1.1.2 Component Diagram

Component diagram also focuses on the physical aspects of the proposed framework. These physical aspects are different from the class diagram. Here, the major goal is to depict the

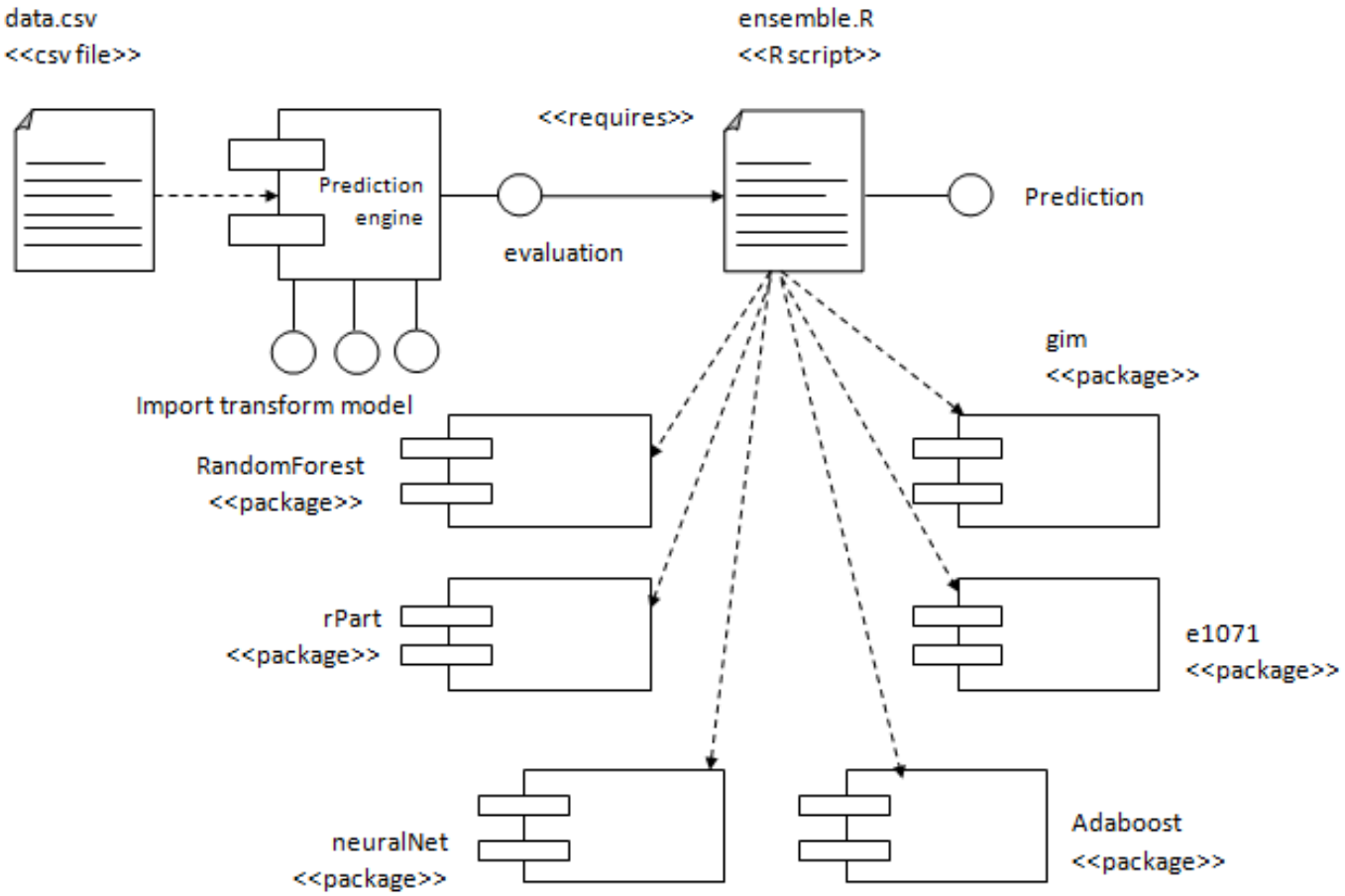

Figure 4.2: Component Diagram

executables, libraries, packages, etc. Figure 4.2 shows the component diagram of the proposed framework. Various packages required to implement the base classifiers for building an ensemble are presented in the diagram. For instance, random forest classifier call 'randomForest' package, neural network needs 'neuralNet' package. Similarly, e1071, rpart, etc. packages are required to implement the different classifiers needed to build an ensemble. R script is implemented to build an ensemble from the base classifiers.

4.1.2 Behavioural Modeling

Behavioral modeling include UML diagrams like activity diagram, usecase diagram, sequence diagram etc. and they depict the dynamic behaviour of the system. Figure 4.3 presents the usecase diagram, Figure 4.4 represents the Sequence Diagram, Figure 4.5 represents the state chart diagram and Figure 4.6 represents the activity diagram.

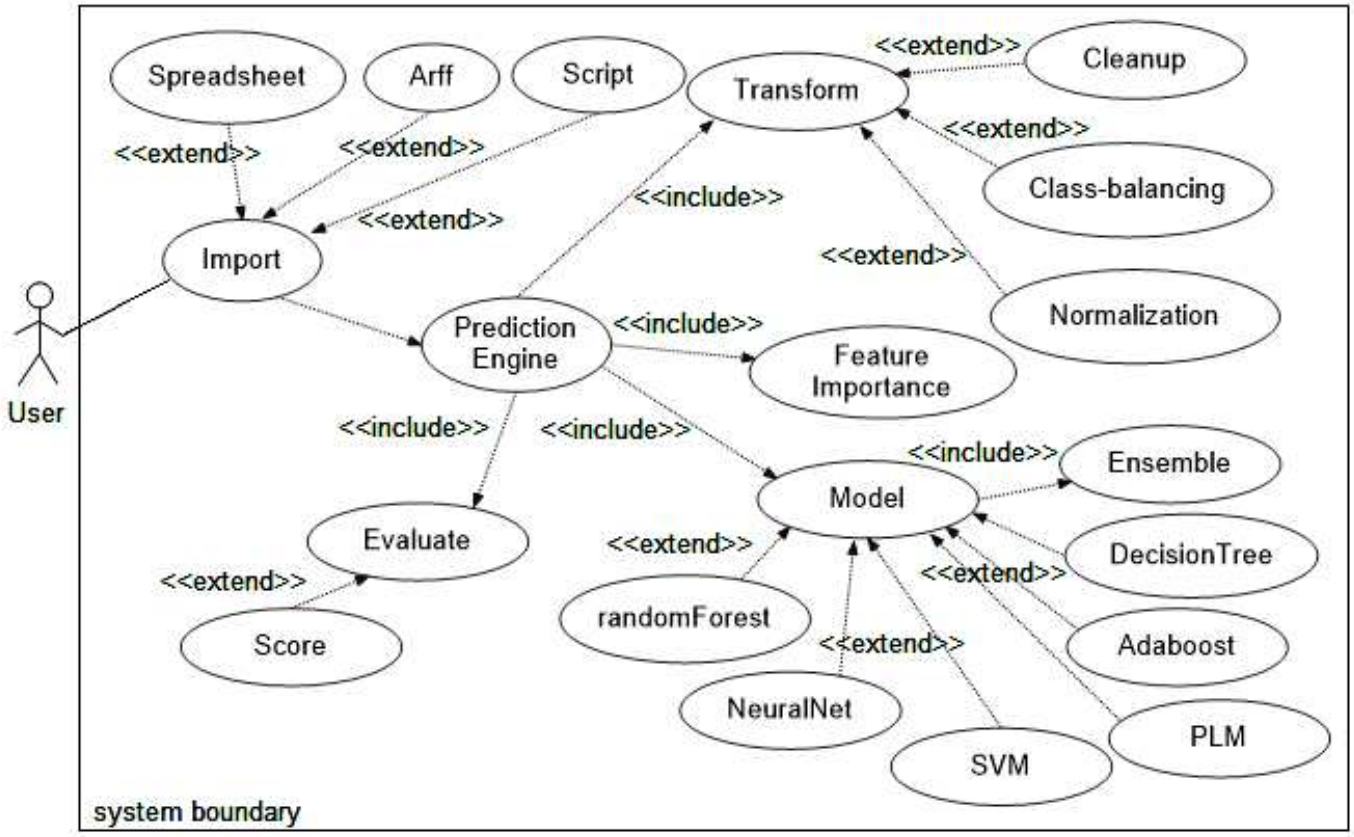

Figure 4.3: Usecase Diagram

4.1.2.1 Usecase Diagram

Use case diagram depicts the dynamic nature of the proposed framework and focuses on the internal and external factors present in the proposed framework. These internal and external agents are known as actors. In the Figure 4.3, system boundary represents the boundary of the proposed framework. Any agent outside it is called an actor. For instance, the user using the predictor is an actor here. User as an actor, submit the input, request the predictor for predicting results. Use case diagram helps in observing the external and external influence on the system. It is also useful in collecting the system requirements and finding the relationship between internal and external agents.

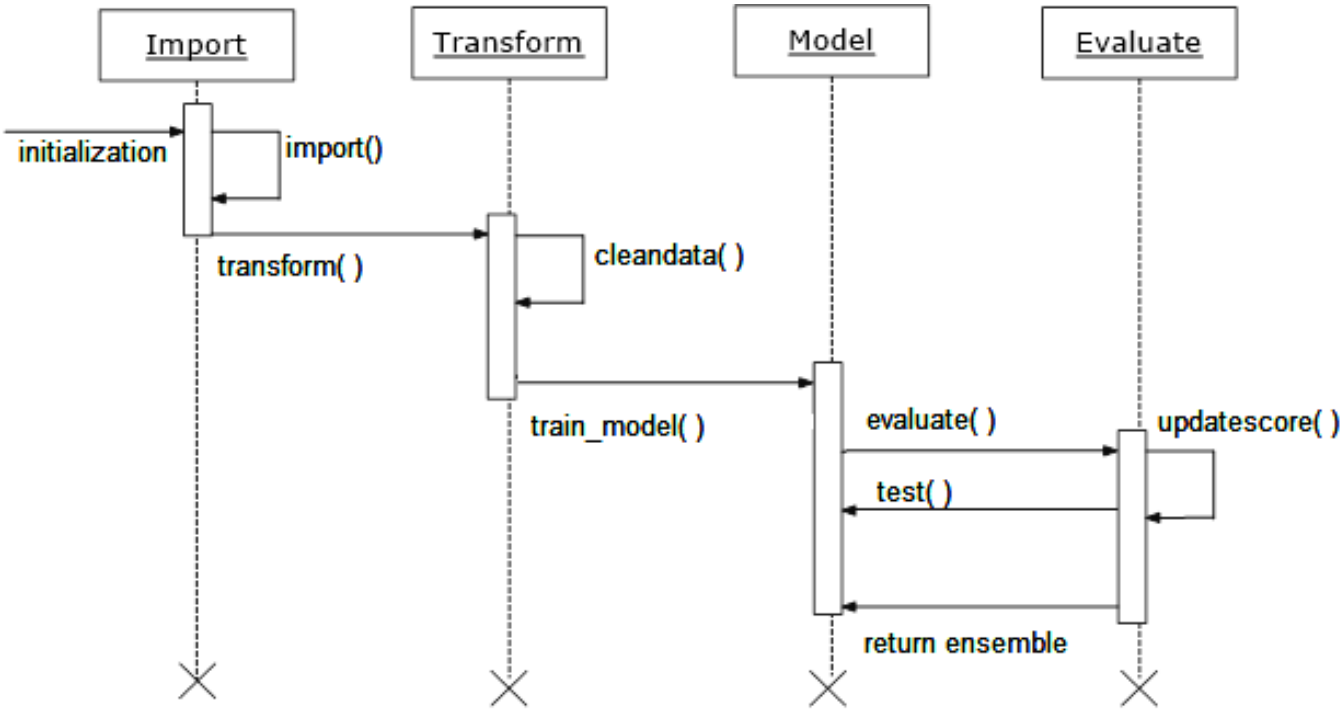

Figure 4.4: Sequence Diagram

4.1.2.2 Sequence Diagram

Sequence diagram depicts the dynamic nature of the proposed framework and focuses on the interactions occurring inside the proposed system. As shown in the Figure 4.4, lifelines of various objects and how these objects are communicating with each other is presented. As shown in the diagram, firstly 'import' object works and initializes the process. After importing the dataset, it is transformed and cleaned. After transformation, model training gets active. The performance of the model is evaluated and a new ensemble models are trained until the best performance score is updated.

4.1.2.3 State-chart Diagram

The Figure 4.5 represents the state chart diagram. It is used to depict the various states of the components or objects of the proposed framework. We can easily observe the flow of the control from one state to another. Firstly, the data is imported and processed. If the dataset is not valid, it can call the 'invalid' state, depicting the error event. If the dataset is valid, it is loaded and processed for missing value check. If missing values are present, cleanup

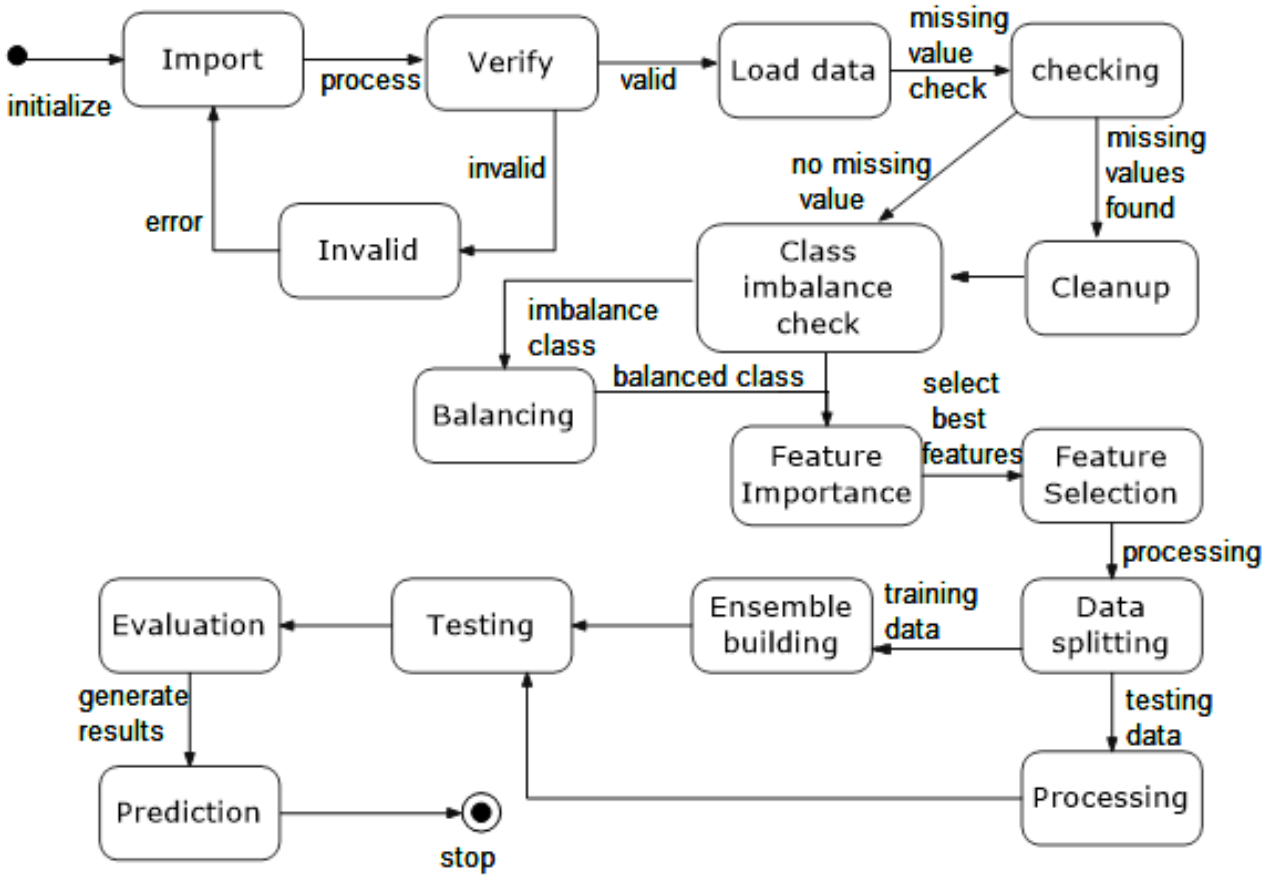

Figure 4.5: State chart Diagram

state is active. Otherwise, class-balancer checkup is active. If the classes of the data are imbalanced, balancing state is active. The balanced dataset is fed for feature selection. For this, importance of the features is calculated and irrelevant features are excluded by feature selector. Now, the remaining is split into two parts. The first part of training samples are used for ensemble building. In order to avoid biasing, it's performance is tested using testing datsets. The prediction results are gathered at the end.

4.1.2.4 Activity Diagram

The Figure 4.6 represent the activity diagram. It depicts the dynamic nature of the proposed framework and focuses on the flow of activities occurring in the proposed framework. Unlike flowchart, they have additional capabilities like parallel processing, branching, etc. User imports data to the system. System checks the validity of data and accepts the data. It tests

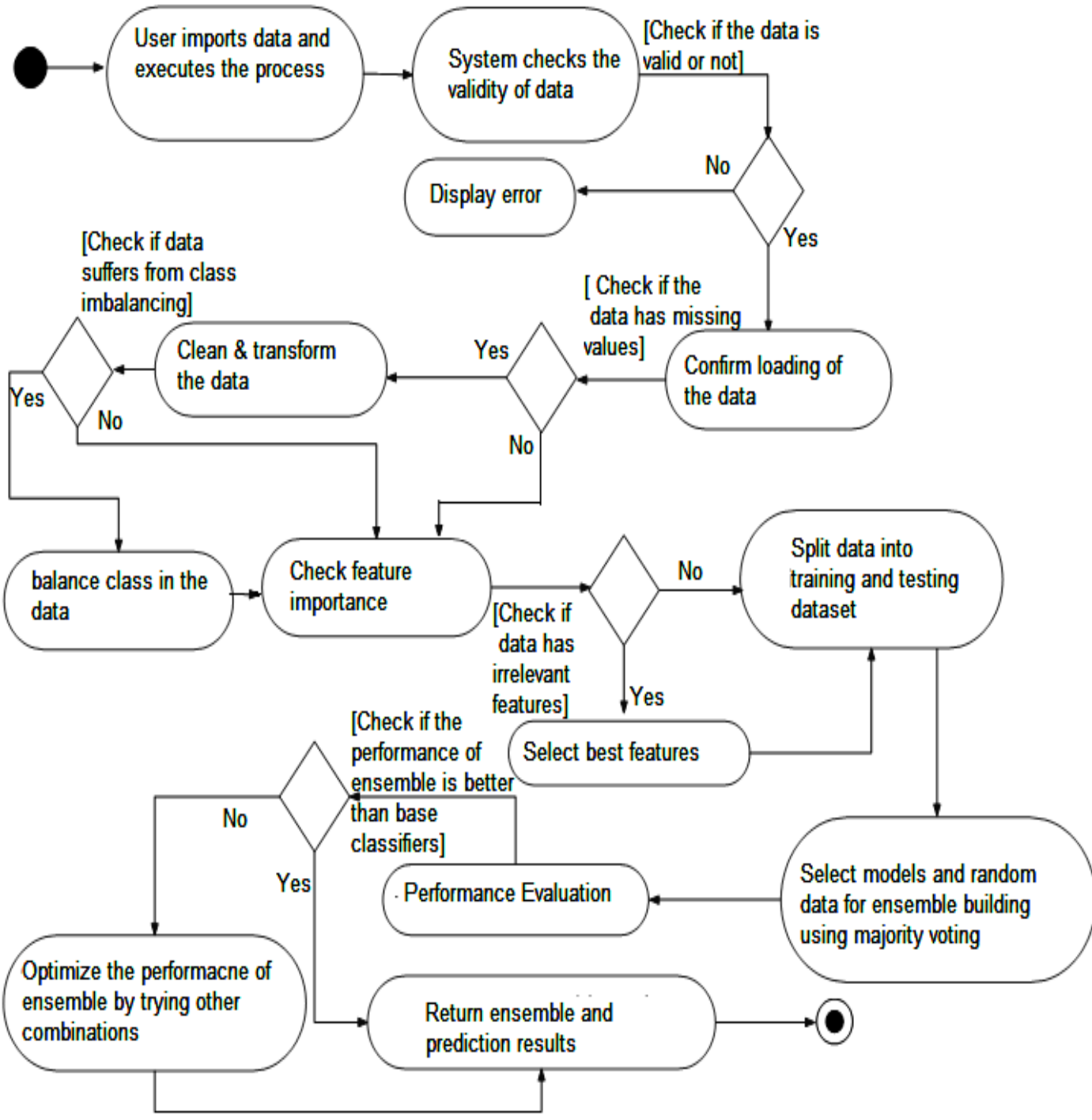

Figure 4.6: Activity Diagram

whether the data is suffering from imbalance or not. The system clean and transform the data and tests the missing values in the data. After balancing, and missing values imputation, it checks the feature importance. The relevant features are used for classification and irrelevant features can be rejected. After complete preparation of data, training samples are utilized for training the ensemble classifier. Performance of classification is optimized by trying different combinations of classifiers. The best ensemble classifier is finally employed for prediction.

4.2 Implementation

This Section discusses the experimental setup and implementation details of various modules and implemented machine learning models.

4.2.1 Experimental Setup

The R packages are used to implement the pre-processing and model building techniques. One shot training and testing technique is adopted here.

To test the robustness of designed framework, the process is iterated. To evaluate the performance of the proposed framework, different parameters namely accuracy, sensitivity, specificity, F-score, MCC, and area under curve (AUC) are used.

The models are available in R open source software. R is licensed under GNU GPL. The brief implementation detail of the models is summarized in the Table 4.1.

4.2.2 Class Balancing

It is implemented in R using SMOTE function and DMwR (Data Mining with R) package (version 0.4.1) of CRAN repository. SMOTE function is used to handle the classification problem of unbalanced dataset. It helps in generating a new SMOTEd dataset which has balanced class samples. We can also use it to directly execute a classifier on the SMOTEd dataset generated by this function.

Table 4.1: Machine learning classification methods

Model	Method	Package	Tuning Parameter	Ref.
DT	rpart	rpart	MinSplit =20, MaxDepth =30, MinBucket =7	[250]
AB	adaboost	fastAdaboost	Default	[262]
RF	rf	randomForest	mtry=500, sampling=bagging	[128]
SVM	ksvm	e1071	nu=10, epsilon=0.5	[172]
PLM	bayesglm	arm	Default	[131, 215]
NN	neuralnet	nnet	size=10, linout=TRUE, skip=TRUE, MaxNWts=10000,trace=FALSE, maxit=100	[169]
DSM	decision stump	RWeka	rules=6,pruned=25,smoothed=0.9	[190]
J48	J48	RWeka	Default	[251]
NB	NaiveBayes	RWeka	Default	[254]
BN	BayesNet	RWeka	Default	[160, 246]

> install.packages("DMwR")

> library(DMwR)

>SMOTE(formula, data, over = 200, K = 5, under = 200, learner = NULL)

-*formula*: It describes the prediction problem.

-*data:* A data frame containing the unbalanced class samples.

-*over:* It describes the percentage amount of over-sampling. It is number that represents the decision of how many extra sample cases are generated from the minority class.

-*K:* It indicates the number of nearest neighbors that are utilized to generated the new samples from the minority class.

-*under:* It describes percentage amount of under-sampling. It is number that represents the decision of how many extra sample cases are selected from the majority class for generating each case from minority class.

-*learner:* It is optional to specify any classifier that will be implemented on SMOTEd dataset.

4.2.3 Feature Importance and Feature Selection

The framework has adopted an ensemble feature selection method to extract the best features which are contributing to the classifier's performance. The steps followed to implement the ensemble technique are:

Step i: Generally, data set contain features which are highly redundant with each other i.e. contains similar information. R CARET package is used to generate a correlation matrix of features. mlbench package is a dependent package.

> install.packages("caret")

> install.packages("mlbench")

The dataset *"data"* is provided.

> load (data)

correlationMatrix function is used to generate the correlation matrix with feature 1 to n of dataset.

> correlationMatrix <- cor(dataset[,1:n])

> print(correlationMatrix)

In order to remove the pointless information, redundant features with the correlation coefficient value (r greater than 0.75) are eliminated.

> highlyCorrelated <- findCorrelation(correlationMatrix, cutoff=0.75)

> print(highlyCorrelated)

The filtered features are then ranked by Mean Decrease Gini metric value which is defined in the Equation 4.1:

$$MDG = \sum_{i=1}^{c_n} pi(1 - pi) \tag{4.1}$$

Where c_n is the number of classes in the target variable and pi is the ratio of this class. Mean Decrease Gini is a variable importance measure works on the Gini impurity index. The intent is to calculate the node-split during the training process. The node split is made on the variable p (parent node), where the gini index value for the parent node is greater than the respective child node. Let FF is the further filtered set obtained by ordering the features in the descending order of Gini importance. If G is the Ginni importance for F_i feature then, a binary tree is produced with nodes represented with descending order of their

Ginni importance.

