The Prediction of Election Outcomes Through Data Mining Techniques

Submitted in partial fulfilment of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

By

Abdul Manan Koli

Enrolment Number (A165081)

Under the Supervision of

Dr. Muqeem Ahmed

Assistant Professor,

Department of CS & IT

SCHOOL OF TECHNOLOGY

September-2020

Department of Computer Science & Information Technology

Maulana Azad National Urdu University

(A Central University)

Gachibowli, Hyderabad, Telangana INDIA

मौलाना आज़ाद नेशनल उर्दू यूनिवर्सिटी مولانا آزاد نیشنل اُردو یونیورسٹی
MAULANA AZAD NATIONAL URDU UNIVERSITY
(A Central University established by an Act of Parliament in 1998)

Department of Computer Science and Information Technology

CERTIFICATE

On the basis of declaration submitted by Abdul Manan Koli, student of Ph.D, I hereby certify that the thesis titled **"The Prediction of Election Outcomes Through Data Mining Techniques"** which is submitted to the Department of Computer Science & Information Technology, School of Technology, Maulana Azad National Urdu University (A Central University), Gachibowli, Hyderabad, India in partial fulfillment of the requirement for the award of the degree of Doctor of Philosophy, is an original contribution with existing knowledge and faithful record of research carried by him under my guidance and supervision.

To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this university or elsewhere.

Dr. Muqeem Ahmed
(Research Supervisor)
Department of CS&IT
MANUU Hyderabad

Date:
Place: Department of CS&IT
School of Technology

Dr. Pradeep Kumar
Head Department of CS&IT
MANUU Hyderabad

DECLARATION

I, **Abdul Manan Koli**, solemnly declare that the thesis entitled "**The Prediction of Election Outcomes Through Data Mining Techniques"** is my original work. The study has been conducted under the guidance of **Dr. Muqeem Ahmed** with the Department of Computer Science & Information Technology, School of Technology, **Maulana Azad National Urdu University** (A Central University), Gachibowli, Hyderabad, India. It is further declared that to the best of my knowledge and belief, it has not been submitted earlier for the award of any other degree, by anyone.

Dated: ___/___/2020

Abdul Manan Koli
Research Scholar
Department of Computer Science & Information Technology
School of Technology
Maulana Azad National Urdu University
Gachibowli, Hyderabad, INDIA

ACKNOWLEDGEMENTS

In the foremost I thank **Almighty Allah** who is the ultimate source of knowledge for providing me with calibre courage and all the blessings, without whose blessings this work would have remained an un-accomplished task.

I want to acknowledge the untiring and prompt help of my supervisor Dr. Muqeem Ahmed, Assistant Professor, Department of Computer Science and Information Technology, School of Technology, who allowed me complete freedom to define and explore my directions in research. I have learned from him how to organize basic concepts and new ideas and how to describe and predict them with clarity under his benevolence and able guidance. He was always ready to provide critical but constructive comments.

I want to express my gratitude to Prof. Abdul Wahid, Dean, School of Technology, Dr. Pradeep Kumar Head, Department of Computer Science and Information Technology, Mr Jameel Ahmed, Assistant Professor, Department of Computer Science and Information Technology for their untiring support, constructive criticism and valuable suggestions as and when needed during this research. I also want to express my heartfelt gratitude to Mr. Mehboob Basha, a professional from software industry, for his technical guidance and valuable feedback during this research.

I am indebted and highly thankful to all faculty members and other supporting staff of the Department of Computer Science and Information Technology, School of Technology, MANUU for their consistent support. I must also thank to all my research colleagues notably, Syed Immamul Ansarullah, Hamza Muthar, Attaullah Niazi, Irfaz Ahmed and Javid Iqbal for healthy discussions, sharing of useful ideas and moral support.

I also want to express my appreciation to all the research scholars of the Department of Computer Science and Information Technology and other individuals for providing their moral support during my study.

Finally, I feel proud to acknowledge the fondness and sacrifice of my beloved parents, wife and other family members who provided me their unlimited affection, love, and support during the entire period of my study. Their consistent support has let me to success in my life and facilitated me in realizing the goal of my research study. My vocabulary fails to acknowledge the affection, care, inspiration and benediction which I received from my whole family members at every stages of my life especially during the research period.