The randomForest package provides the 'randomForest' function.

> install.packages("randomForest")

> library(randomForest)

Now, train the random forest classifier.

> rf<- build randomForest model

The importance of the features is listed as:

> rn <- round(randomForest::importance(rf), 2)

Now, order the features in decreasing level of importance.

> rn[order(decreasing=TRUE),]

4.2.4 Prediction Engine

This section presents the implementation detail of various classifiers.

4.2.4.1 Classification Models

Implementation of state-of-the art classifiers which are employed in the research work are discussed here.

i. Decision Trees (DT): The classifier is an extended variant of C4.5 classification algorithm.

Install package rpart for decision tree.

>install.packages("rpart")

Pseudocode for training decision tree:

> Include library rpart.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target," ",paste(c(selectedInputs),collapse = "+")))

Train decision tree.

>

model <- rpart(formula, trainDataset, method="class",parms=list(split="information"), control=rpart.control(usesurrogate=0, maxsurrogate=0)) Extract predicted values.

>Predicted <- predict(model, testDataset, type="class")

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

AdaBoost (AB): It is a successful ensemble classifier employing multiple learners to finally make a more powerful learning algorithm [262].

Install package ada for building adaboost model.

>install.packages("ada")

Pseudocode for training adaboost:

> Include library ada.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target," ",paste(c(selectedInputs),collapse = "+")))

Train adaboost model.

> model <- ada(formula, trainDataset, control = rpart::rpart.control(maxdepth = 30, cp = 0.01, minsplit = 20, xval = 10), iter = 50)

Extract predicted values.

>Predicted <- predict(model, testDataset, type="class")

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

Random Forest (RF): It is a popular ensemble based learning algorithm [128].

Install package randomForest for building random forest model.

>install.packages("randomForest")

Pseudocode for training decision tree:

> Include library randomForest.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target," ",paste(c(selectedInputs),collapse ="+")))

Train randomforest.

> model <- rf(formula, trainDataset, ntree=500, mtry=4, importance=TRUE, na.action=randomForest::na.roughfix, replace=FALSE)

Extract predicted values.

>Predicted <- predict(model, testDataset, type="class")

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

Support Vector Machine (SVM): It is a preferred technique for classification [172].

Install package kernlab for support vector machine.

>install.packages("kernlab")

Pseudocode for training SVM:

> Include library kernlab.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target," ",paste(c(selectedInputs),collapse = "+")))

Train svm using radial basis kernel rbfdot.

> model <- ksvm(formula, trainDataset, kernel="rbfdot", prob.model=TRUE)

Extract predicted values.

>Predicted <- predict(model, testDataset, type="class")

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

Probit Linear Models (PLM): Linear Model is a traditional regression method for fitting the data. For binary classification, it is transformed using a logistic or probit function and offers similar results to the logistic regression [131, 215].

Install package MASS for probit liner model.

>install.packages("MASS")

Pseudocode for training neural network:

> Include library MASS.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target,“ ”,paste(c(selectedInputs),collapse = “+”)))

Train linear model.

> model <- rpart(formula, trainDataset, family=binomial(link=“probit”))

Extract predicted values.

>Predicted <- predict(model, testDataset, type=“class”)

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

Neural Network (NN): It is inspired from biological neural networks [169].

Install package nnet for neural network.

>install.packages(“nnet”)

Pseudocode for training neural network:

> Include library nnet.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target," ",paste(c(selectedInputs),collapse = "+")))

Train neural network.

> model <- nnet(formula, trainDataset, size=10, linout=TRUE, skip=TRUE, MaxNWts=10000, trace=FALSE, maxit=100) Extract predicted values.

>Predicted <- predict(model, testDataset, type="class")

Extract actual values. >Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

Decision Stump Model (DSM): It is a one-level decision tree. It is also used as base learners in ensemble models ([190]).

Install package RWeka for decision stump.

>install.packages("RWeka")

Pseudocode for training decision tree:

> Include library RWeka.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target," ",paste(c(selectedInputs),collapse = "+")))

Train decision stump.

> model <- DecisionStump(formula, data, subset, na.action, control = Wekacontrol(), options = NULL) Extract predicted values.

>Predicted <- predict(model, testDataset, type="class")

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

J48: It builds decision tree based on the theory of information entropy [251].

Install package RWeka for J48.

>install.packages("RWeka")

Pseudocode for training J48:

> Include library RWeka.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

> trainSample <- sample(totalDataset, totalDataset * training/100)

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target,“ ”,paste(c(selectedInputs),collapse = “+”)))

Train decision tree.

> model <- J48(formula, data, control = Wekacontrol(R = TRUE)) Extract predicted values.

>Predicted <- predict(model, testDataset, type=“class”)

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

>accuracy <- round(mean(Actual==Predicted) *100,2)

Naive Bayesian (NB): It does classification by employing the Bayes Rule [254].

Install package e1071 for naive bayes classifier.

>install.packages(“e1071”)

Pseudocode for training naive bayesian classifier:

> *Include library e1071.*

> *Load dataset.*

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target," ",paste(c(selectedInputs),collapse = "+")))

Train naive bayes.

> model <- naiveBayes(formula, data, subset, na.action = na.pass) Extract predicted values.

>Predicted <- predict(model, testDataset, type="class")

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

Bayesian Network (BN) This model is based on probabilistic and directed acyclic graph theory [13, 160].

Install package bnlearn for bayesian network.

>install.packages("bnlearn")

Pseudocode for training decision tree:

> Include library bnlearn.

> Load dataset.

> Set seed value.

> Divide training and testing data.

> Define training percentage= 70

> Count no. of samples.

> Set target feature.

> Set inputs <- setdiff(names(dataset),target)

> selectedInputs <- Feature Selection.

Generate training data.

*> trainSample <- sample(totalDataset, totalDataset * training/100)*

> trainDataset <- dataset[trainSample,c(selectedInputs, target)]

Generate testing data

> testSample <- setdiff(seq-len(nrow(dataset)), trainSample)

> testDataset <- dataset[testSample,c(selectedInputs, target)]

Build model for training

> formula <- as.formula(paste(target,“ ”,paste(c(selectedInputs),collapse = “+”)))

Train bayesian network.

> model <- bn.fit(formula, trainDataset)

Extract predicted values.

>Predicted <- predict(model, testDataset, type=“class”)

Extract actual values.

>Actual <- as.double(unlist(testDataset[target]))

Evaluate model accuracy.

*>accuracy <- round(mean(Actual==Predicted) *100,2)*

4.2.4.2 Ensemble Model

It is not idiosyncratic to claim that the intent of working on an ensemble construction is to revamp the performance of the classification that is indistinguishable from the single classification system.

In the first step, the data is split into training and testing dataset. It is because the performance analysis is done on entirely independent dataset called testing-dataset to avoid over-fitting. Training data T=$\{(x_1, y_1 \ldots x_n, y_n\}$ which consists of n different training examples is constructed from the product of I (input space) $\times$ O (output space). Each training example $x_i, y_i/$ is represented by a set of condition attribute vector C (c_1, c_2, $\ldots$ c_k) and a target class d_i, which may belong to a set of classes D=$\{$d1, d2,$\ldots$ $d_n\}$

The objective is to approximate an unknown function f : I $\rightarrow$ O by generating a function known as hypothesis. The hypothesis function t defined into some hypothesis space H. To meet this objective, machine learning algorithms utilize the training data to search for the best hypothesis function h. Intuitively, an ensemble classifier generates the hypothesis function t from the combination of individual hypothesis $\{t_1, t_2, t_3, \ldots t_e\}$ of e different classifiers using a specific aggregation method. In the second step, random samples are generated to train the classifiers in the model-pool. In the third step, initially, a preliminary-ensemble is generated by combining m ($1 \leq$ m ≤ 10) different models by random using majority voting aggregation technique. TOPSIS multi-critera decision score of preliminary-ensemble is evaluated using different performance measures. In the next iteration, a new ensemble is produced using different training samples, and with a new set of models in model-pool. The performance of this ensemble is compared with the preliminary ensemble, and the model with the higher TOPSIS score is saved. This phase is called optimization for the best model search. At the end n iterations (say n= 5000 for small dataset), a final ensemble with the highest performance score is declared as the Winning-Ensemble. This is presented in Algorithm 4.1.

4.2.5 TOPSIS Performance Evaluation

For comprehensive evaluation of ensemble model performance, multiple performance metrics should be considered. Among the several solutions available, Multi-Criteria Decision making is the most prevalent approach. TOPSIS method is presented in the Algorithm 4.2 and already discussed in detail in Chapter 3 (Section 3.1.2.4).

Algorithm 4.1 Algorithm

Input: Problem dataset

Output: Print the base classifiers of winning ensemble model.

{Comment: decision file contains the predictions of the selected classifiers from the reduct-pool. topsis stores the value of evaluation score calculated by TOPSIS-algorithm. *columns* stores the count of classifiers selected for building ensemble. After 1000 iterations, base classifiers for the winning ensemble with the highest topsis score are printed. }

1. READ the *decision file*.
2. INITIALIZE topsis=-9999.
3. Set columns = dim(dataset)[2]
4. newS=10
5. FOR (i in 1:5000)
   {
6. n=round(runif(1, min = 1, max = columns-1),0)
7. s=sample(2:columns,n)
8. IF(length(s)$\neq$1)
   {
9. df $\leftarrow$ data.frame(dataset[,s])
10. r $\leftarrow$ data.frame(apply(df[,1:length(df)], 1, mfv))[1,]
    }
11. else
    {
12. r=df $\leftarrow$ data.frame(dataset[,s])
    }
13. Calculate accuracy, Type I error, Type-II error, MCC, F1 score, AUC.
14. topsis= Call TOPSIS (Error rate, Sensitivity, Specificity, MCC, F1 score, AUC)
15. IF (score $<$ topsis)
    {
16. score = topsis
17. newS=s
    }
    }
18. print(names(dataset[,newS]))

Algorithm 4.2 TOPSIS Algorithm

Input: Decision matrix.
Output: Print the score of ensemble-model.
Comment: {Consider m {1 ≤m ≤ 10}models are evaluated against multiple criteria i.e. accuracy, sensitivity, specificity, etc. For simplicity, each criteria has been assigned with equal weight. Each ensemble combination is evaluated for selection of the best.}

1. Standardize the *decision matrix*.
2. Construct *weighted standardized decision matrix* by multiplying evaluation-criteria weight.
3. Determine *ideal solution* i (Ref. Equation 3.3).
4. Determine *negative ideal solution* j (Ref. Equation 3.4).
5. Determine distance from the ideal solution S_i* and negative ideal solution S'_j.
6. Calculate *relative closeness to the ideal solution* C_i* .
7. Set *topsis*= C_i*

4.3 Drug Toxicity Prediction: Case study I

This section discusses the drug toxicity prediction problem case study, need of the problem, and dataset description. The complete process from data collection to implementation of the prediction engine for the problem is discussed further.

4.3.1 Problem Description

Due to ample drug molecular descriptors and properties, harnessing such big data with a high dimensionality, demands inventive methods [34]. With increasing demand of prediction models among the researchers, machine learning techniques have drawn much attention by offering advanced statistics and automated environment. In real applications, data suffers from noise, missing values, errors, inconsistencies, etc. Cleaning and preparing data is pre-eminent part of any data analysis. The practice is quite time-consuming than the data analysis itself.

But, the collected drug data are generally found to be imbalanced and noisy. Careful methods are required while applying feature selection techniques. For instance, if a predictive model:

$$c_1X_1 + c_2X_2 + c_3X_3 \ldots . c_nX_n = P$$

Where X_i are features and c_i are optimized weights given to each feature and P is the predictor

output. When the value X is in thousands and millions, the dire requirement is to correlate these features and select the optimal features, relevant to the data modelling process. The main intent is to construct an ensemble classifier to predict a new drug into a class (toxic or non-toxic), based on the different chemical descriptors of a drug molecule [138].

4.3.2 Need of Drug Toxicity Prediction

Drugs are small organic molecules administered to instigate any biological problem. Biologically, it is a ligand designing which involves a technique of designing a molecule that can bind tightly to the target. Identifying targets, stratifying patients, monitoring drug toxicity, and safety are the key research problems of drug designing and development. Determining the toxicity of a chemical compound is of crucial importance in order to minimize our exposure to many harmful substances in everyday products. For instance, ten thousand additional chemicals like preservatives, flavors, etc. are allowed to put into the food products in the United States (US) [236].

Toxicity is also a central issue in the development of the new drugs, with more than 30 percent of the drug candidates failing in clinical trials because of undetected toxic effects [114]. Therapeutic prediction and decision making regarding the drug toxicity is an eminent research problem to effectuate clinical outcomes [237]. Government agencies NIH, EPA etc. instigated the Tox21 data challenge to encourage the ingenious computational techniques for the drug toxicity prediction [43]. The main intent is to classify a new drug into a class (toxic or non-toxic), based on the different chemical descriptors of a drug molecule.

Ongoing techniques of toxicity assessment for testing a vast number of chemicals are based on conducting a great number of experiments of High-Throughput Screening (HTS) [96]. Dose response curves are analyzed to investigate the toxicity of a specific concentration of chemical compound. HTS is a very rigorous process as it is need to be iterated plenty number of times, and thus entailing million dollar efforts. Therefore, an efficient computational and prediction models for the toxicity prediction are highly urged in this research area.

4.3.3 Dataset

Drug data is collected from TOX21 database containing 1444 different drugs chemical descriptors of 200000 different drug sample molecules. The state parameter is included as a binary-feature that helps in declaring whether the drug molecule is toxic or not, which is built on measuring the different specifically designed chemical descriptors.

4.3.4 Proposed Solution

This section discusses the implementation of MCTOPE framework for prediction of toxicity of a drug molecule.

4.3.4.1 Feature Extraction

2-D structure of 200000 drug molecules are collected as a Structured Data File (SDF) from the database of the chemical molecules. PaDEL is a free and an open source software. PaDEL calculates molecular descriptors of the drug molecule from their chemical structures. The output of the PaDEL can be fed to the proposed framework for the toxicity prediction [25]. Based on the extracted molecular descriptors, drug molecules are predicted to be toxic or not.

4.3.4.2 Prediction Engine

The goal is to avoid class-imbalance issue along with significantly improving the accuracy of the prediction using an innovative feature selection and classification approaches. The abstract view of the proposed framework and the detailed view of the prediction model are discussed here.

The abstract view of the prediction model is presented in the Figure 4.7. The chemical descriptors like molecular weight, surface area, donor count, etc. will act as the inputs of the framework for predicting the toxicity class of the drug molecule. The outcome of the framework will efficiently predict the activity of the newly discovered chemical compound. It is employed as a decision support system as described in Figure 4.8. Decision support system assists the decision maker in predicting the toxic non-toxic class of the drug molecule.

Algorithm 4.3 Prediction Procedure

Comment: {Let F is the features of data to be used for designing of drug toxicity prediction model. Winning model is the combination of the best performing feature subset integrated with the best performing ensemble model }

1. Collect drug samples in SDF format.
2. Extract drug features.
3. Apply data cleaning process using 'data-preparator'.
4. Balance the class using 'class-balancer'.
5. Calculate the correlation coefficient matrix of the features.Drop one feature with correlation greater than 0.75.
6. Rank the features using 'feature-ranker'
7. Remove irrelevant feature.
8. Find best performing ensemble using MCTOPE.
9. Train the ensemble model.
10. Test the performance of model using K fold cross validation.

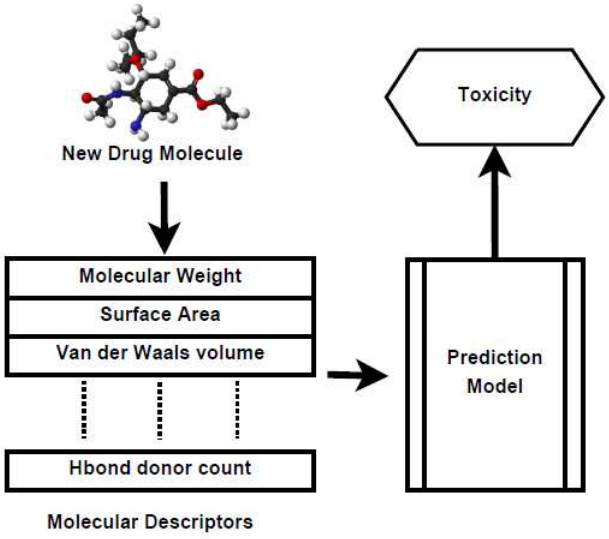

Figure 4.7: Prediction Method

For the sake of clarity, algorithms is described in the Algorithm 4.3.

4.4 Audit Fraudulent Firm Prediction: Case study II

This section discusses the fraudulent firm prediction problem of audit, need of the problem, and dataset description. The complete process from data collection to implementation of the

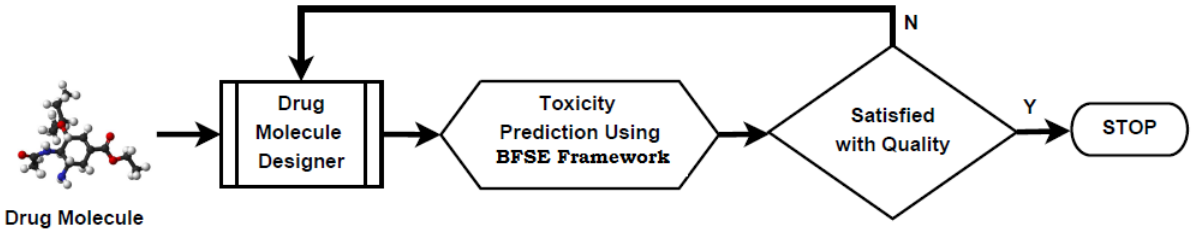

Figure 4.8: Proposed method as decision support system

prediction engine for the problem is discussed further.

4.4.1 Problem Description

Fraud is a critical issue worldwide. Firms that resort to the unfair practices without the fear of legal repercussion have a grievous consequences for the economy and individuals in the society. Auditing practices are responsible for the fraud detection. Audit is defined as the process of examining the financial records of any business to corroborate that their transactions should be in compliance with the standard accounting laws. The prime goal of an auditor during an audit-planning phase is to follow a proper analytical procedure to impartially and appropriately identify the firms that resort to high risk of unfair practices.

Predictive analytics is also implemented using machine learning methods because it provides actionable insights for the audit companies. One of the most common applications of predictive analytics in audit is the classification of suspicious firm. The purpose of classifying the firms during the preliminary stage of an audit is to maximize the field-testing work of high-risk firms that warrant significant investigation [139].

4.4.2 Need of Fraudulent Firm Prediction

Audit includes a very exacting task of detecting firms that are spotting frauds, detecting errors, and disclosing employees guilty of abetting illegal transactions. Data analytics tools for an effective fraud management have become the need of the hour for an audit.

Generally, audits are classified as internal and external auditing [17]. Internal-audit, although is an independent department of an organization, but resides within the organization.

These are company-employees that are accountable for performing audits of financial and non-financial statements as per their annual audit plan. External audit is a fair and independent regular audit authority, which is responsible for an annual statutory audit of financial records.

The external audit company has a fiduciary duty and are critical to the proper conduct of business. For instance, their work is to audit the receipts, expenditures, accounts related to the trading, profit, contingency funds, balance sheets, public accounts, etc. kept in any government office. It is their duty to ensure that the funds allocated to any government department have been put to use as per law. On successful completion of an audit process, auditors deliver an audit and inspection summary report called audit paras to the company comprising of the details of all the findings from the audit. This may include discrepancies, non-compliance of accounting rules, leakages of revenue, inaccurate calculations, etc. The whole audit process flow is summarized in the Figure 4.9. Auditing Standards Board Task Force (ASBTF) is also working on developing an innovative Audit Data Analytics Guide in order to integrate the data analytics tools for the auditing tasks.

The prime goal of an auditor during an audit-planning phase is to follow a proper analytical procedure to impartially and appropriately identify the firms that resort to high risk of unfair practices. Predictive analytics is also implemented using machine learning methods because it provides actionable insights for the audit companies. The purpose of classifying the firms during the preliminary stage of an audit is to maximize the field-testing work of high-risk firms that warrant significant investigation. According to a research, data analytics has benefited internal auditing more as compare to advancements it has contributed for the external audits [273]. This research work is a case-study of an external government audit company which is also an external auditor of government firms of India. During audit-planning, auditors examine business of different government offices but target to visit the offices with very-high likelihood and significance of misstatements. This is calculated by assessing the risk relevant to the financial reporting goals [191].

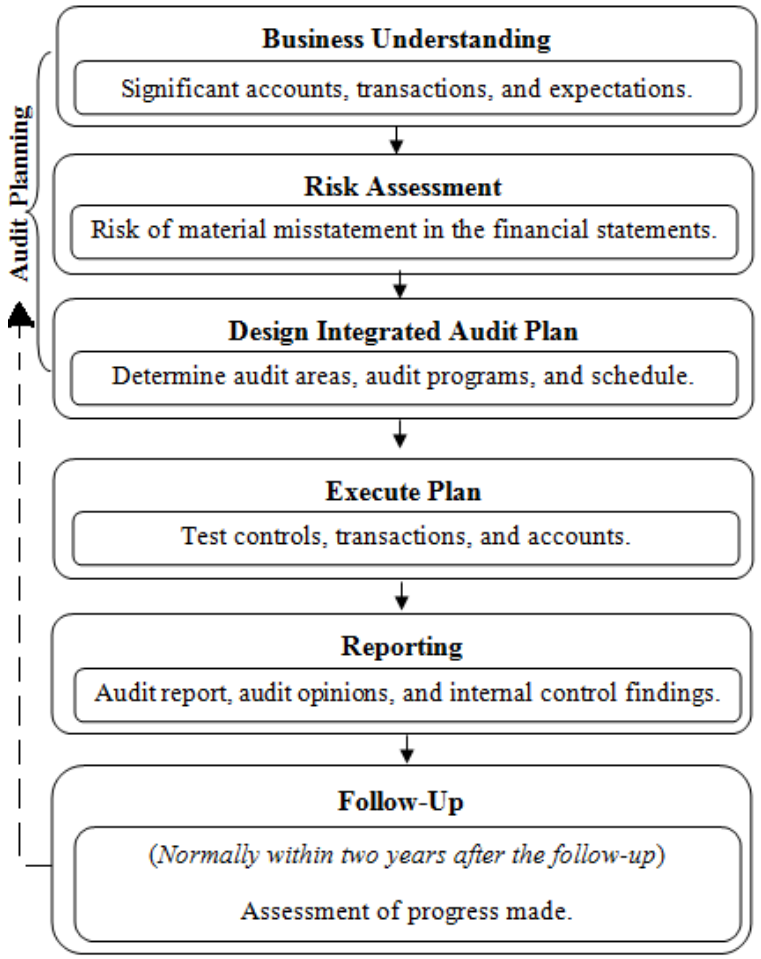

Figure 4.9: Audit Work-Flow

4.4.3 Dataset

Data is collected from an audit office that audits receipts and expenditure of all the firms that are financed by the government of India. While maintaining the secrecy of the data, exhaustive one year non-confidential data $(2015-2016)$ of firms is collected from the Auditor General Office (AGO). There are total 777 firms from 46 different cities of a state that are listed by the auditors for targeting the next field-audit work. The target-offices are listed from 14 different sectors. The information about the sectors and their counts are presented in the Table 4.2.

Table 4.2: Target Sectors

Sector ID	Target sector	Information	Number of target firms
1	IR	Irrigation	114
2	P	Public Health	77
3	BR	Buildings and Roads	82
4	FO	Forest	70
5	CO	Corporate	47
6	AH	Animal Husbandry	95
7	C	Communication	1
8	E	Electrical	4
9	L	Land	5
10	S	Science and Technology	3
11	T	Tourism	1
12	F	Fisheries	41
13	I	Industries	37
14	A	Agriculture	200

4.4.4 Proposed Solution

This section discusses the implementation of MCTOPE framework for the fraudulent firm prediction in the auditing.

4.4.4.1 Feature Extraction

Many risk factors are examined from various areas like past records of audit office, audit-paras, environmental conditions reports, firm reputation summary, on-going issues report, profit-value records, loss-value records, follow-up reports etc. After in-depth interview with

the auditors, important risk factors are evaluated and their probability of existence is calculated from the present and past records. Table 4.3 describes the various examined risk-factors that are involved in the case study. Various risk factors are categorized, but combined audit risk is expressed as one function called an Audit Risk Score (ARS) using an audit analytical procedure. At the end of risk assessment, the firms with high ARS scores are classified as "Fraud" firms, and low ARS score companies are classified as "No-Fraud" firms. Sample audit data of the corporate sector is shown in the Table 4.4. Audit Risk Assessment (ARA) is a deliberate process of evaluating the likelihood of discrepancies or misstatements (event E) [148]. Risk is often measured as the expected value of any an unenviable outcome. During audit-planning, external auditors first quantitatively evaluate the risk of fraud in an organization in order to estimate the need of audit-field work. As Event E is a negative event, so historical data is also analyzed. For calculating the probability of discrepancies or misstatements (event E), formal methods are preferred. The associated formula for calculating the risk R is expressed as

$$R = (P_{\xi}(L_{\xi})) \tag{4.2}$$

Where, P_{ξ} is the probability of discrepancy and (L_{ξ}) is the loss involved in the discrepancy.

Table 4.3: Risk factors classification and other features in model

Inherent Risk Factors		**Control Risk Factors**	
Feature	Information	Feature	Information
Para A Value	Discrepancy found in the planned-expenditure of inspection and summary report A in Rs (in crore).	Sector Score	Historical risk score value of the target-unit in the Table 4.2 using analytical procedure.
Para B Value	Discrepancy found in the unplanned-expenditure of inspection and summary report B in Rs (in crore).	Loss	Amount of loss suffered by the firm last year.
Total	Total Amount of discrepancy found in other reports Rs (in crore).	History	Average historical loss suffered by firm in the last ten years.
Number	Historical discrepancy score.	District score	Historical risk score of a district in the last ten years.
Money Value	Amount of money involved in misstatements in the past audits.		
Other features			
Feature	Information	Feature	Information
Sector ID	Unique ID of the target sector.	Location ID	Unique ID of the city/province.
ARS	Total Risk Score using analytical procedure.	Audit ID	Unique Id assigned to an audit case.
Risk class	Risk Class assigned to an audit-case. (*Target Feature*)		

Due to different categories of risk, situations in an audit are sometimes more complex than the simple possibility case of one risk. In a situation with several possible risk types, total risk is the sum of the different risk type and can be expressed as

$$R = \sum_{i} (P_{\xi}(L_{\xi})) \tag{4.3}$$

Where, i is the count of considered risk types.

When the audits are performed by any external audit company, the risk assessment assists in deciding the amount of field work that would be required before actually visiting the official firms. According to ISA315, an auditor should always obtain a clear understanding

of the firm including all its internal environments, controls, entities, etc. for a complete risk assessment before actually visiting the firm [242]. This process acts as an initial evidence for performing an effective audit at client's firm. As a formula, audit risk is the product of inherent risk (IR), control risk (CR) and detection risk (DR) [253]. It can be calculated as

$$\begin{aligned} AR &= IR \times CR \times DR \\ &= Combined\ Risk \times DR \\ &= (P_{IN}L_{IN} \times P_{CO}L_{CO}) \times DR \end{aligned} \tag{4.4}$$

Inherent Risk (IR) is the risk present due to the discrepancies present in the transactions. For instance, transaction which involves settlement by checks have lower IR as compared to the transaction which involves exchange of cash. CR is the risk due to the discrepancies which are left undetected by internal control system. For instance, CR risk is high when the separation of duties are not properly defined. DR is the risk of discrepancies present in the firm which are not even detected by the audit procedures. Human or sampling error, for instance. Considering all risk factors, complete equation for evaluating an audit risk using Equation (4.3) and Equation (4.4) can be expressed as

$$AR = \sum_{i=1}^{\alpha} P_{IN}L_{IN} \times \sum_{i=1}^{\beta} (P_{CO}L_{CO}) \times DR\{\eta = \alpha + \beta\} \tag{4.5}$$

Where α and β are the number of risk factors causing inherent risk and control risk respectively. For this case study, the complete equation for the risk factors (risk factors categorized in the Table 4.3 can be expressed as

$$\begin{aligned} AR = ((P_{PARA\ A}L_{PARA\ A}) + (P_{PARA\ B}L_{PARA\ A}) + (P_{Total}L_{Total}) + (P_{Number}L_{Number}) \\ + (P_{Money\ Value} + L_{Money\ Value}) \times (P_{Sector\ Score}L_{Sector\ Score}) \\ + (P_{District}L_{District}) + (P_{History}L_{History}) + (P_{Loss}L_{Loss})) \times DR \end{aligned} \tag{4.6}$$

For calculating the audit risk of a firm, probability of each risk factor is calculated using analytical procedure and an audit risk score is calculated for each firm. In order to understand the complete step by step process, it is presented as a Risk Assessment Algorithm 4.4.

Algorithm 4.4 Risk Assessment Algorithm

Input: Agenda list A.

Output: Risk class of each sample in the agenda list A.

Comment: { Let X denote the number of offices to be examined. Assume X is finite. Let A is a list of nominated set of offices called Agenda A, extracted from set X (Here Agenda is a set of all offices that are not being visited since last three years).}

1. Data collection :Collect the unstructured data of all the offices under Agenda A.
2. Feature Extraction: Examine k features (risk factors) ξ_k that may be needed for the inherent and control risk assessment.
3. Risk Calculation:
   - Calculate the loss L_ξ that may be involved for each risk factor.
   - Calculate the probability of loss (p_ξ) for each risk factor ξ.
   - Calculate the risk (R_ξ) for each risk factor ξ as $R = (P_\xi(L_\xi))$.
4. Risk Classification : Classify the risk factors into inherent-risk class(IN) and control risk class (CO) and calculate the sum of risk for each class as Sum_{IN} and SUM_{CO} respectively.
5. Combined Risk: Calculate the combined risk of class CO and IN as $Combined\ Risk = (SUM_{IN}) \times (SUM_{CO})$
6. Detection Risk: Define the detection risk value DR.
7. Audit Risk Score: Calculate the audit risk as the $Audit\ Risk = ((Combined\ Risk) \times DR)$
8. Risk Assessment: Calculate the average of audit risk as $Audit_{avg}$. Classify the audit risk a_i for each audit case as high (fraud class) and low (no fraud class) by the following rules:
   - *if* the audit risk $(a_i <= 1)$, label it as No Fraud.
   - *else* label it as Fraud.

Table 4.4: Sample data of the corporate sector unit

Audit ID	Loc ID	Para A	Para B	Total	Sector score	Loss Score	Num.	Money Value	History score	District Score	ARS	Risk Class
26	4	5.78	57.92	73.70	3.89	0	5	11.16	1	2	4	F
27	4	7.42	2.24	19.66	3.89	1	1	1.25	2	2.5	2.4	F
28	4	0	1.10	4.11	3.89	0	3	0.007	2	2	2	N
29	14	6.85	31.76	58.61	3.89	2	5	1.46	1	4	3.6	F
30	14	0	1.03	5.03	3.89	0	5	0	2	2	1	N
31	37	0	0.75	3.75	3.89	0	5	6.78	2	2	2.2	F
32	37	2.4	16.63	29.73	3.89	0	3	1.16	0	4	3.6	F
33	5	0	0.05	1.23	3.89	1.3	5	152.41	2	2	2.4	F
34	5	0	1.76	4.76	3.89	0	2	1.08	2	2	2	N
35	5	0	2.97	6.97	3.89	0	5	2.84	1	2	2	N

4.4.4.2 Prediction Engine

The intent is to design and implement a prediction model for the proposed audit field work decision support framework. The proposed framework which can also work as a Decision Making System is presented in an abstract view in the Figure 4.10.

The selected features (as described in the Table 4.3) are used as candidates for the input vector of the model. The outcome of the proposed framework helps an auditor to predict an audit risk class (Fraud or No Fraud). The complete flow of the prediction model for the proposed audit field work decision support framework is described in the Figure 4.7.

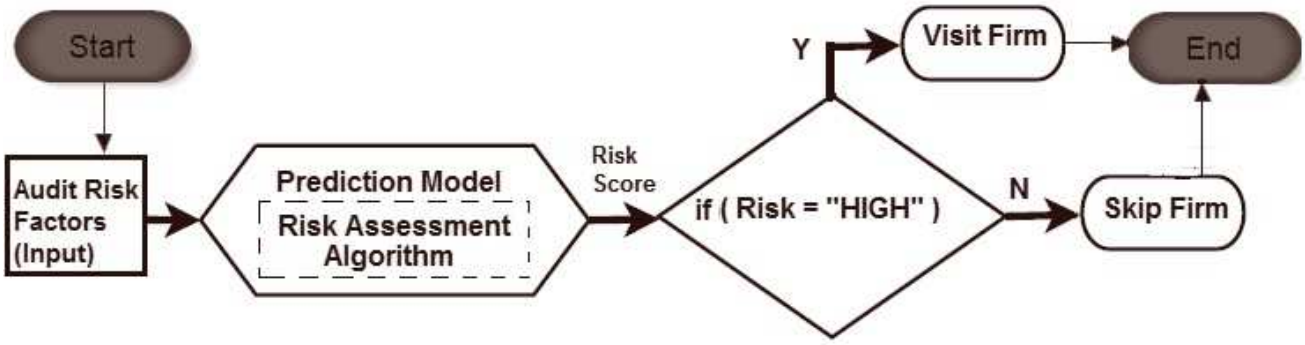

Figure 4.10: Proposed framework for an Audit Field Work Decision Making

The values of audit risk factors are sent as input to the Audit Field Work Decision Support System. Risk Assessment trained based prediction engine predicts the risk of the fraud for the firm and classify the firm as high or low risk firm. If the risk class of the firm is 'high', the auditors must visit the firm. If the risk class is low, the auditors can skip the firm.

Chapter 5

Test and Comparative Analysis

This Chapter presents the testing details of the MCTOPE framework on different machine learning prediction problems. The proposed method is first tested on simple prediction problems to evaluate it's performance on small datasets, and then tested on the investigated case-studies. K-fold cross validation technique is used to calculate the performance metrics of the built ensemble. To test the robustness of the built ensemble, the process is iterated several times. Different parameters namely accuracy, sensitivity, specificity, MCC, F measure, and area under curve (AUC) are used. TOPSIS performance score, a comprehensive performance evaluation approach is employed to compare the built ensemble with the state-of-the art methods.

5.1 UCI Datasets

Initially, the MCTOPE framework is validated on publicly available datasets of UCI machine learning data repository[9]. The detailed description of these datasets is shown in the Table 5.1 .

The experimental results are presented with the help of TOPSIS score graph. On focusing a single criteria, results may not appear significantly the best. For comprehensive assessment, TOPSIS performance scores of the classifiers are compared in Figure 5.1, Figure 5.2, Figure 5.3, Figure 5.4, Figure 5.5, and Figure 5.1. It is clearly visible that the performance of the ensemble framework is exceptionally good on all six UCI prediction problems as discussed

Table 5.1: Description of validation dataset for validation for proposed framework

Dataset	Sample	Attributes	Ref.
Australian Credit Approval	690	15	[19]
German Credit Data	1000	25	[20]
Bank note	1372	5	[241]
Pop failure Networking Data	540	21	[21]
Wholesale Customers Data	440	8	[23]
Sonar	208	61	[22]

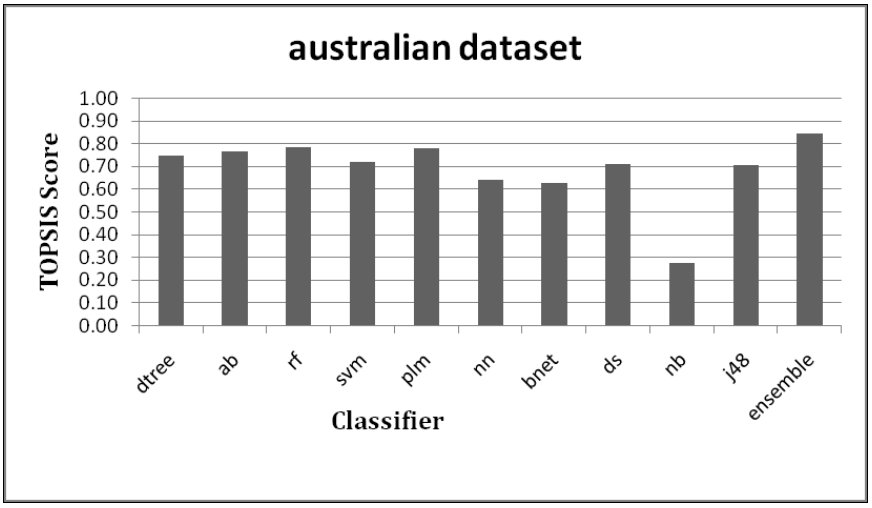

Figure 5.1: Topsis Score Analysis of Australian Dataset

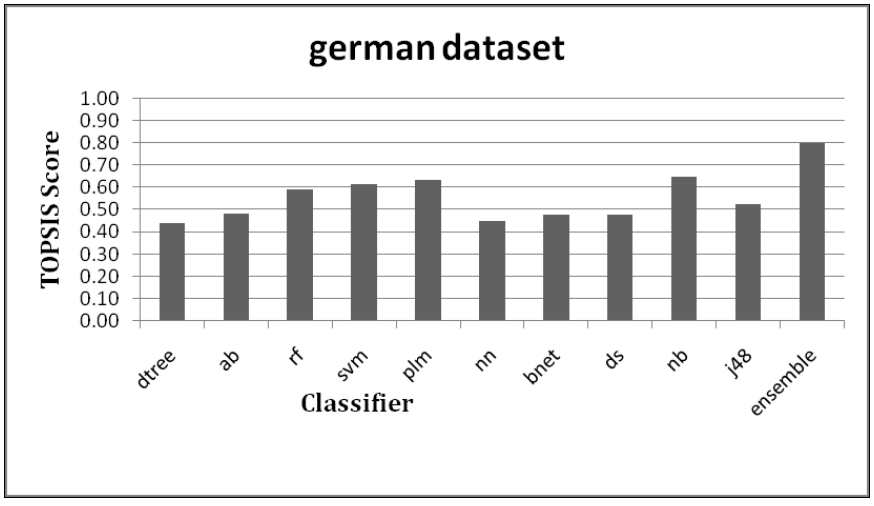

Figure 5.2: Topsis Score Analysis of German Dataset

below.

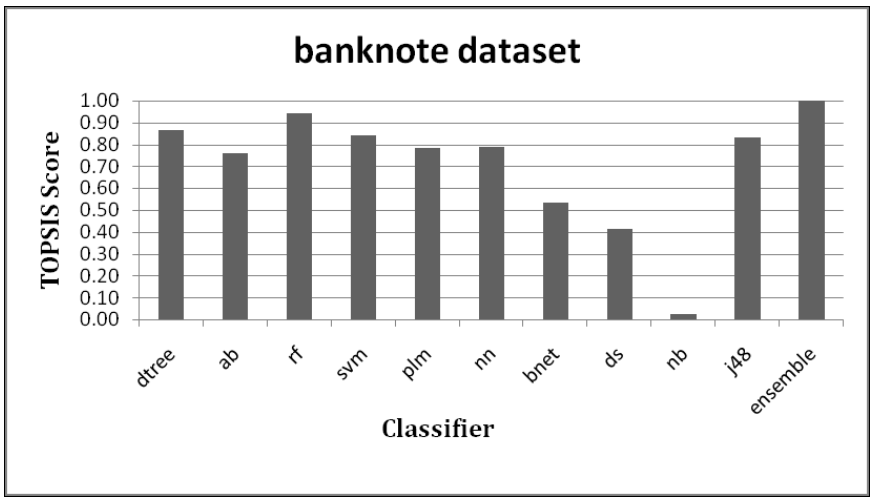

Figure 5.3: Topsis Score Analysis of Banknote Dataset

i. Australian dataset is an interesting dataset related to finance domain applied for credit card approval. The data consists of 690 samples with 15 feature instances. The data is useful because it contains missing values, values of different nature like continuous, integral, etc. In Figure 5.1, an ensemble is constructed using Naive Bayes and J48 algorithm. If we compare the topsis performance score of popular classifiers like SVM, neural network, etc. with the performance of an ensemble, topsis score of an ensemble is considerably better than the base classifiers.

ii. German dataset is an interesting dataset related to finance domain applied for credit card approval. The data consists of 1000 samples with 25 features. The data is useful because it contains missing values, values of different nature like continuous, integral, etc. In Figure 5.2, an ensemble is constructed using Neural Network and J48 algorithm. If we compare the topsis performance score of popular classifiers like SVM, neural network, etc. with the performance of an ensemble, topsis score of the built ensemble is considerably better than the base classifiers.

iii. Bank note dataset is an interesting dataset as it is extracted from images. It is related to banking domain applied for authentication of notes of bank. The data consists of 1372 samples with 5 features. The data is useful because it contains missing values, values of

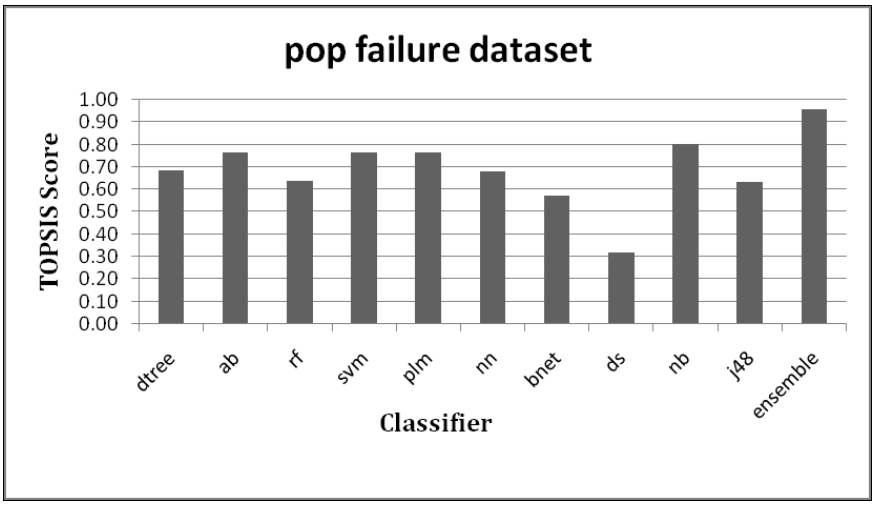

Figure 5.4: Topsis Score Analysis of Popfailure Dataset

different nature like continuous, real, integral, etc. In Figure 5.3, an ensemble is constructed using naive bayes and decision tree algorithm. Comparing topsis score of popular classifiers like SVM, neural network, etc. with performance of an ensemble, topsis score of the built ensemble is considerably better than the base classifiers.

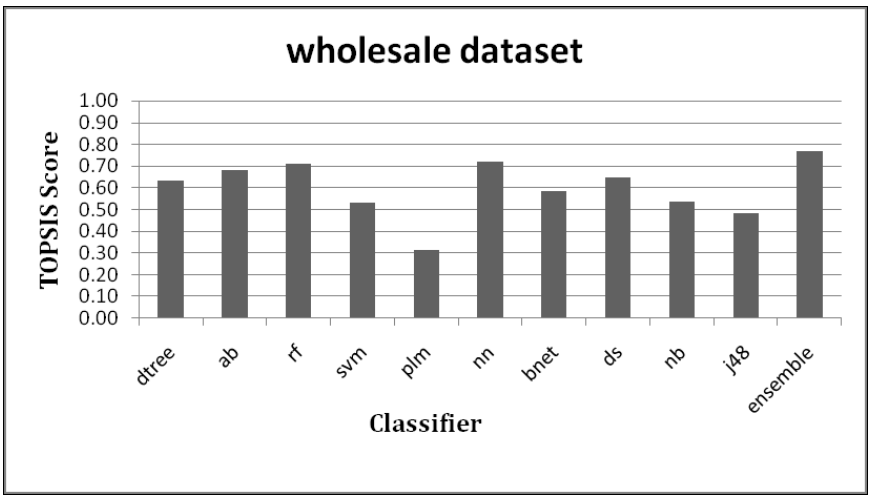

Figure 5.5: Topsis Score Analysis of Wholesale Dataset

iv. Pop Failure dataset is an interesting dataset as it is extracted from the measurements of robot after their failures. It is works on measuring torque and force at regular time-intervals. The data consists of 540 samples with 21 features. The data is useful because it contains missing values, values of different nature like continuous, real, integral, etc. In Figure 5.4, an ensemble is constructed using bayes net and decision tree algorithm. On comparing the

topsis performance score of popular classifiers like SVM, neural network, etc. with the performance of an ensemble, topsis score of built ensemble is considerably better than the base classifiers.

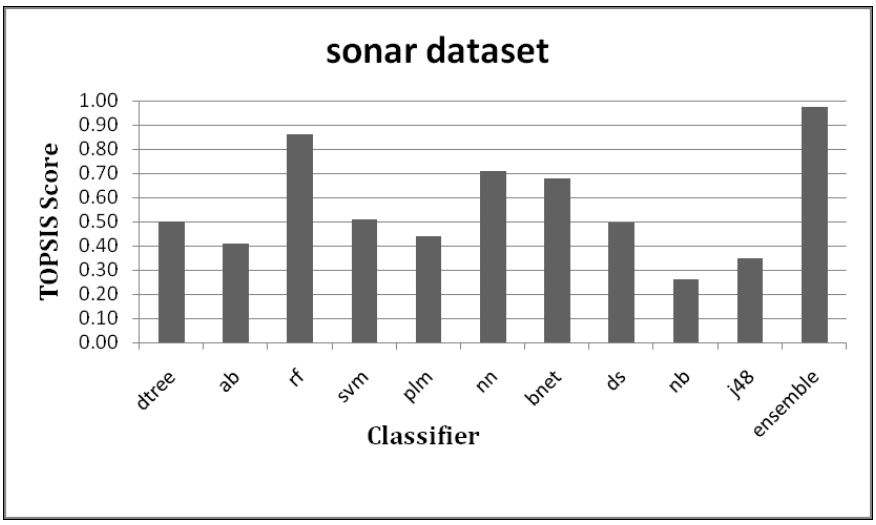

Figure 5.6: Topsis Score Analysis of Sonar Dataset

v. Wholesale dataset is an interesting dataset as it is extracted from the data of clients of a wholesale distributor. It is related to sales and marketting domain applied to unfold the information of clients of a distributor. The data consists of 440 samples with 8 features. The data is useful because it contains missing values, values of different nature like continuous, real, integral, etc. In Figure 5.5, an ensemble is constructed using PLM and J48 algorithm. If we compare the topsis performance score of popular classifiers like SVM, neural network, etc. with ensemble performance, topsis score of ensemble is considerably better than the base classifiers.

vi. Sonar dataset is an interesting dataset where the goal is to classify and predict the sonar signals. It is related to rock mining domain. The data consists of 208 samples with 61 features. The data is useful because it contains missing values, values of different nature like continuous, real, integral, etc. In Figure 5.6, an ensemble is constructed using naive bayes and J48 algorithm. If we compare the topsis performance score of popular classifiers like SVM, neural network, etc. with ensemble performance, topsis score of ensemble is considerably better than the base classifiers.

5.2 Testing Drug Toxicity Predictor

This section discusses the the experimental results, K fold cross validation testing results, the comparative analysis and the testing of a new drug molecules for AIDS therapy.

5.2.1 K-Fold Cross Validation

10 fold cross validation technique is adopted for experiments. 10 fold cross validation divides data into 10 equal size subsets. Ensemble algorithm is built to train the nine subset folds and testing on the last subset fold. To test the robustness of MCTOPE framework, process is iterated. Different parameters namely accuracy, sensitivity, specificity, F measure, MCC and Area under curve (AUC) are used.

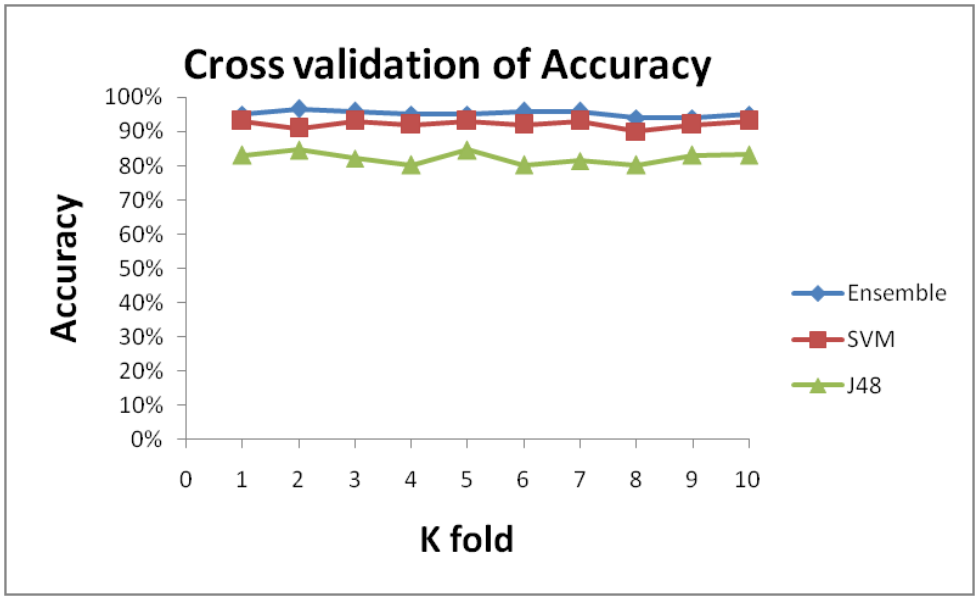

Figure 5.7: K Fold Cross Validation results of accuracy of built ensemble and base classifiers

i. Accuracy of built ensemble using SVM and J48 as candidates is graphically depicted on each fold of 10-cross validation method in the Figure 5.7.

ii. It can be observed that an ensemble accuracy is better than accuracy of candidate classifiers and it is quite robust as value of accuracy is not changing abruptly.

iii. Sensitivity of built ensemble using SVM and J48 as candidates is graphically depicted on each fold of 10-cross validation method in the Figure 5.8.