ABDUL MANAN KOLI

ABSTRACT

The forecasting of election outcome remained an interesting topic of research in prominence from pre-historic times and is still a delightful topic of the current era. Despite tremendous improvements and challenging complications, election prediction remains an inspiring task for researchers and political forecasting organisations, with changing scenarios before voting time. There are numerous ways to predict the election outcomes, like predicting the election results using social media (Twitter, Facebook data). However, the main challenge with such social media sites are the presence of number of fake IDs which cannot be ignored. In the literature and practical experiments, many researchers have used random sampling and predict the outcomes of the election but the main problem with such sampling is that they produce biased results because of fewer surveys and it shows results based on the views of voters who have taken part in the survey and excludes the opinion of a large number of voters. Even though the latest advances in technologies like computation have now become the standard of prediction but these technologies are not sophisticated and are time-consuming.

After reviewing the literature and technical reports, it has been found that there are so many shortcomings and limitations in the previous election prediction models. The existing election prediction models are applicable only in some areas having sophisticated infrastructure but in case of Jammu and Kashmir (J&K), these types of models are not applicable as public cannot express their sentiments towards political parties due to internet blockade in the region. In order to predict the election outcomes in Jammu and Kashmir (J&K), one must build the model based on significant election prediction parameters.

In this thesis, the researcher developed a model by taken into consideration the most significant parameters which are central government influence, religion followers, party

wave, party abbreviations, sensitive areas, vote bank, hereditary factor, incumbent party and caste factor. These parameters are identified by political domain experts and their reliability is investigated in the prediction of election outcomes through different feature selection techniques like filter method, wrapper method, embedded method, and finally their average mean is calculated. The enhancements of applying specific investigation technique like Decision Tree, K-Nearest Neighbor, Random Forest, Support Vector Machine, and finally ensemble them into one model for election predictions are tested. Further, to predict election prediction more accurately with minimum error rate, the hyperparameter optimization is performed.

This model is developed using the Jupyter notebook web application and its performance is tested not only by political domain experts but also through measures like sensitivity, specificity, precision, misclassification rate, accuracy, AUROC, and cross-validation with a statistical test like (T-paired test). To increase the efficiency and minimize the error rate, the election prediction model is optimized. Experimental results showed that the ensemble election prediction model outperforms other election prediction models on the hyperparameter settings with the sensitivity of 86%, the specificity of 92%, the accuracy of 89%, the precision of 91%, miss-classification rate of 10.1%, and AUROC score of 90%. The election prediction feature combination subset [Central govt. Influence, Religion Followers, Party Wave, Party Abbreviations, and Sensitive Areas] showed the highest scores using ensemble model.

This model would help political forecasters and general public to have a view of the probable winning or losing chance of political parties or independent candidates' constituency-wise before the announcement of actual results. The model is applicable for the constituencies/areas where people do not have the regular access of social media and exit poll technologies for early prediction. The election prediction rules generated by the model has been evaluated and validated by various political domain experts. The extracted election

prediction rules are suggestive but not definitive as they are based on the Jammu and Kashmir (J&K).

Table of Contents

LIST OF ABBREVIATIONS

ABS	Australian Bureau of Statistics
AUROC	Area Under the Receiver Operating Curve
BJP	Bharatiya Janata Party
CRISP_DM	Cross-Industry Standard Process for Data Mining
DMP	Default Model Parameter
DMX	Data Mining Extension
DSS	Decision Support System
DTC	Decision Tree Classifiers
EDA	Exploratory Data Analysis
EM	Ensemble Model
EPM	Election Prediction Model
FNR	False Negative Rate
FPR	False Positive Rate
GBC	Gradient Boosting Classifier
GNI	Gross National Income
HPT	Hyper-Parameter Tuning
HV	Hard Voting
IEPS	Intelligent Election Prediction System
INC	Indian National Congress
JKN	Jammu and Kashmir National Conference
JKPDP	Jammu and Kashmir People Democratic Party
KDD	Knowledge Discovery from Data
KNN	K Nearest Neighbor
MLA	Member of Legislative Assemble