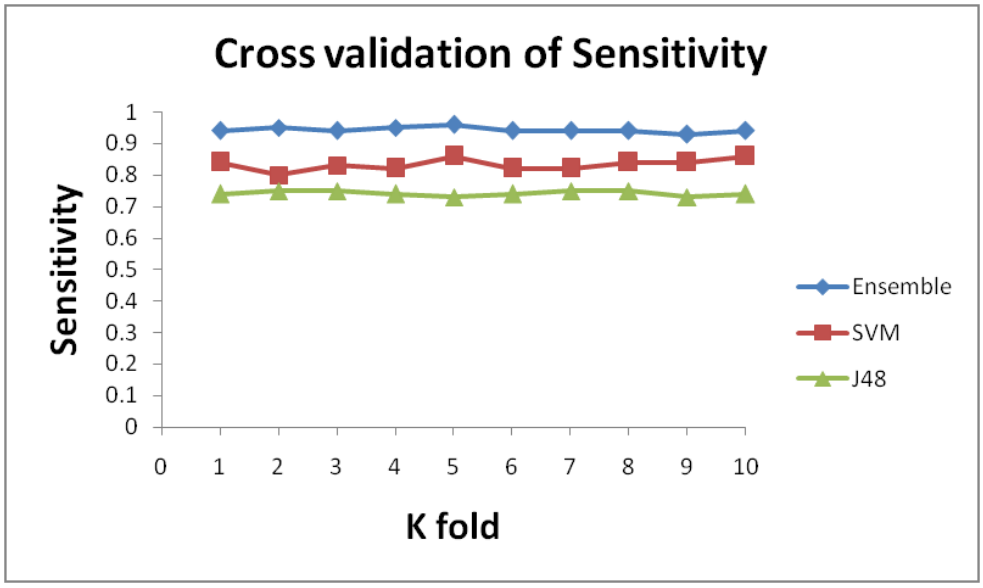

Figure 5.8: K Fold Cross Validation results of sensitivity of built ensemble and base classifiers

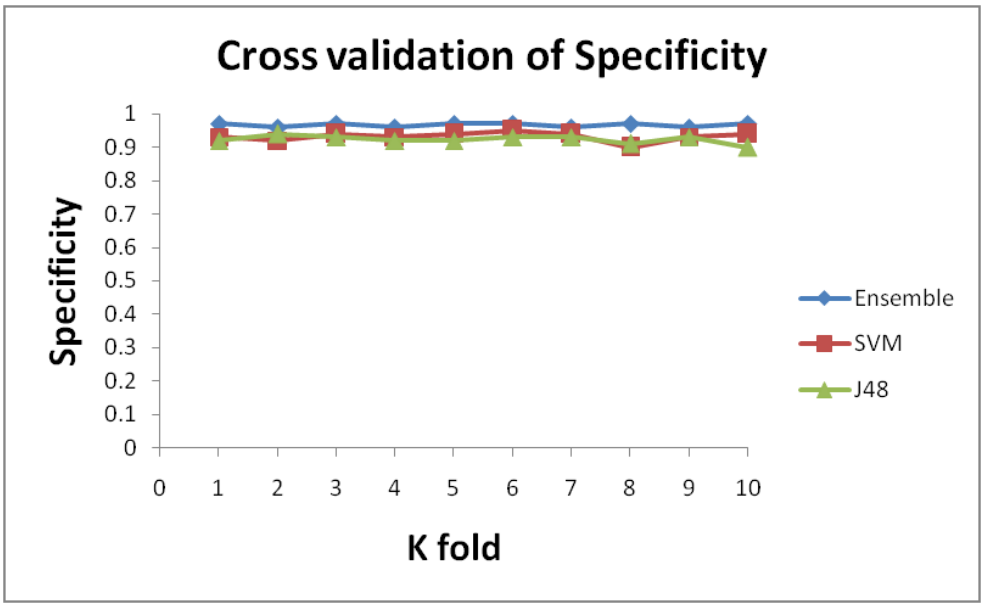

Figure 5.9: K Fold Cross Validation results of specificity of built ensemble and base classifiers

iv. It can be observed that the sensitivity of ensemble is better than the candidate classifier's sensitivity and it is quite robust as values are stable and are not changing abruptly.

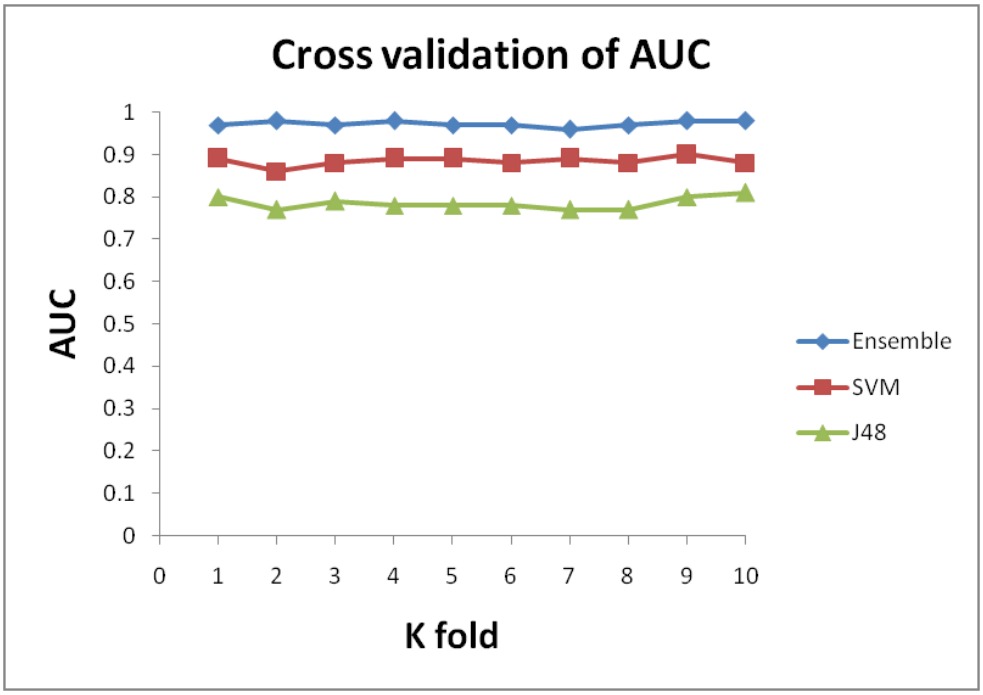

Figure 5.10: K Fold Cross Validation results of specificity of built ensemble and base classifiers

v. Specificity of built ensemble using SVM and J48 as candidates is graphically depicted on each fold of 10-cross validation method in the Figure 5.9.

vi. It can be observed that sensitivity of ensemble is better than candidate classifier's sensitivity and it is quite robust as values are stable and are not changing abruptly.

vii. Area under the curve value (AUC) of built ensemble using SVM and J48 as candidates is graphically depicted on each fold of 10-cross validation method in the Figure 5.10.

viii. It can be observed that ensemble's sensitivity is better than candidate classifier's sensitivity and it is quite robust as values are stable and are not changing abruptly.

5.2.2 Experimental Results

The five random training dataset is fed to the classifier for testing. The performance of different random sub-samples (RS) are measured and analyzed iteratively for feature selection.

The average performances of top 50, 100, 150, 200 and 400 features for these random samples are also summarized in the Table 5.2. The different evaluation parameters are calculated for random samples of data such as RSI, RSII, RSIII, etc [268]. The results for different random-samples are compared graphically in the performance of accuracy, sensitivity, and specificity in the Figure 5.11, Figure 5.12, and Figure 5.13 respectively.

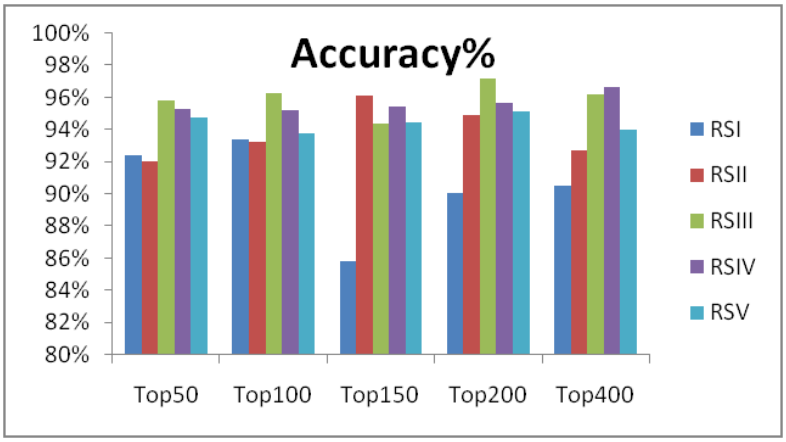

Figure 5.11: Accuracy

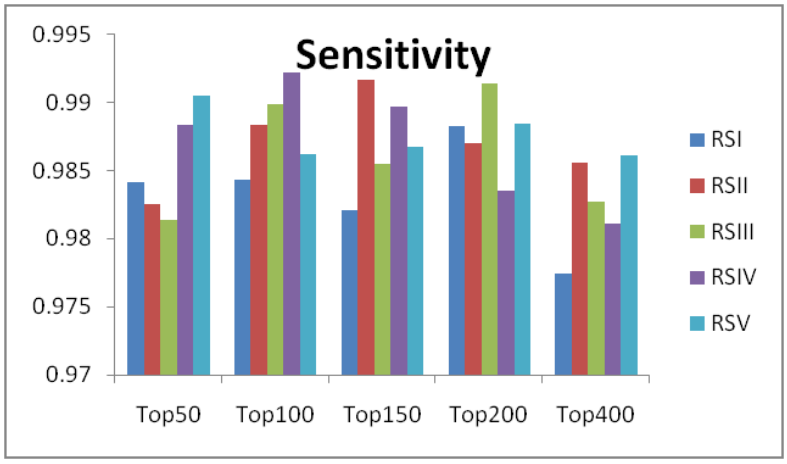

Figure 5.12: Sensitivity

We can observe following from the graphs: (i) Accuracy is the percentage count of correct predictions out of the total count of prediction performed. Accuracy of random sample III (RSIII) with top-200 feature-set is the best.

ii. Sensitivity of the feature-sets 100, 150, and 200 are good. As there is lot of variations in

Table 5.2: Experimental Results

Parameter	Random Subset	Top 50	Top 100	Top 150	Top 200	Top 400
Accuracy	RSI	0.9245	0.9339	0.8585	0.9009	0.9056
	RSII	0.9201	0.9322	0.9613	0.9492	0.9274
	RSIII	0.9583	0.9632	0.9436	0.9718	0.9620
	RSIV	0.9531	0.9519	0.9542	0.9565	0.9668
	RSV	0.9480	0.9382	0.9447	0.9512	0.9398
	Average	0.9408	**0.9438**	0.9324	**0.9459**	0.9403
Sensitivity	RSI	0.9942	0.9943	0.9821	0.9883	0.9774
	RSII	0.9825	0.9884	0.9917	0.9970	0.9856
	RSIII	0.9914	0.9899	0.9855	0.9914	0.9927
	RSIV	0.9884	0.9922	0.9897	0.9935	0.9911
	RSV	0.9905	0.9962	0.9868	0.9885	0.9961
	Average	0.9894	**0.9922**	0.9871	**0.9917**	0.9885
Specificity	RSI	0.4210	0.6388	0.3863	0.5365	0.5428
	RSII	0.6143	0.6471	0.7500	0.7101	0.6154
	RSIII	0.7647	0.8145	0.712	0.8534	0.8015
	RSIV	0.6768	0.6400	0.6596	0.7000	0.7442
	RSV	0.7033	0.6000	0.6786	0.7447	0.6392
	Average	0.6360	0.6680	0.6373	**0.7089**	0.6686
AUC	RSI	0.7076	0.8166	0.6842	0.7624	0.7601
	RSII	0.7984	0.8177	0.8708	0.8536	0.8005
	RSIII	0.8780	0.9022	0.8488	0.9224	0.8971
	RSIV	0.8326	0.8161	0.8247	0.8467	0.8677
	RSV	0.8469	0.7981	0.8327	0.8666	0.8177
	Average	0.8127	0.8301	0.8122	**0.8503**	0.8286
PPV (Precision)	RSI	0.9411	0.9583	0.8500	0.9166	0.8260
	RSII	0.9258	0.9342	0.9650	0.9449	0.9321
	RSIII	0.9611	0.9675	0.9498	0.9761	0.9632
	RSIV	0.9599	0.9552	0.9602	0.9583	0.9726
	RSV	0.9505	0.9356	0.9510	0.9555	0.9365
	Average	0.9476	**0.9501**	0.9352	**0.9502**	0.9260

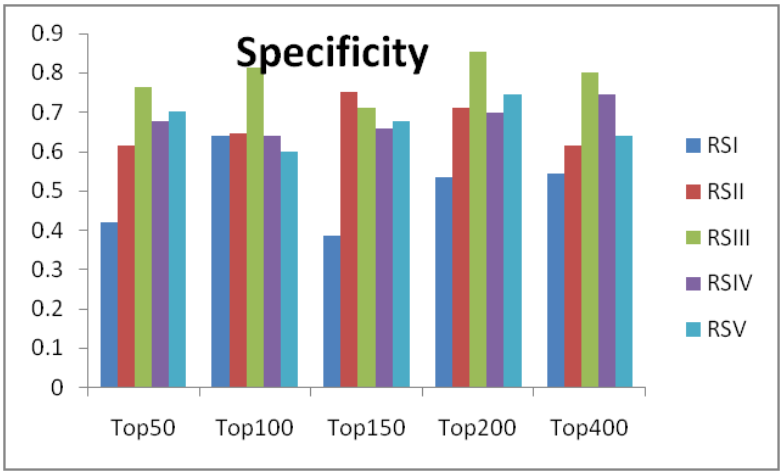

Figure 5.13: Specificity

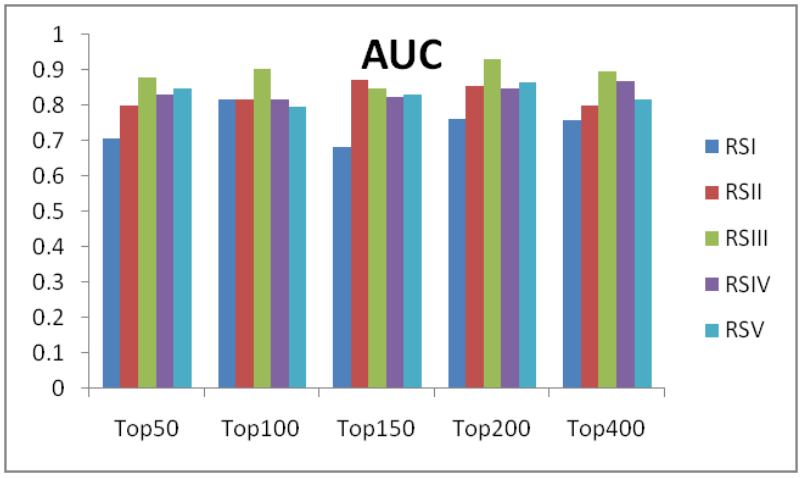

Figure 5.14: AUC

the graph, the sensitivity parameter cannot be used to compare the samples. Specificity of random sample III (RSIII) is constantly good in all feature sets.

The best performing random sample set RS3 with top 200 feature-set have shown the highest performance and can be declared as the winning set. This winning set is employed as training sample in final prediction model.

5.2.3 Comparison with State-of-the-Art Methods

For comparison analysis, state-of-the-art classifiers are explored in the area of drug toxicity prediction and computational biology. The results of the MCTOPE framework is compared with the outstanding classical classifiers like SVM, randomforest, C4.5, adabag, etc and presented in the Table 5.3. The accuracy of the proposed framework is 95%, which is much better than the other classifiers. But, focusing only on the accuracy of the proposed framework can be misleading. Considering other parameters, sensitivity of the classifier which tells us the hit rate or true positive rate which is quite promising. On comparing the specificity or true negative rate and F measure of the classifiers, random forest is performing better than the proposed ensemble. It shows random forest is performing better in detecting true negative cases. For this reason, F measure of the random forest is little better than the other built ensemble. However, focusing completely on the true negative and F measure can be misleading. As it is the case of imbalanced data, MCC and AUC parameters will also be considered. In the results shown in the Table 5.3, there is a significant rise of AUC which is 0.97 when it is compared with the other classical methods. It can also be observed that the value of AUC in the proposed method is quite closer to the best value which is one. It signifies the framework worked perfectly in bringing down the biasing suffered by the data. The prediction method outperforms other techniques hence chosen as the prediction model for the drug toxicity decision support system. In an attempt to perform comprehensive performance evaluation, classifiers are evaluated against multiple criteria or metrics using TOPSIS score. The performance of the proposed method outperforms the other standard frameworks in terms of TOPSIS score performance as presented in the Figure 5.15.

Table 5.3: Experimental Results

Metric	Ensemble	SVM	Rf	J48	Abag	Aboost	NB	BN	DT	PLM	DS
Accuracy	**0.95**	0.93	0.91	0.83	0.88	86.06	0.78	0.86	0.82	87.23	0.74
Sensitivity	**0.94**	0.84	0.84	0.74	0.80	0.85	0.85	0.82	0.72	0.85	0.75
Specificity	**0.97**	0.83	**0.98**	0.92	0.90	0.84	0.70	0.90	0.91	0.88	0.71
F measure	**0.89**	0.84	**0.91**	0.83	0.85	0.85	0.79	0.86	0.81	0.87	0.72
MCC	**0.83**	0.78	**0.83**	0.67	0.75	0.72	0.56	0.72	0.65	0.75	0.45
AUC	**0.97**	0.89	0.96	0.79	0.92	0.90	0.80	0.90	0.81	0.93	0.75

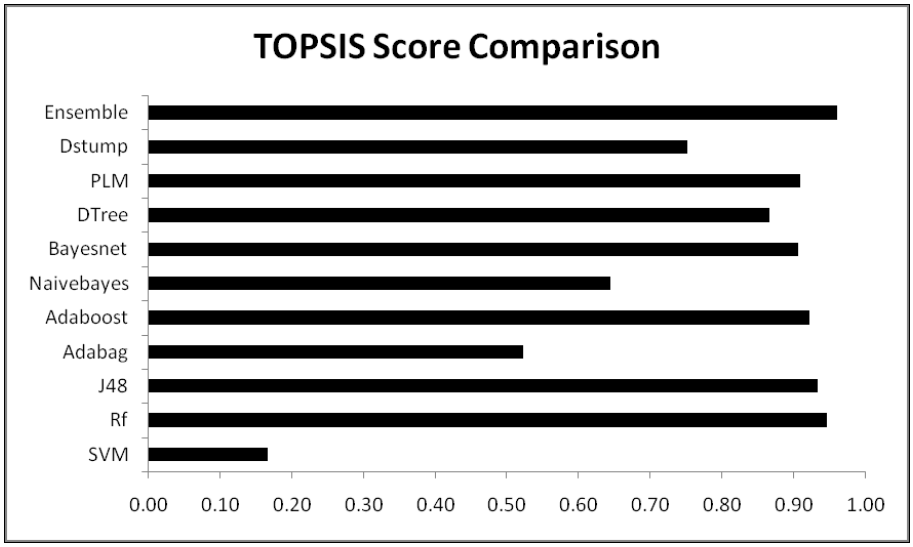

Figure 5.15: TOPSIS Performance Comparison Analysis

5.2.4 Testing AIDS Therapy drug molecules

Nucleoside reverse transcriptase inhibitors (NNRTIs) like nevirapine (NVP), delavirdine (DLV), and efavirenz (EFV) are crucial drugs acquired for the AIDS therapy [92]. These drugs exhibit potent anti-HIV-1 activity and the modest toxicity. Nevirapine is associated with hepatic toxicity and it causes liver injury during therapy [233]. Major toxicity of delavirdine is rash [234]. It is also accompanied by fever, blistering, oral lesions, conjunctivitis, swelling, muscle or joint aches [234]. Efavirenz also has serious life threatening side effects on the liver, and the central nervous system of the body [235].

The complete process of validation is presented in the Figure 5.16. 2-D structure of all the three drug molecules namely, nevirapine (NVP), delavirdine (DLV), and efavirenz (EFV) are downloaded as a Spatial Data File (SDF) from the PubChem database of the chemical molecules [275].

PaDEL is a free and an open source software for calculating molecular descriptors of the drug molecule from their chemical structures. The output of the PaDEL can be fed to the proposed framework for the toxicity prediction [25]. The output of proposed framework is depicted in the Table 5.4. Based on the extracted molecular descriptors, drug molecules are predicted to be toxic. The correct prediction results serve as a proof of validity of MCTOPE framework to perform an efficient toxicity prediction.

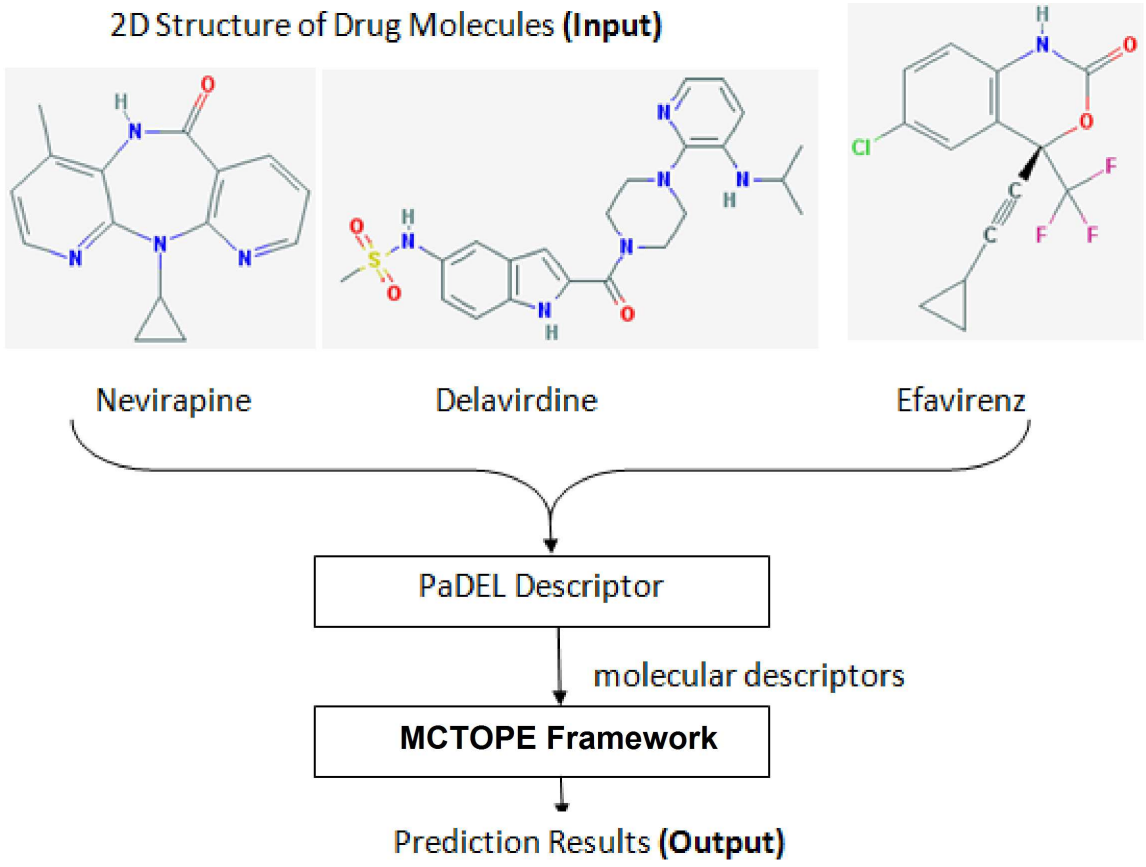

Figure 5.16: Toxicity prediction of AIDS therapy drug molecules

Table 5.4: Toxicity prediction results

Target Drug molecule	Actual Class	Predicted Class	Accuracy
Nevirapine (NVP)	T	T	100%
delavirdine (DLV)	T	T	100%
efavirenz (EFV)	T	T	100%

5.3 Testing Audit Fraudulent Firm Predictor

This section discusses the experimental results of K-fold cross validation method.

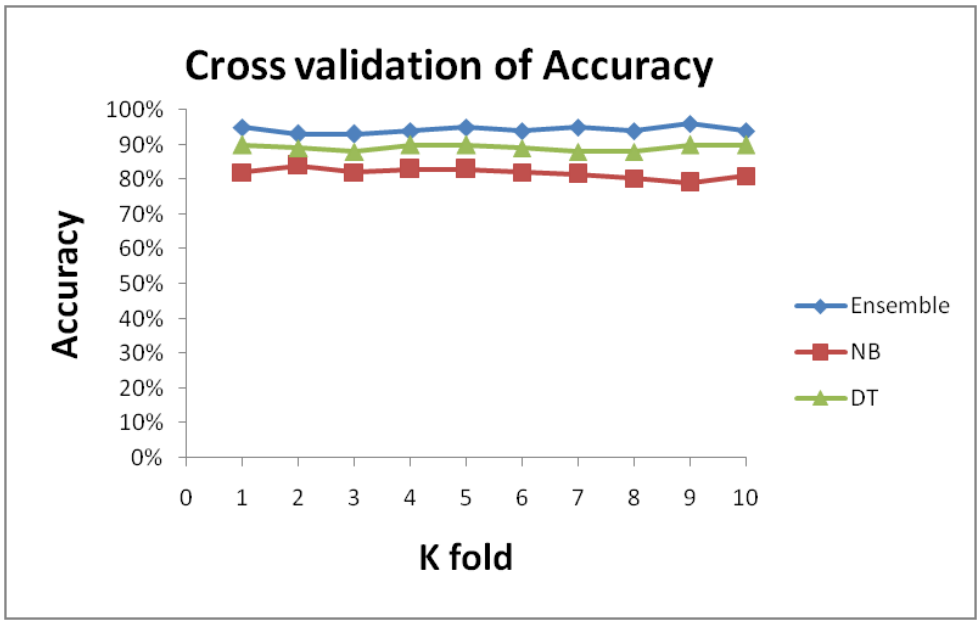

Figure 5.17: K Fold Cross Validation results of accuracy of built ensemble and base classifiers

5.3.1 K-Fold Cross Validation

i. Accuracy of built ensemble using Naive Bayes (NB) and Decision Tree (DT) as candidates is graphically depicted on each fold of 10-cross validation method in the Figure 5.17.
ii. It can be observed that the accuracy of ensemble is better than candidate classifier's accuracy and it is quite robust as value of accuracy is not changing abruptly.

iii. Sensitivity of built ensemble using Naive Bayes (NB) and Decision Tree (DT) as base classifiers is graphically depicted on each fold of 10-cross validation method in the Figure 5.18.

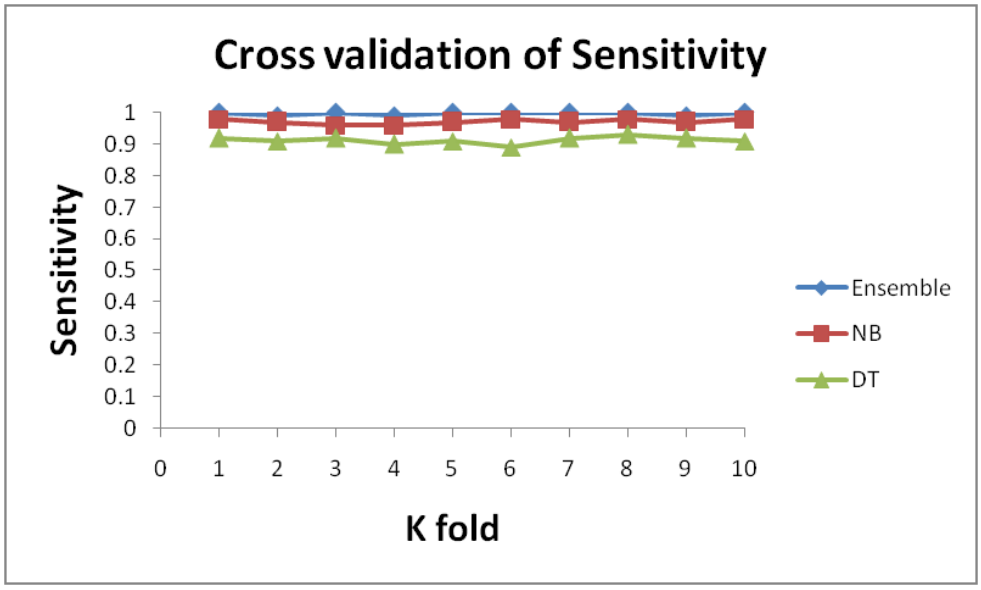

Figure 5.18: K Fold Cross Validation results of sensitivity of built ensemble and base classifiers

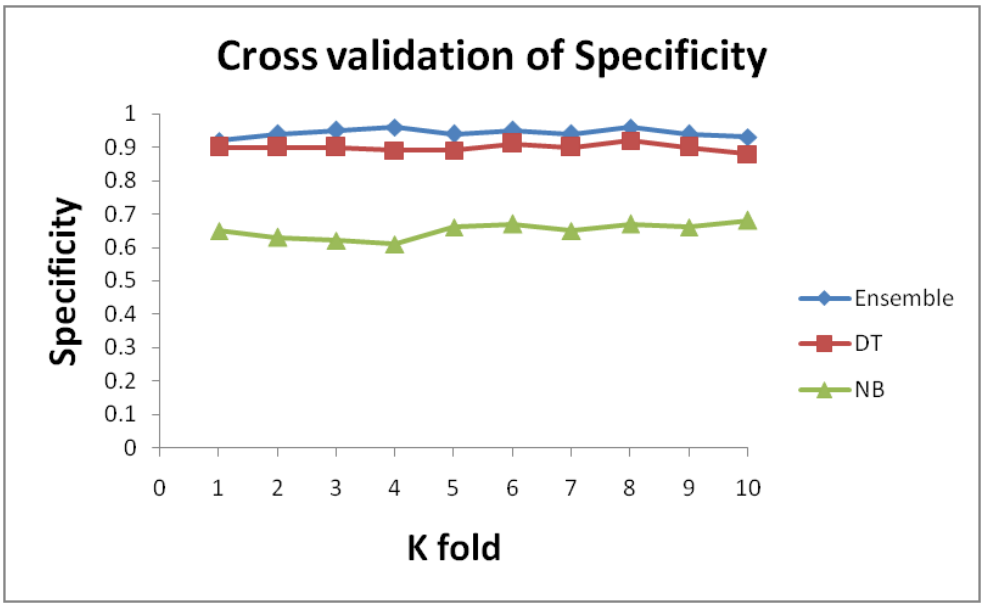

Figure 5.19: K Fold Cross Validation results of specificity of built ensemble and base classifiers

iv. It can be observed that the sensitivity of the ensemble is better than the base classifiers and it is quite robust as values are stable and are not changing abruptly.

v. Specificity of built ensemble using Naive Bayes (NB) and Decision Tree (DT) as base classifiers is graphically depicted on each fold of 10-cross validation method in the Figure 5.19.

vi. It can be observed that sensitivity of ensemble is better than candidate classifier's sensitivity and it is quite robust as values are stable and are not changing abruptly.

vii. Area under the curve value (AUC) of built ensemble using Naive Bayes (NB) and Decision Tree (DT) as base classifiers is graphically depicted on each fold of 10-cross validation method in the Figure 5.20.

viii. It can be observed that AUC of the ensemble is better than candidate classifiers and it is quite robust as values are stable and are not changing abruptly.

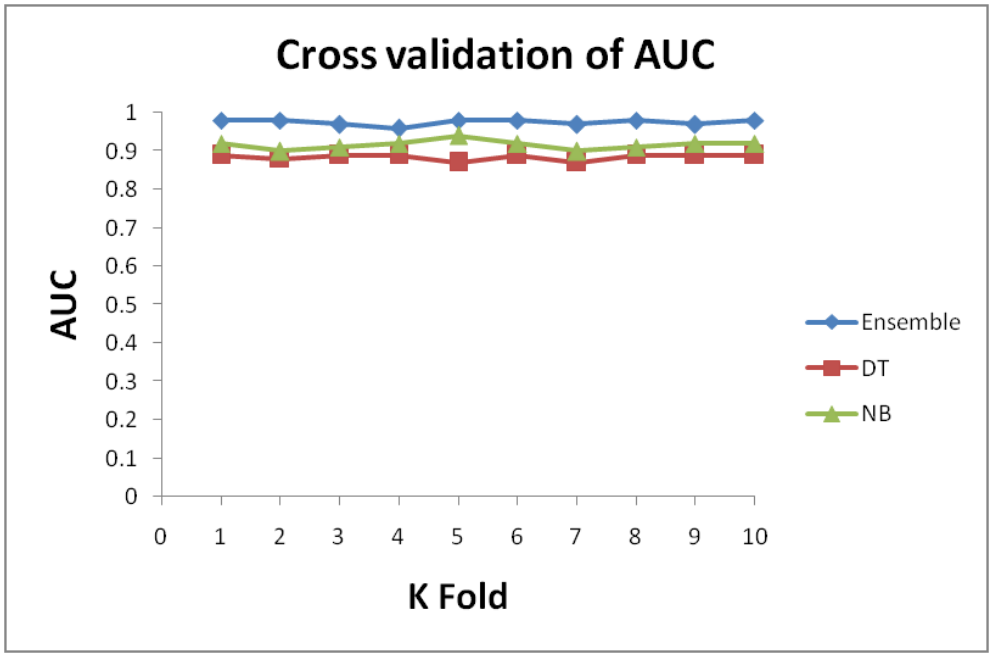

Figure 5.20: K Fold Cross Validation results of AUC of built ensemble and base classifiers

5.3.2 Comparison with State-of-the-Art Methods

For comparison analysis, state-of-the-art classifiers are explored in auditing and finance domain. Results of MCTOPE framework is compared with the outstanding classical classifiers like Support vector machine (SVM), randomforest, neural network, C4.5, adaboost, naive

bayes, etc in the Table 5.5. In machine learning, support vector machine is very popular method of learning from examples [260]. It performs classification using kernel method by implicitly mapping the inputs into high dimensional feature space [14]. C4.5 algorithm is used to build the decision tree from the training data [259]. It works by choosing the effective splits at every node by selecting the best attribute in data. The idea of splitting is to choose the attribute with maximum information gain. Adaboost is used to improve the predictive performance of the machine learning system [30]. Bagging, boosting, random forest, decision trees are successfully used by researchers in biological sciences [88, 207].

As focusing only on the accuracy is not a great deal when using a highly imbalanced data [39, 65, 129], results in the Table 5.5 show that the performance of the proposed framework outperforms the other standard frameworks in terms of sensitivity, specificity, F measure, and AUC. The MCC value of the proposed framework is lesser than random forest and J48 classifiers. MCC takes into account the true positives, false positives and false negatives and describe these measures by a single number. It helps us to know that Random forest and J48 can also be considered for training the final prediction system.For this purpose AUC value of all three classifiers are compared. The results prove that there is a significant rise of AUC of the proposed ensemble when it is compared with the other classical methods. It can also be observed that the value of AUC in the proposed method is quite closer to the best value which is one. It signifies the framework worked perfectly in bringing down the biasing suffered by the data. The prediction method outperforms other techniques hence chosen as the prediction model for the drug toxicity decision support system. In an attempt to perform comprehensive performance evaluation, TOPSIS score is compared in the Figure 5.21. It can be easily observed that the performance of the ensemble from decision tree and naive bayes is performing much better than other classifiers.

5.3.3 Testing Web Application

In order to predict a new firm for next year fraud risk, a web-application is developed that takes input of the important features and predict the probability of risk using ensemble model, working in the back-end.

Table 5.5: Experimental Results

Metric	Ensemble	SVM	Rf	J48	NN	Aboost	NB	BN	DT	PLM	DS
Accuracy	**94**	65.63	93	92	79.15	92.65	82	91	90	92	87.68
Sensitivity	**1.00**	0.50	0.95	0.97	0.89	0.96	0.99	0.95	0.92	0.95	0.98
Specificity	**0.92**	0.81	0.91	0.89	0.63	0.89	0.65	0.87	0.90	0.88	0.75
F measure	**0.94**	0.64	0.93	0.93	0.79	0.92	0.81	0.91	0.91	0.92	0.87
MCC	**0.83**	0.32	*0.87*	0.87	0.59	0.85	0.69	0.84	0.83	0.84	0.77
AUC	**0.98**	0.65	0.96	0.93	0.86	0.93	0.92	0.95	0.89	0.93	0.86

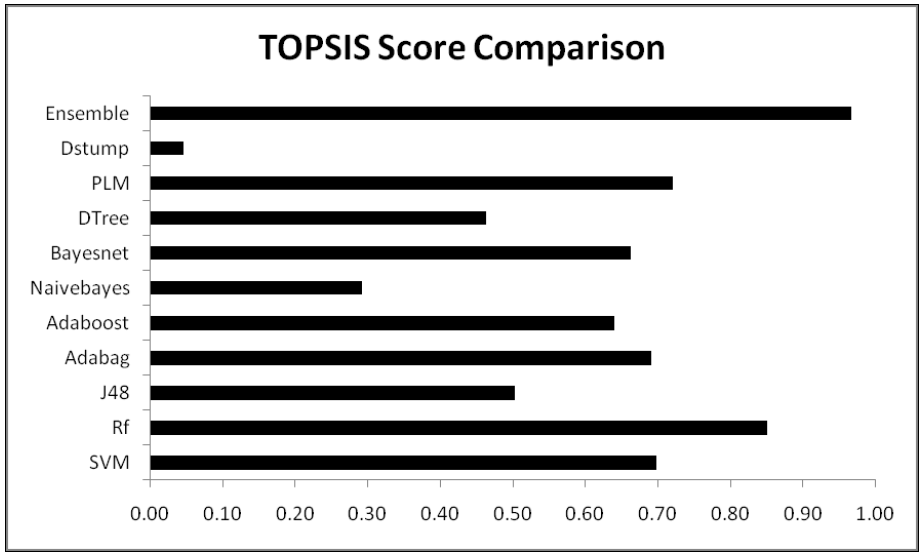

Figure 5.21: TOPSIS Performance Comparison Analysis

5.3.4 Test Case Execution

The test cases are developed to check the fraudulent firm predictor for the firm. Numerous test-cases are executed and which get failed are worked upon again. Passed testcases are presented in the Table 5.6. Test case 1 is executed to check the positive class. So, the data of firm with high probability of risk is entered in the web application. In the Figure 5.23, "There is high probability of fraud detected for this firm" output is obtained which is also

the expected output. The test is passed.

For testing the negative class, test-case 2 is executed. It is shown in the Figure 5.24. So, the data of firm with less probability of risk is entered in the web-application. “There is low probability of fraud detected for this firm” output is obtained which is also the expected output. Hence, the test is passed.

To test the field validation of first feature, out of range values are entered in the web application. The output “Value is out of range” is obtained, which is the expected output. So, the test is passed. Figure 5.25 shows the positive field testing. The right range of values are entered and no error message is displayed. For negative field validation of feature, alphabet is entered for negative testing. Output showing “Please enter a number” is the expected output. Hence, the test is passed again. Again, negative value less than zero is entered in the feature input. The output showing “Please select a value that is no less than 0” shows the test is passed again. Similarly, tests are executed for other features. For button testing, ‘contact us,’ button is clicked and the links moves the user to next page showing the contacts. Similarly, ’About us,’ button is tested and link moves the user to next page showing the information of the product.

Table 5.6: Test Cases

Test ID	Test Objective	Test Procedure	Expected Result	Actual Result	Status
1	To test the firm having probability of risk.	Enter the sample of firm with high risk of fraud. Click submit.	Output: "There is high probability of fraud detected for this firm"	Output: "There is high probability of fraud detected for this firm"	PASS
2	To test the firm having no probability of risk.	Enter the sample of firm with low risk of fraud. Click submit.	Output: "There is less probability of fraud detected for this firm"	Output: "There is less probability of fraud detected for this firm"	PASS
3	To test the field validation of first feature.	Enter out of range value.	Output: "Value is out of range"	Output: "Value is out of range"	PASS
4	To test the field validation of second feature.	Enter alphabets.	Output: "Please enter a number"	Output: "Please enter a number"	PASS
5	To test the field validation of third feature.	Enter negative value.	Output: "Please select a Value that is no less than 0"	Output: "Please select a Value that is no less than 0"	PASS
6	To test the field validation of feature.	Enter no input.	Output: "Please enter a number"	Output: "Please enter a number"	PASS
7	To test the field validation of fourth feature.	Enter negative value.	Output: "Please select a Value that is no less than 0"	Output: "Please select a Value that is no less than 0"	PASS
8	To test the field validation of fourth feature.	Enter alphabets.	Output: "Please enter a number"	Output: "Please enter a number"	PASS
9	To test the field validation of fourth feature.	Enter no input.	Output: "Please enter a number"	Output: "Please enter a number"	PASS
10	To test submit button	Click on 'submit' button.	Move on next page showing prediction results.	Move on next page showing prediction results.	PASS
11	To test 'contact us' button	Click on 'contact us' button.	Move on next page showing the contacts.	Move on next page showing the contacts.	PASS
12	To test 'About us' button	Click on 'About us' button.	Move on next page showing the product information.	Move on next page showing the product information.	PASS