MLC	Member of Legislative Councils
NDA	National Democratic Alliance
PCA	Principal Component Analysis
RFC	Random Forest Classifiers
RFE	Recursive Feature Elimination
SAS	Statistical Analysis Software
SEMMA	Sample Explore Modify Model Assess
SVM	Support Vector Machine
SV	Soft Voting
TNR	True Negative Rate
TPR	True Positive Rate
WEKA	Waikato Environment for Knowledge Analysis
XGB	Extreme Gradient Boosting

List of Tables

List of Figures

Appendix A

List of Publications

1. **Abdul Manan Koli, Muqeem Ahmed,** "Machine Learning Based Parametric Estimation Approach for Poll Prediction", Recent Advances in Computer Science and Communications (Formerly: Recent Patents on Computer Science), Special Issue, Volume 13, April 2019 (**Scopus Indexed**)
2. **Abdul Manan Koli, Muqeem Ahmed,** "Exploring the Power of Social Media in Election Predictions", International Journal of Recent Technology and Engineering (IJRTE) (**Scopus Indexed**) ISSN: 2277-3878, Volume-8 Issue-2, July 2019.
3. **Abdul Manan Koli, Muqeem Ahmed,** "Election Prediction Using Big Data Analytic A Survey", International Journals of Engineering and Technology (IJET) 7 (4.5) (2018) 366-369 (**Scopus Indexed**)
4. **Abdul Manan Koli, Muqeem Ahmed and Jatinder Manhas,** "An Empirical Study on Potential and Risks of Twitter Data for Predicting Election Outcomes" at International Conference on Emerging Trends in Expert Applications & Security (ICETEAS 2018) held on Feb 17-18, 2018 at JECRC, Jaipur, INDIA (Published by Springer, Nature)
5. **Abdul Manan Koli, Muqeem Ahmed,** "Analyzing the Predicting Power of Facebook data in Election Outcomes" at National Conference on Emerging Trends and Issues in Information Technology & Communication, ETIITC-18, MANUU Hyderabad, INDIA.

CHAPTER 1

1. Introduction

1.1 Background

The field of election prediction remained core area of fascinating not only for political forecasters but also for the general public as well, and is in prominence for millions of peoples throughout the world. This result numerous political scientists to enter in the field of election forecasting system with intend to make accurate predictions before actual results are announced. Election is a selection procedure in which the general publics cast their votes and choose a political leader or representative for a specified time. Such elected representatives take decision for the welfare of people within a specified time. Out of the total 195 countries in the world, 123 countries relied upon democratic government setup, which means present world population is mainly governed by the democratic type of governments as shown in Figure 1.1[1].

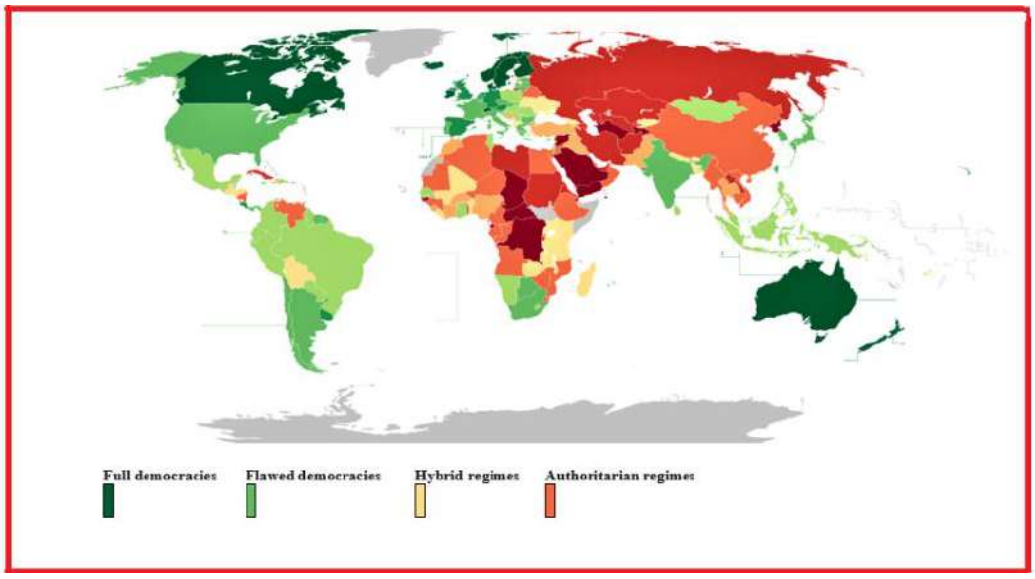

Figure 1.1: Democracy Index Map of the World.