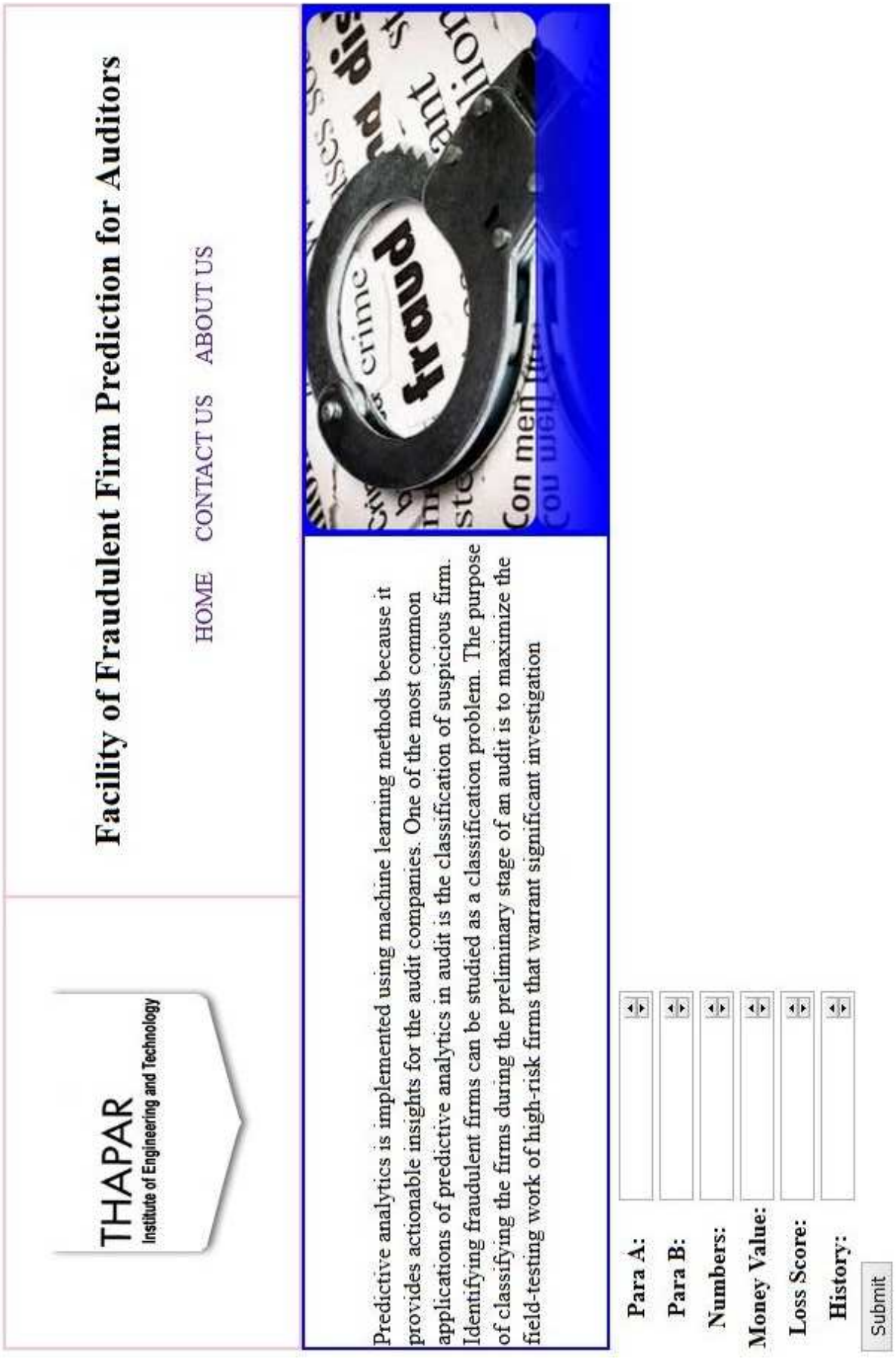

Figure 5.22: Home page of Fraudulent Firm Application

Figure 5.23: Fraud class testing of Fraudulent Firm Application

Figure 5.24: No fraud class testing of Fraudulent Firm Application

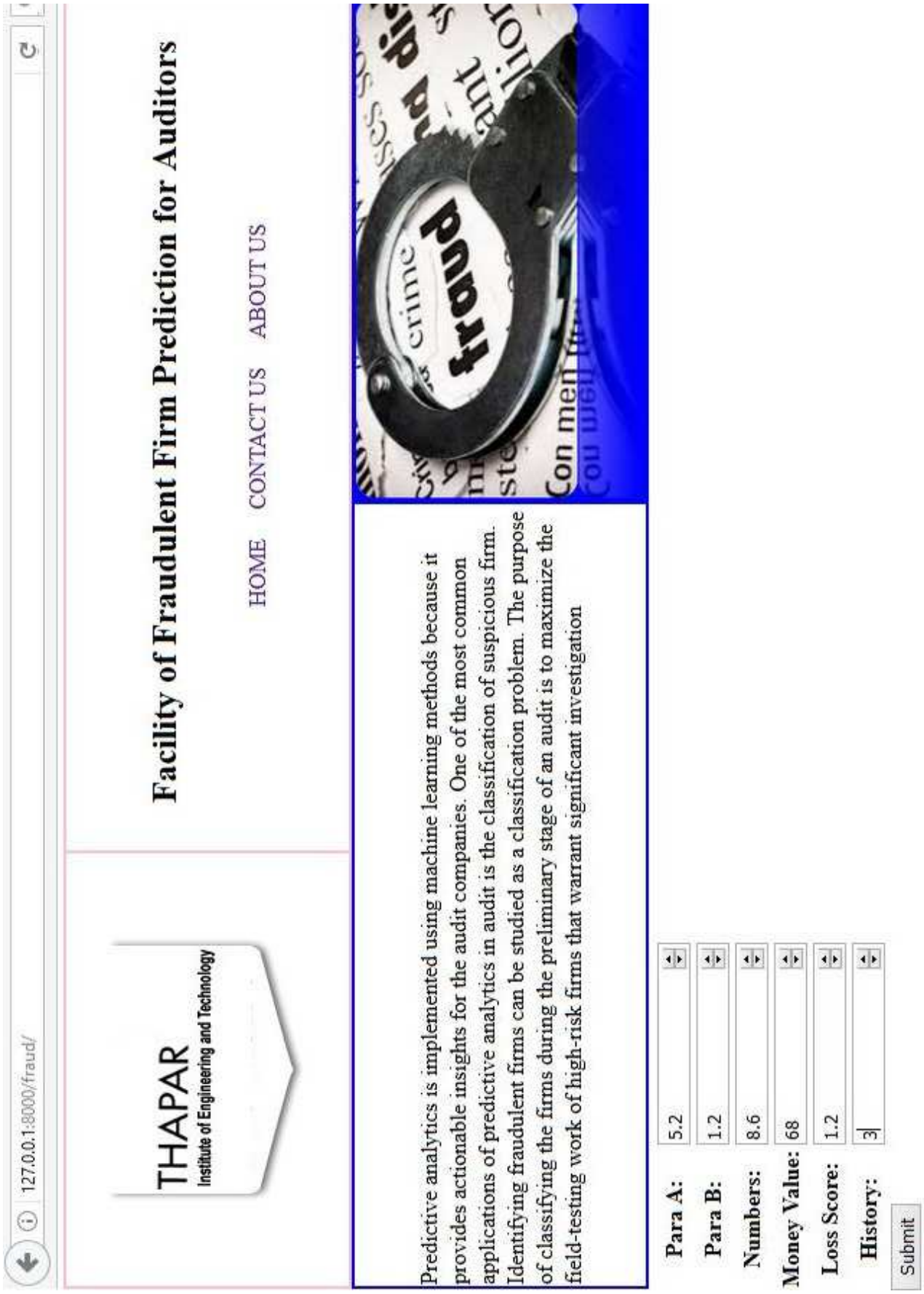

Figure 5.25: Positive field validation testing of Fraudulent Firm Application

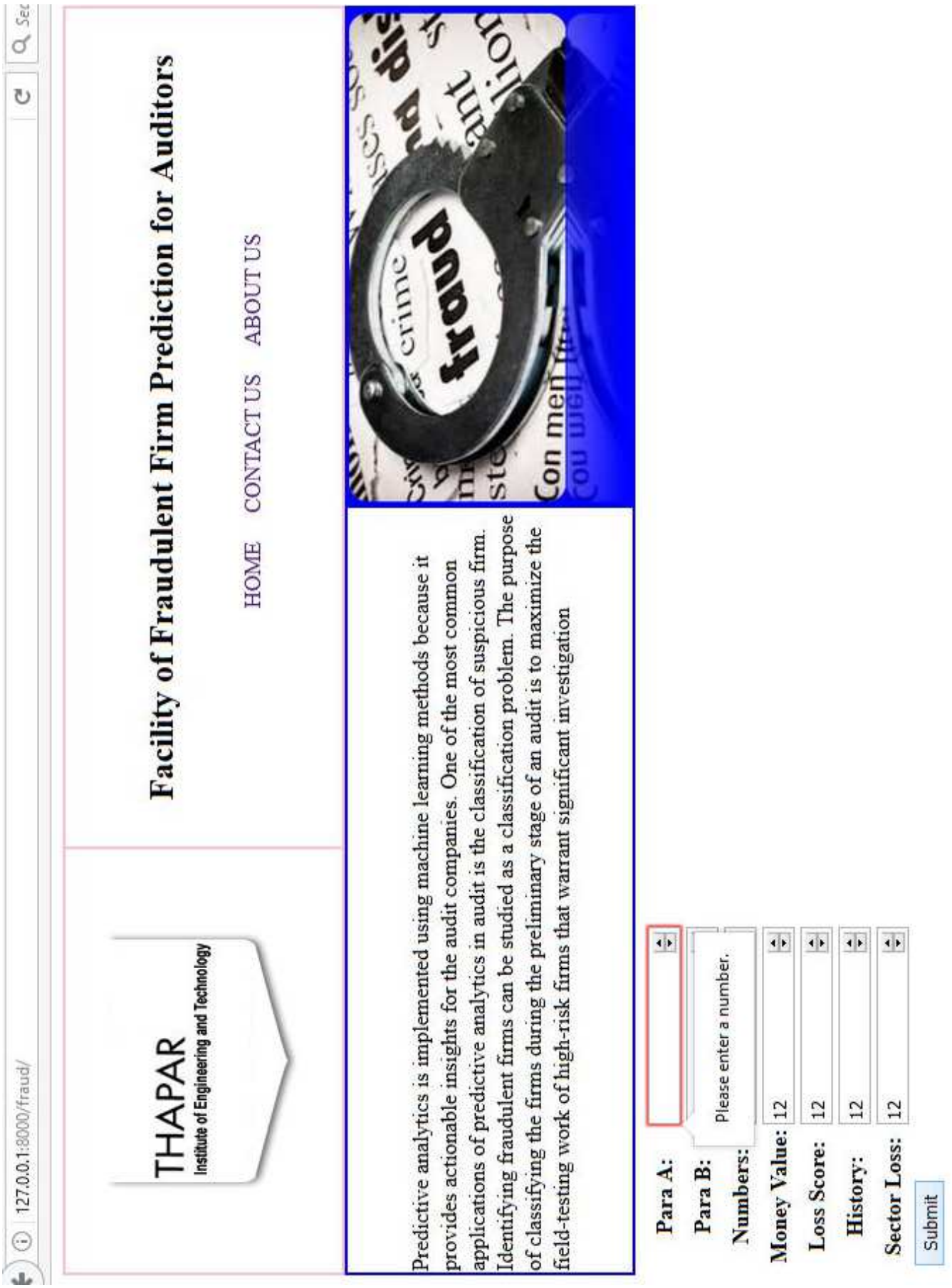

Figure 5.26: Field validation testing of Fraudulent Firm Application

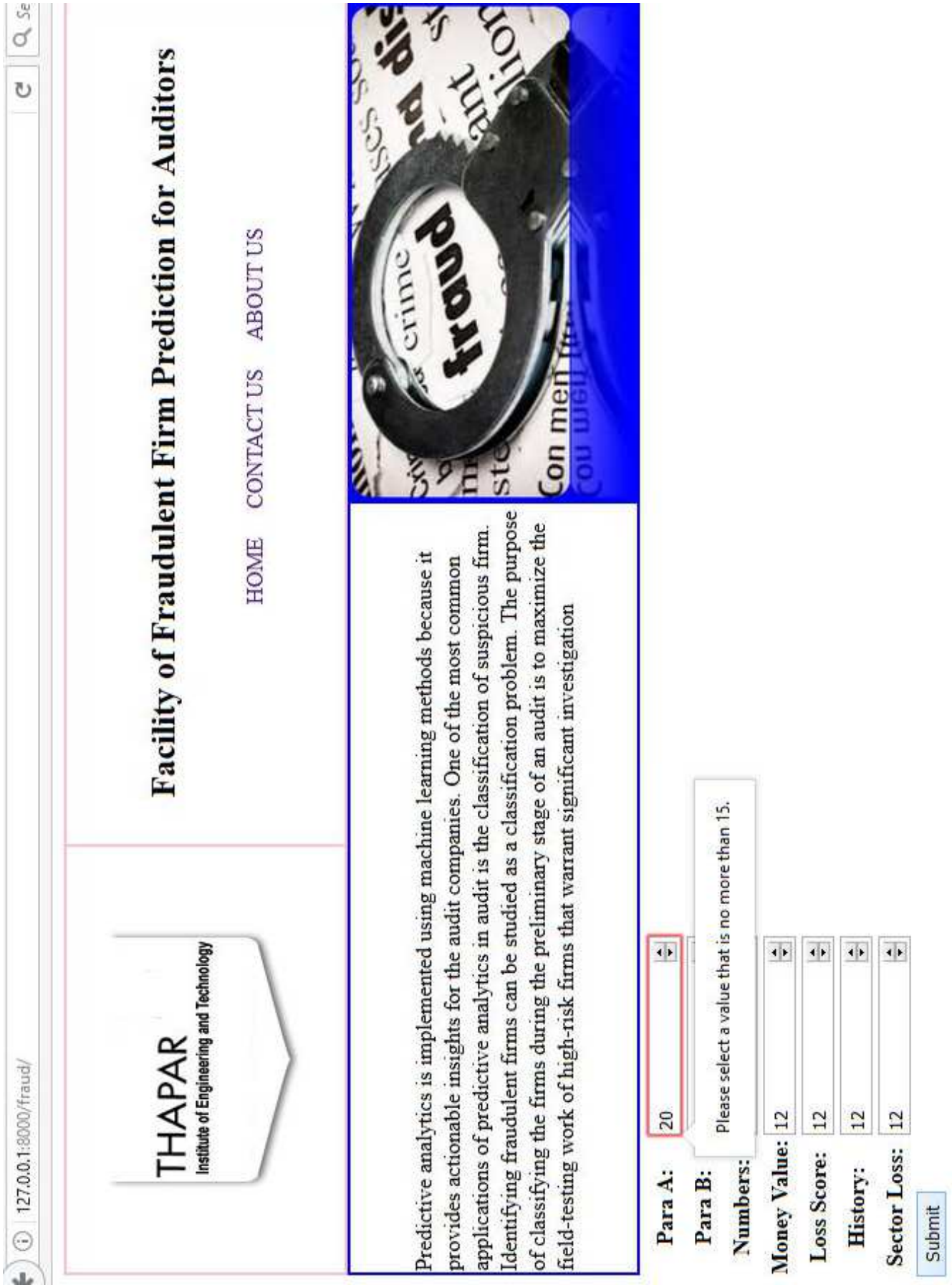

Figure 5.27: Field validation testing of Fraudulent Firm Application

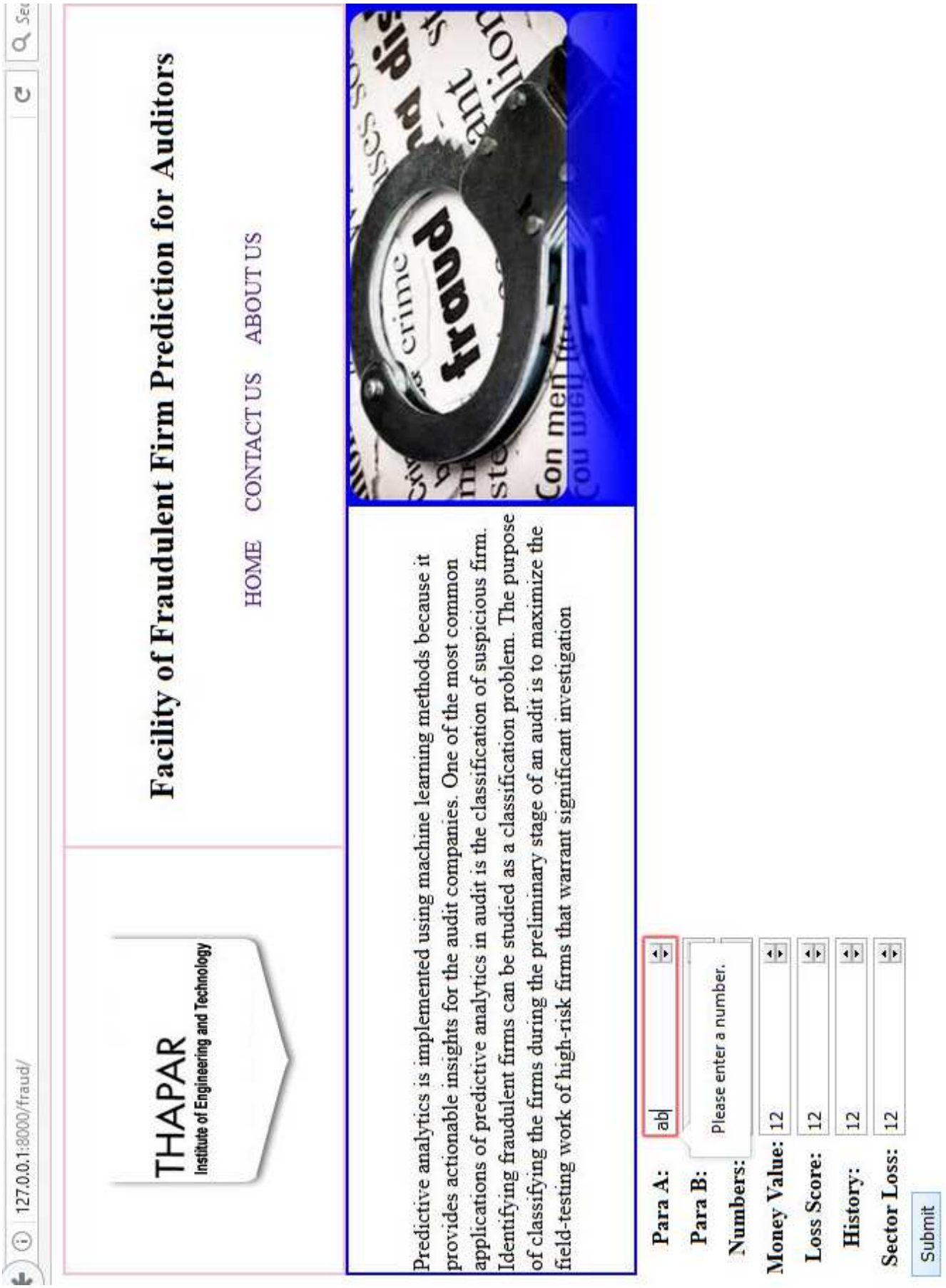

Figure 5.28: Field validation testing of Fraudulent Firm Application

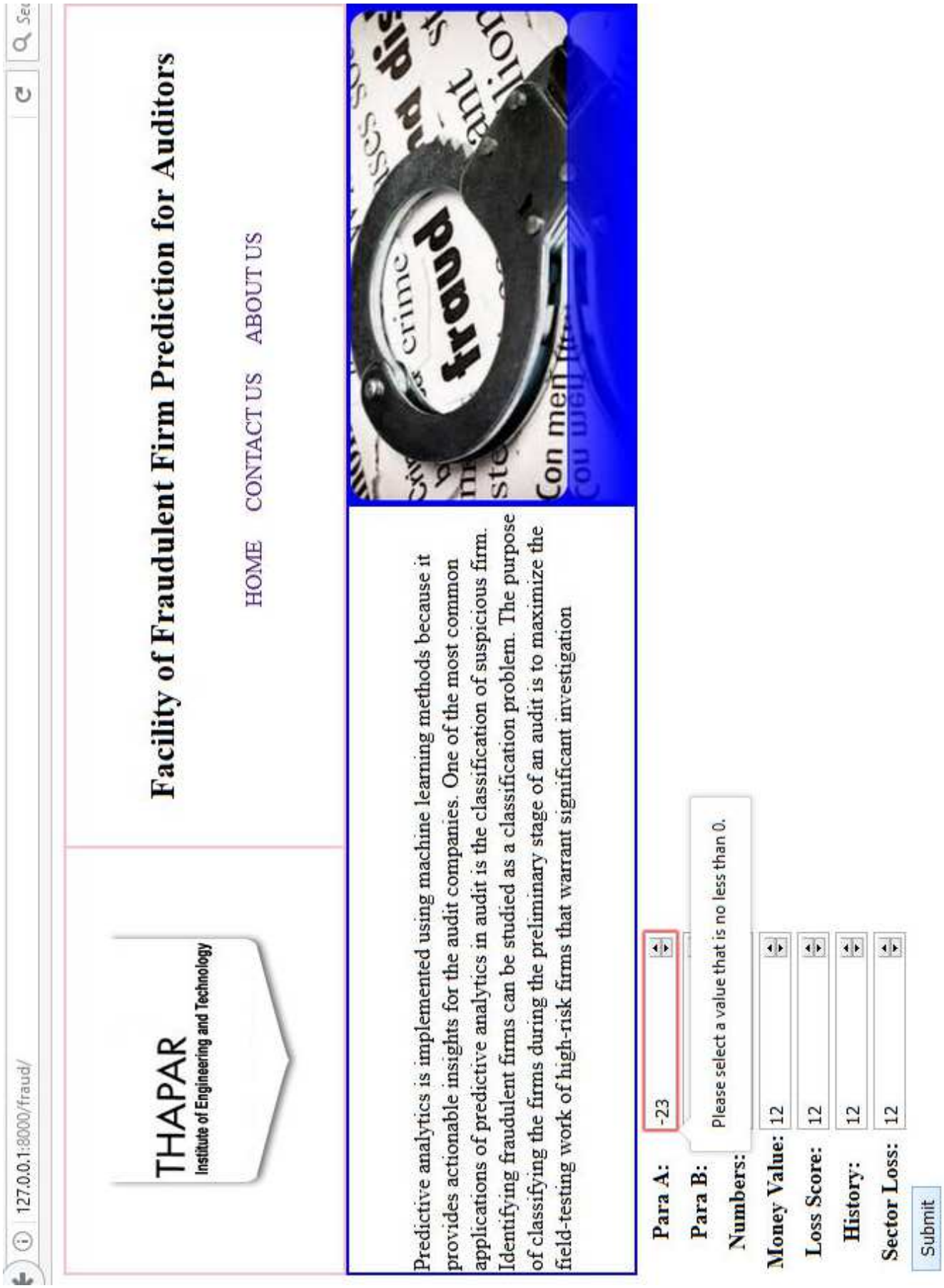

Figure 5.29: Field validation testing of Fraudulent Firm Application

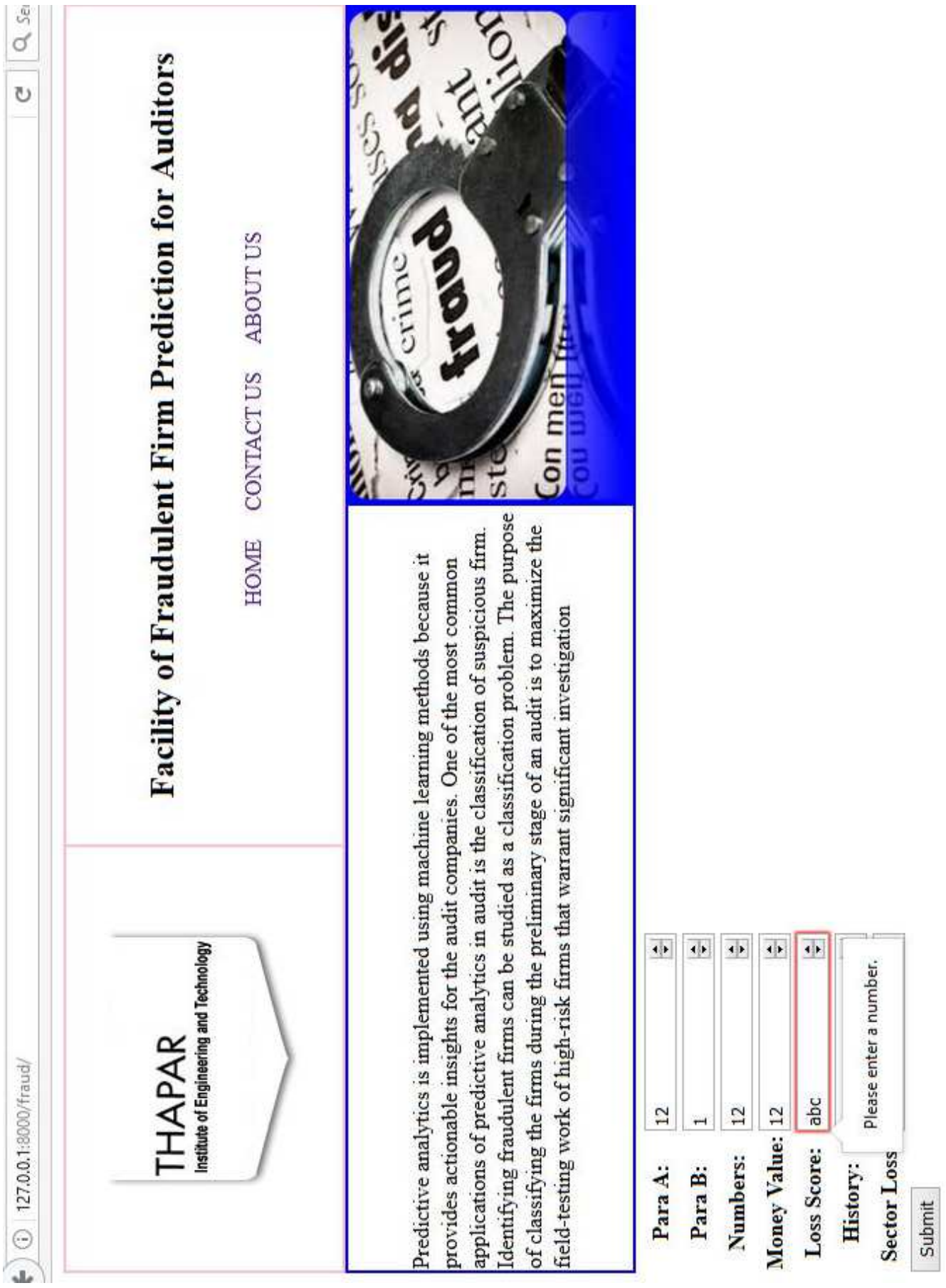

Figure 5.30: Field validation testing of Fraudulent Firm Application

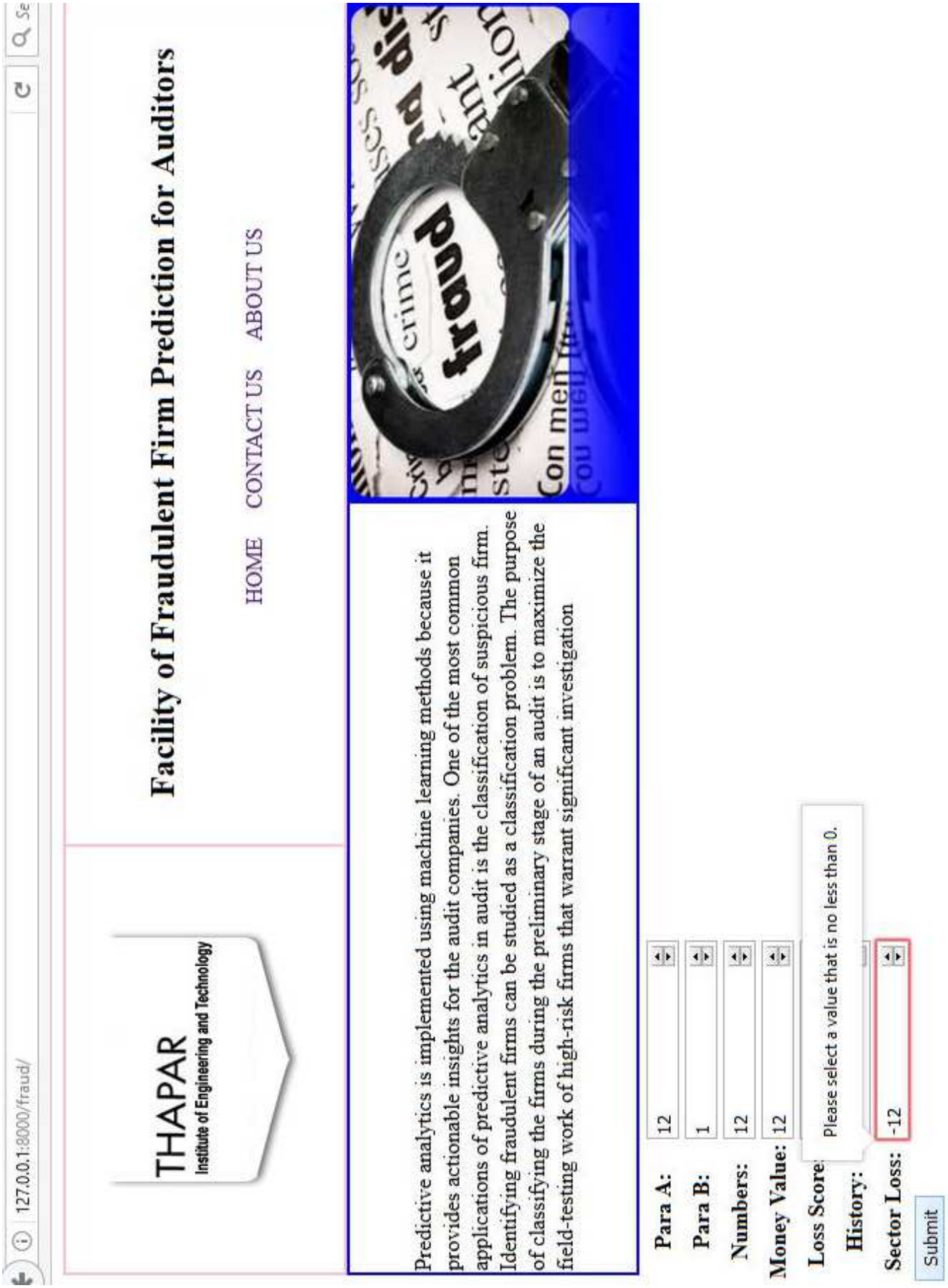

Figure 5.31: Field validation testing of Fraudulent Firm Application

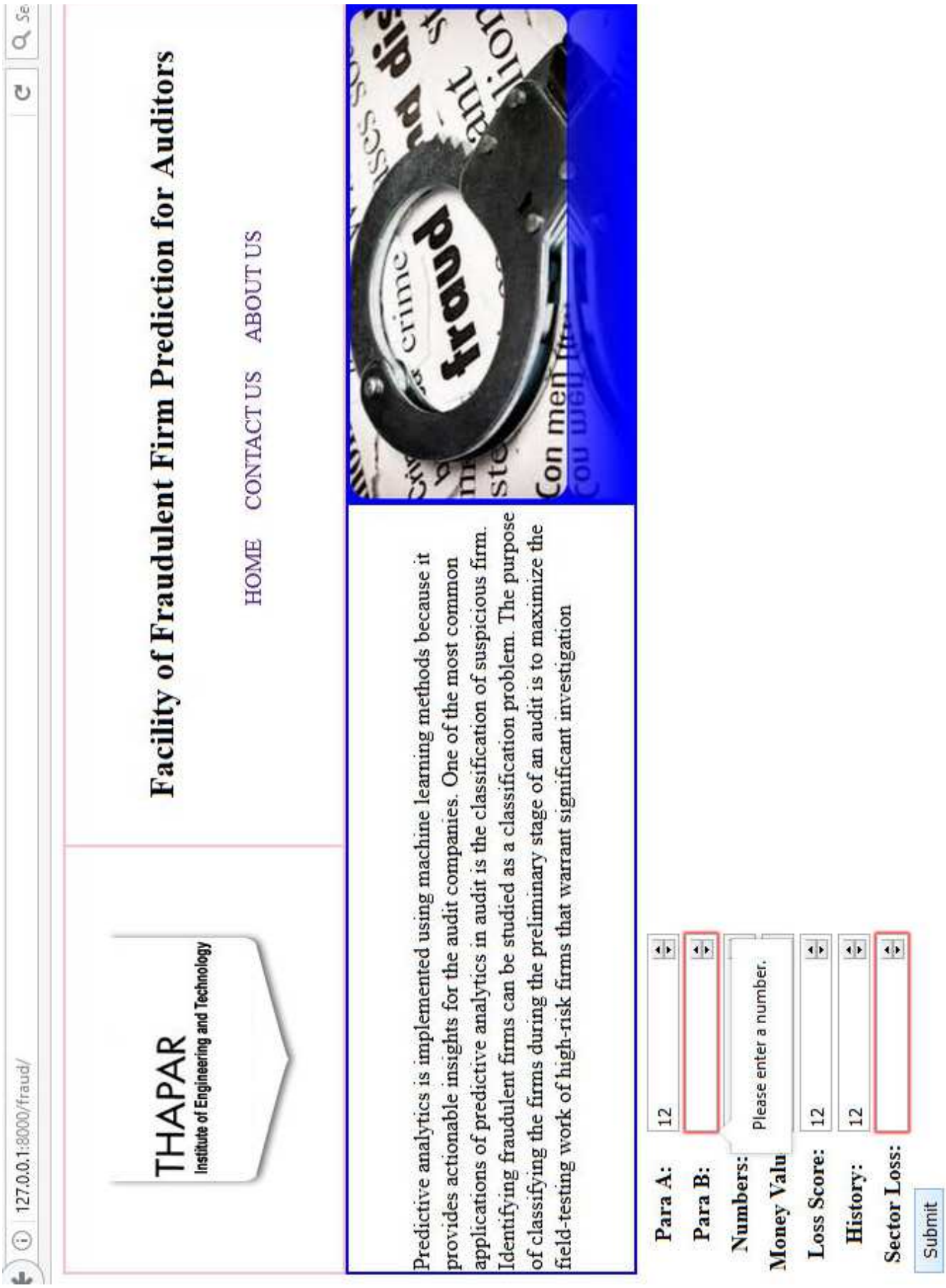

Figure 5.32: Field validation testing of Fraudulent Firm Application

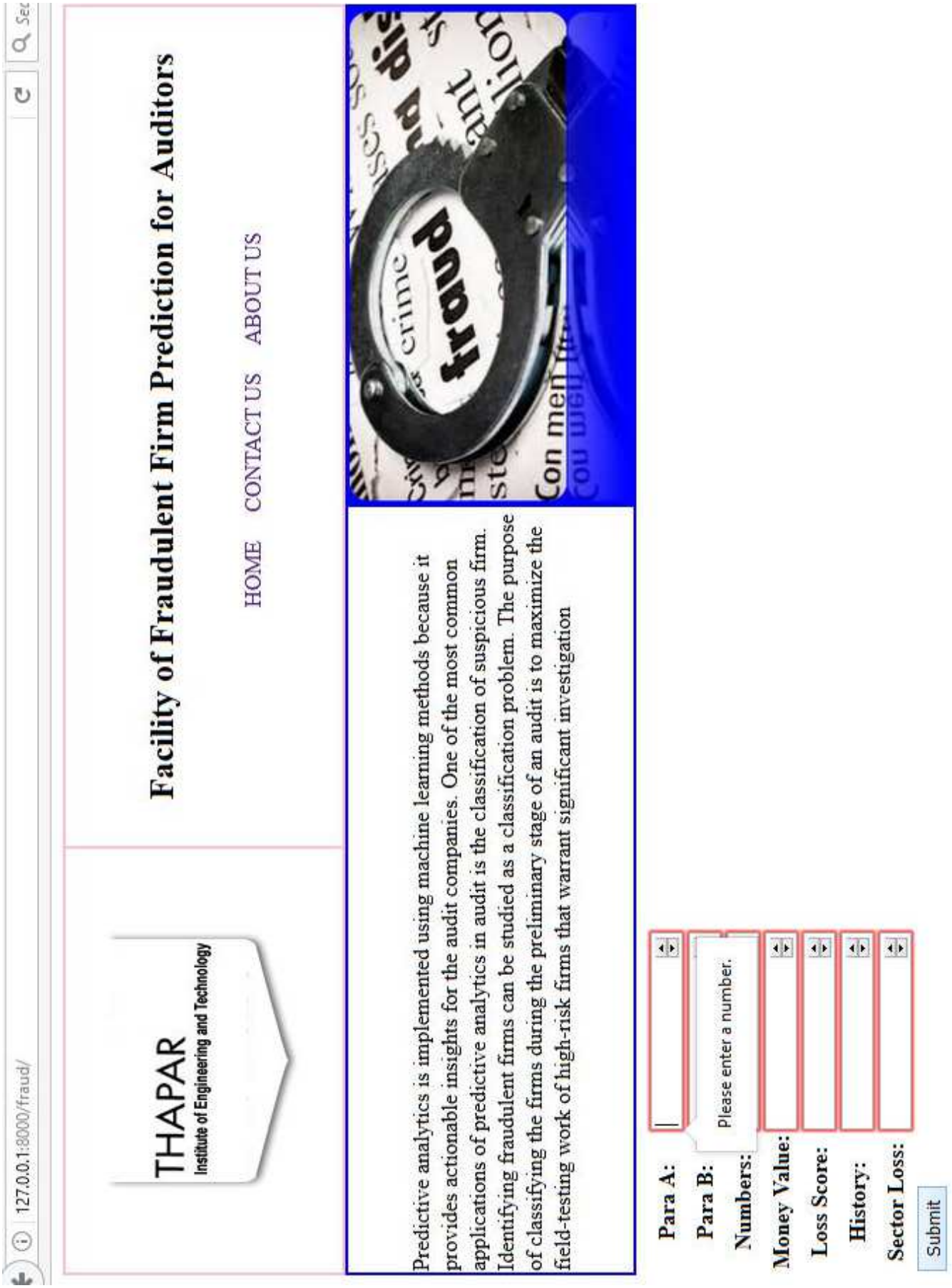

Figure 5.33: Field validation testing of Fraudulent Firm Application

Chapter 6

Conclusions and Future scope

This Chapter concludes the thesis and discusses the scope for future work.

6.1 Conclusion

In this research, a framework called Multi Criteria based TOPsis Ensemble (MCTOPE) is proposed to solve the prediction problems. The research is motivated by the increasing role of ensemble machine learning algorithms in the predictive analytics. In the research work carried out so far in the domain of machine learning and data analytics, an ensemble machine learning approach is used for improving the accuracy of the prediction results. However, slightly different hybrid architecture integrating multi-criteria decision making for ensemble building is presented here. The classical methods test accuracy improvements in the prediction results after ensemble building, which is not always guaranteed. The proposed architecture is different because it considers different evaluation metrics like accuracy, sensitivity, specificity, AUC, etc. during ensemble building process using multi-criteria analysis algorithm and works on optimizing the results for more than thousand iterations to find the best candidates for the ensemble model.

The proposed method is first tested on simple prediction problems to evaluate it's performance and then tested on the two case-studies. K-fold cross validation technique is adopted to test the performance of the classifiers as well to test the built ensemble. To test the robustness of the built ensemble, the process is iterated several times. TOPSIS performance score,

a comprehensive performance evaluation approach is employed for comparing the proposed ensemble method with the state-of-the art methods of different domain.

Drug toxicity prediction (Case Study1): The proposed method, despite having highly imbalanced data in the case study, produced an accuracy of 95% on the testing data. It is also found to be superior in the overall performance, when an extensive comparison is made with the standard SVM and other state-of-the-art methods using different goodness-of-fit classification metrics. The proposed framework can be used as the decision support system to predict the toxicity of an unknown drug molecule. To validate it analytically, the molecular descriptors of three unknown drug molecules, namely nevirapine, delavirdine, and efavirenz, which play a key role in the AIDS therapy are used as testing drug molecules. The 100% correct prediction results serve as a proof of eligibility of the proposed framework to perform an efficient toxicity assessment task.

Fraudulent Firm Prediction (Case Study 2): Fraud is a critical issue worldwide. Firms that resort to the unfair practices without the fear of legal repercussion have a grievous consequences for the economy and individuals in the society. Auditing practices are responsible for the fraud detection. With the appearance of tremendous growth of financial fraud cases and corruption in the Indian society, auditors have become overburdened with ever-increasing amount of data. Machine learning is a type of artificial intelligence that offers a variety of algorithms, enabling the computer to recognize the features of suspicious firms through the learning from previous audit experiences. Audit data analysis using machine learning is a kind of business intelligence that can help the auditors in improving the quality of an audit field work decision making process. A web-application is built that can predict the fraudulent/non-fraudulent class of a unknown firm. This case study is an important contribution that highlights the applicability of machine learning algorithms for improving the quality of an audit work.

6.2 Future Work

Research is iterative and continuous procedure. The work presented in the thesis focuses on solving the prediction problems using multi criteria evaluation based ensemble building technique. There are several directions in which this research work could be expanded. Some of the suggestions for the future work are as follows:

i. In the thesis, ten machine learning models are used for testing the candidates for ensemble building. Several other machine learning models are available and can be explored for improving the performance of prediction.

ii. The computer program of MCTOPE framework is not commercially available right now. For future works, we are working on it to make it available as an open source program.

iii. For processing the high volume of data, we are suggesting to enhance the MCTOPE framework by implementing it on the top of modern big data techniques like Hadoop, Spark, etc.

References

[1] Big Data in Action. www-01.ibm.com/software/data/bigdata.html, 2015. [Accessed: 25-June-2017].

[2] The Awesome Ways Big Data is Used Today to Change Our World. www.datasciencecentral.com/change-our-world, 2015. [Accessed: 25-June-2017].

[3] Insights into what the world is searching for – the new Google Trends. https://search.googleblog.com/2012/09/insights-into-what-world-is-searching.html, 2017. [Accessed: 25-June-2017].

[4] C. Aggarwal. An introduction to social network data analytics. *Social network data analytics*, pages 1–15, 2011.

[5] MOA Massive Online Analysis. http://moa.cms.waikato.ac.nz/. [Accessed:25-June-2017].

[6] Angoss. http://www.angoss.com/. [Accessed:25-June-2017].

[7] K. E Arnold. Signals: Applying academic analytics. *Educause Quarterly*, 33(1):n1, 2010.

[8] P. Bruza et al. B. Koopman. Evaluating medical information retrieval. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 1139–1140. ACM, 2011.

[9] CL Blake and Christopher J Merz. Uci repository of machine learning databases [http://www. ics. uci. edu/˜ mlearn/mlrepository. html]. irvine, ca: University of california. *Department of Information and computer science*, 55, 1998.

[10] Thomas O Boucher and Elin L MacStravic. Multiattribute evaluation within a present value framework and its relation to the analytic hierarchy process. *The Engineering Economist*, 37(1):1–32, 1991.

[11] Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

[12] L. Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[13] Wray Buntine. Learning classification rules using bayes. In *Proceedings of the sixth international workshop on Machine learning*, pages 94–98, 2016.

[14] V. Vladimir C. Cortes. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[15] J. Caumaros. Big Data Alchemy: How can Banks Maximize the Value of their Customer Data? www.capgemini-consulting.com/bigdatainbanking.pdf, 2013. [Accessed: 25-Mar-2018].

[16] Chui-Hui et al. Chiu. An effective distributed ghsom algorithm for unsupervised clustering on big data. In *Big Data (BigData Congress), 2017 IEEE International Congress on*, pages 297–304. IEEE, 2017.

[17] G. Cosserat. *Modern auditing*. Wiley, 2009.

[18] Oracle data mining. http://www.oracle.com/technetwork/database/options/advanced-analytics/odm/overview/index.html. [Accessed:25-June-2017].

[19] Australian dataset. https://archive.ics.uci.edu/ml/datasets/statlog. [Accessed:25-June-2017].

[20] German dataset. https://archive.ics.uci.edu/statlog/german/. [Accessed:25-June-2017].

[21] Pop Failure Dataset. http://networkrepository.com/pop-failures.php). [Accessed:25-June-2017].

[22] Sonar dataset. https://archive.ics.uci.edu/sonar/sonar.all-data. [Accessed:25-June-2017].

[23] Wholesale dataset. https://archive.ics.uci.edu/datasets/wholesale. [Accessed:25-June-2017].

[24] Thomas H. Davenport. At the Big Data Crossroads: turning towards a smarter travel experience. www.bigdata.amadeus.com/Amadeus-Big-Data.pdf, 2013. [Accessed: 25-Mar-2018].

[25] PaDEL Descriptor. Software. http://www.yapcwsoft.com/dd/padeldescriptor/(Accessed: 2017-04-10).

[26] F. Diebold. A personal perspective on the origin (s) and development of'big data': The phenomenon, the term, and the discipline, second version. 2012.

[27] Thomas G Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine learning*, 40(2):139–157, 2000.

[28] ELKI. https://elki-project.github.io/. [Accessed:25-June-2017].

[29] A. Bechini et al. A mapreduce solution for associative classification of big data. *Information Sciences*, 332:33–55, 2016.

[30] A. Chandra et al. Evolving hybrid ensembles of learning machines for better generalisation. *Neurocomputing*, 69(7):686–700, 2006.

[31] A. Fahad et al. A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE transactions on emerging topics in computing*, 2(3):267–279, 2014.

[32] A. Fernández et al. Fuzzy rule based classification systems for big data with mapreduce: granularity analysis. *Advances in Data Analysis and Classification*, 11(4):711–730, 2017.

[33] A. Fernández et al. An insight into imbalanced big data classification: outcomes and challenges. *Complex & Intelligent Systems*, 3(2):105–120, 2017.

[34] A. Gandomi et al. Beyond the Hype: Big Data Concepts, Methods, and Analytics. *International Journal of Information Management*, 35(2):137–144, 2015.

[35] A. Haque et al. Evolving Big Data Stream Classification with Map-Reduce. In *IEEE 7th International Conference on Cloud Computing*, pages 570–577, 2014.

[36] A. Lazarevic et al. Feature bagging for outlier detection. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 157–166. ACM, 2005.

[37] A. Lemmens et al. Bagging and boosting classification trees to predict churn. *Journal of Marketing Research*, 43(2):276–286, 2006.

[38] A. McAfee et al. Big Data: The Management Revolution. www.hbr.org/revolution, 2012. [Accessed: 25-Mar-2018].

[39] A. Ramezankhani et al. The impact of oversampling with smote on the performance of 3 classifiers in prediction of type 2 diabetes. *Medical Decision Making*, page 0272989X14560647, 2014.

[40] A. Sehgal et al. Enhancement in k-mean clustering in big data. *International Journal Of Scientific Research And Education*, 5(06), 2017.

[41] A. Toosi et al. Resource provisioning policies to increase iaas provider's profit in a federated cloud environment. In *High Performance Computing and Communications (HPCC), 2011 IEEE 13th International Conference on*, pages 279–287. IEEE, 2011.

[42] A. Zamil et al. The Application of Semantic-Based Classification on Big Data. In *5th IEEE International Conference Information and Communication Systems*, pages 1–5, 2014.

[43] Andersen et al. Toxicity Testing in the 21st Century: Bringing the Vision to Life. *Toxicological sciences*, 107(2):pp. 324–330, 2009.

[44] Arun et al. Mining association rules between sets of items in large databases. In *Acm sigmod record*, volume 22, pages 207–216. ACM, 1993.

[45] B. Fischer et al. Bagging for path-based clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(11):1411–1415, 2003.

[46] B. Fox et al. Analytics: Real-world use of big data in telecommunications. www-935.ibm.com/Anaytics.pdf, 2013. [Accessed: 25-Mar-2018].

[47] B. Krawczyk et al. Combining Nearest Neighbour Classifiers Based on Small Subsamples for Big Data Analytics. In *2nd International Conference on Cybernetics*, pages 311–316, 2015.

[48] B. Liu et al. Scalable Sentiment Classification for Big Data Analysis Using Naive Bayes Classifier. In *IEEE International Conference on Big Data*, pages 99–104, 2013.

[49] Brown et al. Diversity creation methods: a survey and categorisation. *Information Fusion*, 6(1):5–20, 2005.

[50] C. L. Chen et al. Data Intensive Applications, Challenges, Techniques and Technologies: A Survey on Big Data. *Information Sciences*, 275:314–347, 2014.

[51] C. Shi et al. Machine learning under big data. 2016.

[52] C. Shipp et al. Relationships between combination methods and measures of diversity in combining classifiers. *Information fusion*, 3(2):135–148, 2002.

[53] C. Tew et al. Behavior-based clustering and analysis of interestingness measures for association rule mining. *Data Mining and Knowledge Discovery*, 28(4):1004–1045, 2014.

[54] C. Yang et al. Big data and cloud computing: innovation opportunities and challenges. *International Journal of Digital Earth*, 10(1):13–53, 2017.

[55] Chih-Fong Tsai et al. Using neural network ensembles for bankruptcy prediction and credit scoring. *Expert systems with applications*, 34(4):2639–2649, 2008.

[56] CR Mi et al. Climate change would enlarge suitable planting areas of sugarcanes in china. *International Journal of Plant Production*, 11(1), 2017.

[57] Cuzzocrea et al. Cloud-based Machine Learning Tools for Enhanced Big Data Applications. In *15th International Symposium on Cluster, Cloud and Grid Computing*, pages 908–914, 2015.

[58] D. Benbouzid et al. Multiboost: a multi-purpose boosting package. *Journal of Machine Learning Research*, 13(Mar):549–553, 2012.

[59] D. Feldman et al. Turning big data into tiny data: Constant-size coresets for k-means, pca and projective clustering. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*, pages 1434–1453. SIAM, 2013.

[60] D. Peralta et al. Evolutionary feature selection for big data classification: A mapreduce approach. *Mathematical Problems in Engineering*, 2015, 2015.

[61] D. Ruta et al. Classifier selection for majority voting. *Information fusion*, 6(1):63–81, 2005.

[62] D. Wang et al. High-dimensional Data Stream Classification via Sparse Online Learning. In *IEEE International Conference on Data Mining*, pages 1007–1012, 2014.

[63] D. West et al. Neural network ensemble strategies for financial decision applications. *Computers & operations research*, 32(10):2543–2559, 2005.

[64] D. Wolpert et al. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.

[65] Daddabbo et al. Parallel Selective Sampling method for imbalanced and large data classification. *Pattern Recognition Letters*, 62:pp. 61–67, 2015.

[66] E. Arabmakki et al. RLS-A Reduced Labeled Samples Approach for Streaming Imbalanced Data with Concept Drift. In *IEEE 15th International Conference Information Reuse and Integration*, pages 779–786, 2014.

[67] E. Bauer et al. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning*, 36(1):105–139, 1999.

[68] E. Ngai et al. The Application of Data Mining Techniques in Financial Fraud Detection: A Classification Framework and an Academic Review of Literature. *Decision Support Systems*, 50(3):559–569, 2011.

[69] Eun Bae Kong et al. Error-correcting output coding corrects bias and variance. In *ICML*, pages 313–321, 1995.

[70] F. Boran et al. A multi-criteria intuitionistic fuzzy group decision making for supplier selection with topsis method. *Expert Systems with Applications*, 36(8):11363–11368, 2009.

[71] F. Jianqing et al. Challenges of big data analysis. *National Science Review*, 1(2):293–314, 2014.

[72] F. Junior et al. A comparison between fuzzy ahp and fuzzy topsis methods to supplier selection. *Applied Soft Computing*, 21:194–209, 2014.

[73] F. Kargarfard et al. Knowledge discovery and sequence-based prediction of pandemic influenza using an integrated classification and association rule mining (cba) algorithm. *Journal of biomedical informatics*, 57:181–188, 2015.

[74] F. Roli et al. Methods for designing multiple classifier systems. In *International Workshop on Multiple Classifier Systems*, pages 78–87. Springer, 2001.

[75] F. Sun et al. Efficient and rapid machine learning algorithms for big data and dynamic varying systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(10):2625–2626, 2017.

[76] Fellus et al. Asynchronous Gossip Principal Components Analysis. *Neurocomputing*, 169:262–271, 2015.

[77] G. Cavallaro et al. On Understanding Big Data Impacts in Remotely Sensed Image Classification Using Support Vector Machine Methods. *Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, PP(9):1–13, 2015.

[78] G. Czibula et al. Software defect prediction using relational association rule mining. *Information Sciences*, 264:260–278, 2014.

[79] G. Dietterich et al. Ensemble methods in machine learning. In *Multiple classifier systems*, pages 1–15. Springer, 2000.

[80] G. Huang et al. Trends in Extreme Learning Machines: A Review. *Neural Networks*, 61:32–48, 2015.

[81] G. Manogaran et al. Machine learning based big data processing framework for cancer diagnosis using hidden markov model and gm clustering. *Wireless Personal Communications*, pages 1–18, 2017.

[82] Gu Min et al. Optical storage arrays: a perspective for future big data storage. *Light: Science and Applications*, 3(5):177–179, 2014.

[83] Gu Xiao-Feng et al. An Improving Online Accuracy Updated Ensemble Method in Learning from Evolving Data Streams. In *11th International Computer Conference on Wavelet Active Media Technology and Information Processing), 2014*, pages 430–433, 2014.

[84] H. Erdal et al. Bagging ensemble models for bank profitability: An emprical research on turkish development and investment banks. *Applied Soft Computing*, 49:861–867, 2016.

[85] H. Graf et al. Parallel support vector machines: The cascade svm. In *NIPS*, volume 17, 2004.

[86] H. Mojaddadi et al. Ensemble machine-learning-based geospatial approach for flood risk assessment using multi-sensor remote-sensing data and gis. *Geomatics, Natural Hazards and Risk*, 8(2):1080–1102, 2017.

[87] H. Zhao et al. Constrained cascade generalization of decision trees. *IEEE Transactions on Knowledge and Data Engineering*, 16(6):727–739, 2004.

[88] Hanczar et al. Using the bagging approach for biclustering of gene expression data. *Neurocomputing*, 74(10):1595–1605, 2011.

[89] Hilbert et al. The worlds technological capacity to store, communicate, and compute information. *science*, 332(6025):60–65, 2011.

[90] Hyun-Chul Kim et al. Constructing support vector machine ensemble. *Pattern recognition*, 36(12):2757–2767, 2003.

[91] I. Maqsood et al. An ensemble of neural networks for weather forecasting. *Neural Computing & Applications*, 13(2):112–122, 2004.

[92] I. Usach et al. Non-nucleoside reverse transcriptase inhibitors. *Journal of the International AIDS Society*, 16(1):176–181, 2013.

[93] J. Bennett et al. The netflix prize. In *Proceedings of KDD cup and workshop*, volume 2007, page 35. New York, NY, USA, 2007.

[94] J. Darrell et al. Human neuroimaging as a Big Data science. *Brain Imaging and Behavior*, 8(2):323–331, 2014.

[95] J. Friedman et al. On bias, variance, 0/1loss, and the curse-of-dimensionality. *Data mining and knowledge discovery*, 1(1):55–77, 1997.

[96] J. Inglese et al. Quantitative high-throughput screening: a titration-based approach that efficiently identifies biological activities in large chemical libraries. *Proceedings of the National Academy of Sciences*, 103(31):11473–11478, 2006.

[97] J. Li et al. Challenges of feature selection for big data analytics. *IEEE Intelligent Systems*, 32(2):9–15, 2017.

[98] J. Li et al. Feature selection: A data perspective. *ACM Computing Surveys (CSUR)*, 50(6):94, 2017.

[99] J. Miers et al. Digitizing Energy:Analytics Powered Performance. *www.accenture.com/Analytics-Powered.pdf*, 2013. [Accessed: 25-Mar-2018].

[100] J. Moeyersoms et al. Including High-Cardinality Attributes in Predictive Models: A Case Study in Churn Prediction in the Energy Sector. *Decision Support Systems*, 72:72–81, 2015.

[101] J. Qiu et al. A survey of machine learning for big data processing. *EURASIP Journal on Advances in Signal Processing*, 2016(1):67, 2016.

[102] J. Quinlan et al. Bagging, boosting, and c4. 5. In *AAAI/IAAI, Vol. 1*, pages 725–730, 1996.

[103] J. Ullman et al. Mining Data Streams. In *Mining of Massive Datasets*, pages 81–98. Cambridge University Press, 2014.

[104] J. Zhang et al. Detecting Anomalies from Big Network Traffic Data using an Adaptive Detection Approach. *Information Sciences*, 2014.

[105] Judd et al. *Data analysis: A model comparison approach*. Routledge, 2011.

[106] K. Govindan et al. Sustainable material selection for construction industry–a hybrid multi criteria decision making approach. *Renewable and Sustainable Energy Reviews*, 55:1274–1288, 2016.

[107] K. Govindarajan et al. Continuous Clustering in Big Data Learning Analytics. In *IEEE Fifth International Conference on Technology for Education*, pages 61–64, 2013.

[108] K. Nishchal et al. Comparative analysis of gaussian mixture model, logistic regression and random forest for big data classification using map reduce. In *Industrial and Information Systems (ICIIS), 2016 11th International Conference on*, pages 333–338. IEEE, 2016.

[109] K. Ting et al. Issues in stacked generalization. *J. Artif. Intell. Res.(JAIR)*, 10:271–289, 1999.

[110] Kempler et al. Earth science data analytics: Bridging tools and techniques with the co-analysis of large, heterogeneous datasets. 2016.

[111] Kerdels et al. Analysis of high-dimensional data using local input space histograms. *Neurocomputing*, 2015.

[112] Kim et al. Big data and statistics. *Journal of the Korean Data and Information Science Society*, 24(5):959–974, 2013.

[113] Kim et al. An Ensemble Regularization Method for Feature Selection in Mass Spectral Fingerprints. *Chemometrics and Intelligent Laboratory Systems*, 2015.

[114] Kola et al. Can the Pharmaceutical Industry Reduce Attrition Rates? *Nature Reviews Drug Discovery*, 3(8):pp. 711–716, 2004.

[115] Kung-Jeng et al. A hybrid classifier combining borderline-smote with airs algorithm for estimating brain metastasis from lung cancer: A case study in taiwan. *Computer methods and programs in biomedicine*, 119(2):63–76, 2015.

[116] L. Blum et al. Selection of Relevant Features and Examples in Machine Learning. *Artificial intelligence*, 97(1):245–271, 1997.

[117] L. Gabralla et al. Oil price prediction using ensemble machine learning. In *Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference on*, pages 674–679. IEEE, 2013.

[118] L. Garg et al. Intelligent patient management and resource planning for complex, heterogeneous, and stochastic healthcare systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 42(6):1332–1345, 2012.

[119] L. Hall et al. Distributed learning on very large data sets. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 79–84. Citeseer, 2000.

[120] L. Hongfei et al. Improving Rail Network Velocity: A Machine Learning Approach to Predictive Maintenance. *Transportation Research Emerging Technologies*, 45:17–26, 2014.

[121] L. Kuncheva et al. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181–207, 2003.

[122] L. Mirsadeghi et al. Evaluation of ensemble classifier (ec) machine learning methods for introduction of breast cancer genomic biomarkers. *Multidisciplinary Cancer Investigation*, 1:0–0, 2017.

[123] L. Nanni et al. An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring. *Expert systems with applications*, 36(2):3028–3033, 2009.

[124] L. Naraor et al. Cisco Visual Networking Index Mobile Data Traffic Forecast Update. www.cisco.com/visual-networking/520862.pdf, 2015. [Accessed:25-Mar-2018].

[125] L. Zhou et al. Machine learning on big data: Opportunities and challenges. *Neurocomputing*, 237:350–361, 2017.

[126] Lee et al. Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. *Neurocomputing*, 2015.

[127] LI Kuncheva et al. An experimental study on diversity for bagging and boosting with linear classifiers. *Information fusion*, 3(4):245–258, 2002.

[128] Liaw et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[129] Lusa et al. Joint use of Over-and Under-Sampling Techniques and Cross-Validation for the Development and Assessment of Prediction Models. *BMC bioinformatics*, 16(1):pp. 2–5, 2015.

[130] M. Behzadian et al. A state-of the-art survey of topsis applications. *Expert Systems with Applications*, 39(17):13051–13069, 2012.

[131] M. John et al. Computational methods for data analysis. Technical report, 1977.

[132] M. Khan et al. Seven V's of Big Data Understanding Big Data to Extract Value. In *Conference of American Society for Engineering Education*, pages 1–5, 2014.

[133] M. Khandelwal et al. Dos attack detection technique using back propagation neural network. In *Advances in Computing, Communications and Informatics (ICACCI), 2016 International Conference on*, pages 1064–1068. IEEE, 2016.

[134] M. Misra et al. Prediction of number of zombies in a ddos attack using polynomial regression model. *Journal of advances in information technology*, 2(1):57–62, 2011.

[135] M. Salehan et al. Predicting the performance of online consumer reviews: A sentiment mining approach to big data analytics. *Decision Support Systems*, 81:30–40, 2016.

[136] M. Siddiquee et al. Association rule mining and audio signal processing for music discovery and recommendation. *International Journal of Software Innovation (IJSI)*, 4(2):71–87, 2016.

[137] M. Skurichina et al. Bagging and boosting for the nearest mean classifier: Effects of sample size on diversity and accuracy. In *International Workshop on Multiple Classifier Systems*, pages 62–71. Springer, 2002.

[138] N. Hooda et al. B2fse framework for high dimensional imbalanced data: A case study for drug toxicity prediction. *Neurocomputing*, 2017.

[139] N. Hooda et al. Fraudulent firm classification: A case study of an external audit. *Applied Artificial Intelligence*, 32(1):48–64, 2018.

[140] N. Littlestone et al. The weighted majority algorithm. *Information and computation*, 108(2):212–261, 1994.

[141] N. Liu et al. Ensemble based extreme learning machine. *IEEE Signal Processing Letters*, 17(8):754–757, 2010.

[142] N. Tsapanos et al. A Distributed Framework for Trimmed Kernel K-Means Clustering. *Pattern Recognition*, 48(8):2685–2698, 2015.

[143] P. Brazdil et al. Cascade generalization. *Machine learning*, 41(3):315–343, 2000.

[144] P. Büchlmann et al. Analyzing bagging. *Annals of Statistics*, pages 927–961, 2002.

[145] P. Chapman et al. CRISP-DM 1.0 Step-by-step data mining guide. www.the-modeling-agency.com/crisp-dm.pdf, 2015. [Accessed: 25-Mar-2018].

[146] P. Martelli et al. An ensemble machine learning approach for the prediction of all-alpha membrane proteins. *Bioinformatics*, 19(suppl 1):i205–i211, 2003.

[147] P. Melville et al. Constructing diverse classifier ensembles using artificial training examples. In *IJCAI*, volume 3, pages 505–510, 2003.

[148] P. Nikolovski et al. The concept of audit risk. *International Journal of Sciences Basic and Applied Research (IJSBAR)*, 27(3):22–31, 2016.

[149] P. Russom et al. Big Data Analytics. *TDWI Best Practices Report Fourth Quarter*, 2(8):20–31, 2011.

[150] Paul et al. Kernel methods for heterogeneous feature selection. *Neurocomputing*, 2015.

[151] Q. Wang et al. Addressing complexities of machine learning in big data: Principles, trends and challenges from systematical perspectives. 2017.

[152] Q. Zhang et al. Secure weighted possibilistic c-means algorithm on cloud for clustering big data. *Information Sciences*, 2018.

[153] R. Barzegar et al. Mapping groundwater contamination risk of multiple aquifers using multi-model ensemble of machine learning algorithms. *Science of The Total Environment*, 621:697–712, 2018.

[154] R. Bryll et al. Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets. *Pattern recognition*, 36(6):1291–1302, 2003.

[155] R. Buyya et al. Introduction to cloud computing. *Cloud computing: Principles and paradigms*, pages 1–41, 2011.

[156] R. Casado et al. Emerging trends and technologies in big data processing. *Concurrency and Computation: Practice and Experience*, 27(8):2078–2091, 2015.

[157] R. Dehkharghani et al. Sentimental causal rule discovery from twitter. *Expert Systems with Applications*, 41(10):4950–4958, 2014.

[158] R. Lichtenwalter et al. Adaptive Methods for Classification in Arbitrarily Imbalanced and Drifting Data Streams. In *New Frontiers in Applied Data Mining*, pages 53–75. Springer, 2010.

[159] R. Nambiar et al. A Look at Challenges and Opportunities of Big Data Analytics in Healthcare. In *IEEE International Conference on Big Data*, pages 17–22, 2013.

[160] R. Neapolitan et al. *Learning bayesian networks*, volume 38. Pearson Prentice Hall Upper Saddle River, NJ, 2004.

[161] R. Ranawana et al. Multi-classifier systems: Review and a roadmap for developers. *International Journal of Hybrid Intelligent Systems*, 3(1):35–61, 2006.

[162] R. Vilalta et al. A perspective view and survey of meta-learning. *Artificial Intelligence Review*, 18(2):77–95, 2002.

[163] S. AlAref et al. A novel ensemble machine learning-based method versus clinical risk scoring for discrimination of individuals who will versus will not experience acute coronary syndrome after coronary computed tomographic angiography: Results from the iconic study. *Journal of the American College of Cardiology*, 71(11 Supplement):A1628, 2018.

[164] S. Bandyopadhyay et al. Hdk-means: Hadoop based parallel k-means clustering for big data. In *Calcutta Conference (CALCON), 2017 IEEE*, pages 452–456. IEEE, 2017.

[165] S. Chen et al. Construct support vector machine ensemble to detect traffic incident. *Expert systems with applications*, 36(8):10976–10986, 2009.

[166] S. Dudoit et al. Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, 19(9):1090–1099, 2003.

[167] S. Džeroski et al. Is combining classifiers with stacking better than selecting the best one? *Machine learning*, 54(3):255–273, 2004.

[168] S. Ilango et al. Optimization using artificial bee colony based clustering approach for big data. *Cluster Computing*, pages 1–9, 2018.

[169] S. J. Russell et al. *Artificial intelligence: a modern approach*, volume 2. Prentice hall Upper Saddle River, 2003.

[170] S. Jia et al. Learning to Classify Gender from Four Million Images. *Pattern Recognition Letters*, 58:35–41, 2015.

[171] S. Jin et al. Towards mapreduce approach with dynamic fuzzy inference/interpolation for big data classification problems. In *Cognitive Informatics & Cognitive Computing (ICCI* CC), 2017 IEEE 16th International Conference on*, pages 407–413. IEEE, 2017.

[172] S. Keerthi et al. Convergence of a generalized smo algorithm for svm classifier design. *Machine Learning*, 46(1-3):351–360, 2002.

[173] S. Kotsiantis et al. Machine learning: A review of classification and combining techniques. *Artificial Intelligence Review*, 26(3):159–190, 2006.

[174] S. Kotsiantis et al. Supervised machine learning: A review of classification techniques, 2007.

[175] S. Liu et al. A Survey on Information Visualization: Recent Advances and Challenges. *The Visual Computer*, 30(12):1373–1393, 2014.

[176] S. Opricovic et al. Compromise solution by mcdm methods: A comparative analysis of vikor and topsis. *European journal of operational research*, 156(2):445–455, 2004.

[177] S. Sra et al. *Optimization for machine learning*. Mit Press, 2012.

[178] S. Ullah et. al. The Rise of Big Data on Cloud Computing: Review and Open Research Issues. *Information Systems*, 47:98–115, 2015.

[179] Safavian et al. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, 21(3):660–674, 1991.

[180] Segaran et al. *Beautiful data: the stories behind elegant data solutions*. " O'Reilly Media, Inc.", 2009.

[181] Sheau-Ling Hsieh et al. Design ensemble machine learning model for breast cancer diagnosis. *Journal of medical systems*, 36(5):2841–2847, 2012.

[182] T. Bikku et al. Hadoop based feature selection and decision making models on big data. *Indian Journal of Science and Technology*, 9(10), 2016.

[183] Tang et al. An analysis of diversity measures. *Machine learning*, 65(1):247–271, 2006.

[184] U. Srinivasan et. al. Leveraging Big Data Analytics to Reduce Healthcare Costs. *IT Professional*, 15(6):21–28, 2013.

[185] V. Ayma et. al. Classification Algorithms for Big Data Analysis, A Map Reduce Approach. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1:17–21, 2015.

[186] V. Bolón-Canedo et al. Recent advances and emerging challenges of feature selection in the context of big data. *Knowledge-Based Systems*, 2015.

[187] V. Kukkala et al. Privacy preserving network analysis of distributed social networks. In *International Conference on Information Systems Security*, pages 336–355. Springer, 2016.

[188] V. Lopez et. al. On the Use of Map Reduce to Build Linguistic Fuzzy Rule Based Classification Systems for Big Data. In *IEEE International Conference on Fuzzy Systems*, pages 1905–1912, 2014.

[189] W. Ho et al. Multi-criteria decision making approaches for supplier evaluation and selection: A literature review. *European Journal of operational research*, 202(1):16–24, 2010.

[190] W. Iba et al. Induction of one-level decision trees. In *Proceedings of the ninth international conference on machine learning*, pages 233–240, 1992.

[191] W. Richard et al. The audit risk model, business risk and audit-planning decisions. *The Accounting Review*, 74(3):281–298, 1999.

[192] W. Street et al. A Streaming Ensemble Algorithm (SEA) for Large-Scale Classification. In *ACM International Conference on Knowledge Discovery and Data Mining*, pages 377–382, 2001.

[193] W. Wang et al. Statistical Wavelet-Based Anomaly Detection in Big Data with Compressive Sensing. *EURASIP Journal on Wireless Communications and Networking*, 2013(1):1–6, 2013.

[194] W. Zhao et al. Pscan: A Parallel Structural Clustering Algorithm for Big Networks in Map Reduce. In *27th International Conference on Advanced Information Networking and Applications*, pages 862–869, 2013.

[195] Wang Shuo et al. A Multi-Objective Ensemble Method for Online Class Imbalance Learning. In *IEEE International Joint Conference on Neural Networks*, pages 3311–3318, 2014.

[196] X. Cui et al. Optimized big data k-means clustering using mapreduce. *The Journal of Supercomputing*, 70(3):1249–1259, 2014.

[197] X. Jin et al. Significance and challenges of big data research. *Big Data Research*, 2(2):59–64, 2015.

[198] X. Wang et al. Parallelized Extreme Learning Machine Ensemble Based on Min-Max Modular Network. *Neurocomputing*, 128:31–41, 2014.

[199] Xu Mao et al. Compliance testing for data quality assurance: Definitions, models and applications. In *International Conference On Signal And Information Processing, Networking And Computers*, pages 412–419. Springer, 2017.

[200] Y. LeCun et al. Deep learning. *Nature*, 521(7553):436–444, 2015.

[201] Y. Piao et al. Ensemble Method for Classification of High-Dimensional Data. In *International Conference on Big Data and Smart Computing*, pages 245–249, 2014.

[202] Yan-Shi Dong et al. A comparison of several ensemble methods for text categorization. In *Services Computing, 2004.(SCC 2004). Proceedings. 2004 IEEE International Conference on*, pages 419–422. IEEE, 2004.

[203] Yegnanarayana et al. *Artificial neural networks*. PHI Learning Pvt. Ltd., 2009.

[204] Ying-Ming Wang et al. Fuzzy topsis method based on alpha level sets with an application to bridge risk assessment. *Expert systems with applications*, 31(2):309–319, 2006.

[205] You Zhu et al. Predicting chinas sme credit risk in supply chain finance based on machine learning methods. *Entropy*, 18(5):195, 2016.

[206] You Zhu et al. Comparison of individual, ensemble and integrated ensemble machine learning methods to predict chinas sme credit risk in supply chain finance. *Neural Computing and Applications*, 28(1):41–50, 2017.

[207] Yu-Dong et al. Predicting triplet of transcription factor–mediating enzyme–target gene by functional profiles. *Neurocomputing*, 74(17):3677–3681, 2011.

[208] Z. Deng et. al. A Scalable and Fast OPTICS for Clustering Trajectory Big Data. *Cluster Computing*, 18(2):549–562, 2015.

[209] Z. Deng et al. Efficient knn classification algorithm for big data. *Neurocomputing*, 195:143–148, 2016.

[210] Z. Dikopoulou et al. A modified fuzzy topsis method aggregating 8.921 partial rankings for companies attractiveness. In *The Application of Fuzzy Logic for Managerial Decision Making Processes*, pages 59–71. Springer, 2017.

[211] Z. Sun et al. A novel ensemble method for classifying imbalanced data. *Pattern Recognition*, 48(5):1623–1637, 2015.

[212] Zhang et al. *Ensemble Machine Learning*. Springer, 2012.

[213] Zhou et al. Stacked Extreme Learning Machines. *IEEE Transactions on Cybernetics*, 45(9):2013 – 2025, 2014.

[214] T. Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[215] D. Finney. *Probit Analysis*, volume 4. JSTOR, 1992.

[216] ML flex. https://github.com/srp33/ml-flex. [Accessed:25-June-2017].

[217] IT Gartner. Glossary (nd). *Accessed October*, 16, 2017.

[218] GATE. https://gate.ac.uk/. [Accessed:25-June-2017].

[219] G.K. Gupta. Introduction. In *Introduction to data mining with case studies*, pages 33–38. PHI Learning Pvt. Ltd., 2011.

[220] C. Hadjigeorgiou. Performance and Scaling. In *RDBMS vs NoSQL: Performance and Scaling Comparison, Master Thesis*, pages 13–38. The University of Edinburgh, 2013.

[221] S. Hamm. Insights in Motion: Deep Analytics Shows How Cities Really Work. www.asmarterplanet.com/life-of-place.html, 2013. [Accessed: 25-Mar-2018].

[222] Mohamed Hassan. A literature study of bottlenecks in 2d and 3d big data visualization, 2017.

[223] J. Hendler. Web 3.0 Emerging. *IEEE Computer*, 42(1):111–113, 2009.

[224] HV Jagadish. Big data and science: myths and reality. *Big Data Research*, 2(2):49–52, 2015.

[225] KNIME. http://www.knime.org/. [Accessed:25-June-2017].

[226] Scikit learn. http://scikit-learn.org/stable/. [Accessed:25-June-2017].

[227] T. Raz L.Wang. Analytic hierarchy process based on data flow diagram. *Computers & industrial engineering*, 20(3):355–365, 1991.

[228] J. Mao. A case study on bagging, boosting and basic ensembles of neural networks for ocr. In *Neural Networks Proceedings, 1998. IEEE World Congress on Computational Intelligence. The 1998 IEEE International Joint Conference on*, volume 3, pages 1828–1833. IEEE, 1998.

[229] M. Marquez. Big Data Analytics for Improving the CERNs Large Hadron Collider Operations. www.ieondemand.com/operations, 2014. [Accessed: 25-Mar-2018].

[230] MEPX. http://www.mep.cs.ubbcluj.ro/olteanmep.pdf. [Accessed:25-June-2017].

[231] Rapid Miner. https://rapidminer.com/. [Accessed:25-June-2017].

[232] Mlpack. http://mlpack.org/. [Accessed:25-June-2017].

[233] National Center for Biotechnology Information. Pubchem compound database. https://pubchem.ncbi.nlm.nih.gov/compound/4463(Accessed: 2017-04-10).

[234] National Center for Biotechnology Information. Pubchem compound database. https://pubchem.ncbi.nlm.nih.gov/compound/5625(Accessed: 2017-04-10).

[235] National Center for Biotechnology Information. Pubchem compound database. https://pubchem.ncbi.nlm.nih.gov/compound/64139(Accessed: 2017-04-10).

[236] Thomas G Neltner. Navigating the US Food Additive Regulatory Program. *Comprehensive Reviews in Food Science and Food Safety*, 10(6):pp. 342–368, 2011.

[237] Thomas G Neltner. Data Gaps in Toxicity Testing of Chemicals Allowed in Food in the United States. *Reproductive Toxicology*, 42:pp. 85–94, 2013.

[238] Netowl. https://www.netowl.com/. [Accessed:25-June-2017].

[239] Open neural network. http://www.opennn.net/. [Accessed:25-June-2017].

[240] NLTK. http://www.nltk.org/. [Accessed:25-June-2017].

[241] Bank note dataset. https://archive.ics.uci.edu/ml/datasets/banknote+authentication. [Accessed:25-June-2017].

[242] American Institute of Certified Public Accountants (AICPA). Understanding the entity and its environment and assessing the risks of material misstatement, 2006.

[243] Orange. https://orange.biolab.si/. [Accessed:25-June-2017].

[244] K. Baid P. Mitra. Targeted advertising for online social networks. In *Networked Digital Technologies, 2009. NDT'09. First International Conference on*, pages 366–372. IEEE, 2009.

[245] K. Baid P. Mitra. Generation of targeted advertisements for online social networks. *IJWA*, 2(2):129–136, 2010.

[246] Judea Pearl. Bayesian netwcrks: A model cf self-activated memory for evidential reasoning. 1985.

[247] R. Polikar. Ensemble learning. In *Ensemble machine learning*, pages 1–34. Springer, 2012.

[248] D. Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. 2011.

[249] P Putrus. Accounting for intangibles in integrated manufacturing (nonfinancial justification based on the analytical hierarchy process). *Information Strategy*, 6(4):25–30, 1990.

[250] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[251] J. Ross Quinlan. Improved use of continuous attributes in c4. 5. *Journal of artificial intelligence research*, 4:77–90, 1996.

[252] R. Seshadri R. Swathi. Systematic survey on evolution of machine learning for big data. In *Intelligent Computing and Control Systems (ICICCS), 2017 International Conference on*, pages 204–209. IEEE, 2017.

[253] P. Rajendra. Belief-function formulas for audit risk. *Accounting Review*, pages 249–283, 1992.

[254] Irina Rish. An empirical study of the naive bayes classifier. 3(22):41–46, 2001.

[255] David Rossell. Big data and statistics. *Metode Science Studies Journal*, 2015.

[256] M. Rozenfeld. The Future of Crime Prevention. www.theinstitute.ieee.org/future-of-crime, 2014. [Accessed: 25-Mar-2018].

[257] O. Rud. *Business intelligence success factors: tools for aligning your business in the global economy*, volume 18. John Wiley and Sons, 2009.

[258] D. Weiss S. Osinski. Open source search results clustering engine. http://project.carrot2.org/, 2017. [Accessed:25-June-2017].

[259] Steven L Salzberg. C4. 5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Machine Learning*, 16(3):235–240, 1994.

[260] V David Sánchez A. Advanced support vector machines and kernel methods. *Neurocomputing*, 55(1-2):5–20, 2003.

[261] SASEM. https://www.sas.com/enin/home.html. [Accessed:25-June-2017].

[262] Schapire. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999.

[263] Microsoft Analysis Services. https://technet.microsoft.com/library/ms175609(v=sql.90).aspx. [Accessed:25-June-2017].

[264] G. Shapiro. Discovery, analysis, and presentation of strong rules. *Knowledge discovery in databases*, pages 229–238, 1991.

[265] Ömer M Soysal. Association rule mining with mostly associated sequential patterns. *Expert Systems with applications*, 42(5):2582–2592, 2015.

[266] SPSS. http://www-01.ibm.com/software/analytics/spss/products/modeler/. [Accessed:25-June-2017].

[267] Statistica. https://www.quest.com/products/statistica/. [Accessed:25-June-2017].

[268] Stephen V Stehman. Selecting and interpreting measures of thematic classification accuracy. *Remote sensing of Environment*, 62(1):77–89, 1997.

[269] R studio. https://www.r-project.org/. [Accessed:25-June-2017].

[270] Torch. http://torch.ch/. [Accessed:25-June-2017].

[271] Evangelos Triantaphyllou. Multi-criteria decision making methods. In *Multi-criteria Decision Making Methods: A Comparative Study*, pages 5–21. Springer, 2000.

[272] John W Tukey. The future of data analysis. *The Annals of Mathematical Statistics*, 33(1):1–67, 1962.

[273] K. Tysiac. Data analytics helps auditors gain deep insight. *Journal of Accountancy*, 219(4):52, 2015.

[274] UIMA. https://uima.apache.org/. [Accessed:25-June-2017].

[275] https://pubchem.ncbi.nlm.nih.gov/. National Center for Biotechnology Information. PubChem Compound Database(Accessed: 2017-04-10).

[276] G. Wills V. Chang. A model to compare cloud and non-cloud storage of big data. *Future Generation Computer Systems*, 57:56–76, 2016.

[277] C. Wang. New ensemble machine learning method for classification and prediction on gene expression data. In *Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE*, pages 3478–3481. IEEE, 2006.

[278] G. Webb. Multiboosting: A technique for combining boosting and wagging. *Machine learning*, 40(2):159–196, 2000.

[279] Weka. http://www.cs.waikato.ac.nz/ ml/weka/. [Accessed:25-June-2017].

[280] Xiuli Yuan. An improved apriori algorithm for mining association rules. In *AIP Conference Proceedings*, volume 1820, page 080005. AIP Publishing, 2017.

[281] H. Zimmermann. *Fuzzy set theoryand its applications*. Springer Science & Business Media, 2011.