To conduct fair elections numerous countries, describe their own written constitutions. Some countries have indirect type of election (e.g. the United states of America) and some have direct type of election (e.g. European Parliament) with different tenure for election activity to be conducted again and again. Hence it is clear that majority of countries are controlled by constitutional authority that is mainly illustrated as a self-possession organization of citizen political-participation-constraints on the representative's power.

1.2 Motivation

In this work, the researcher predicted the election outcomes of Jammu and Kashmir(J&K) because social media analysis is obsolete due to frequent internet blockade. To the extent of researcher knowledge, this research work is innovative because this is first research work for Jammu and Kashmir and it has not been carried out till date. The election process of J&K is different from rest of the India because in J&K elections are conducted after six years instead of five years. The first assembly election of J&K was held in year 1957 and the same is continue till now[2]. The Jammu and Kashmir witness large scale social unrest since 1957 and because of this there is frequent internet blockade.

Due to these reasons less numbers of people are accessing internet and hence are unable to express their views or sentiments regarding political events. Keeping this under consideration the researcher developed an election prediction model for J&K formulated on significant parameters. The parameters are selected after exploring ever corners of J&K, expert domain knowledge and literature survey.

In Jammu and Kashmir, there are two division, first is Jammu division and second is Kashmir division. The Jammu and Kashmir have total of 87 Assembly constituency seats, out of which 50 constituency seats are in Kashmir province while 37 constituency seats are in

Jammu provision[3]. There are total 22 districts in J&K, out of which 11 districts are in Jammu division and 11 districts are in Kashmir. Because of the geographical and socio-culture ethnic diversity of J&K, it operates at higher levels portraying the complicated picture of society. To begin here J&K have religion, region and caste diversity. The J&K have followers of three major religion i.e Islam, Buddhism, and Hinduism etc. Muslims dominate the J&K as they are in majority followed by Hinduism, Sikhism and Buddhist etc. Besides the religious diversity, J&K witness language, culture, caste and tribal-based diversity. The two regions of J&K are not only culturally and religiously different but are also diverse in terms of geographical terrains as well. As the state capital changes according to the seasons, the Srinagar (in Kashmir division) becomes summer capital and Jammu remained winter capital. Since J&K witness the highest level of disturbance after India got freedom and this affects the J&K drastically be it politically, educationally, and economically etc [4]. Politically J&K witnessed large scale turmoil since 1957 and still the process is continuing now, be it a president rule or coalition governments breakup etc. Since first assembly elections of J&K, National conference (JKN) remained main political party upto 1996 because JKN won maximum number of elections however, in the year 2002 when People Democratic Party (JKPDP) formed coalition governments in Jammu and Kashmir and in the year 2008 Bhartiya Janata Party (BJP) won eleven seats things get changed, as the seat share got divided between different political parties as described in below table 1.2[3].

Table 1.2: Government Formation in Jammu and Kashmir Year Wise

Year	Election	Government formed
1957	First Assembly	JKN
1962	Second Assembly	JKN
1967	Third Assembly	INC
1972	Fourth Assembly	INC and JKN

1977	Fifth Assembly	JKN
1983	Sixth Assembly	JKN
1987	Seventh Assembly	JKN
1990-1996	No Election	President Rule
1996	Eighth Assembly	JKN
2002	Ninth Assembly	JKPDP and INC
2008	Tenth Assembly	JKN and INC
2014	Eleventh Assembly	JKPDP and BJP

1.3 Data Mining and its Applications in Election Prediction

Data mining is defined as a mechanism of extracting useful knowledge from raw data stored in the data warehouse. According to Gartner Group, the important aspect of data mining is to find latest meaningful correlations, patterns and trends by shifting throughout the huge chunk of data saved in different repositories, employing numerous techniques like pattern recognition, statistical and mathematical techniques. However, Cabena et al. described data mining as an interdisciplinary area that acquire together the methods of machine learning, pattern recognition, statistics, database and visualization to acknowledge the problems of knowledge extraction from a large database [5]. Hence, from the above definitions of data mining, one can conclude that data mining is basically an information extraction process from metadata repositories and categorizing or summarization them into meaningful information is a challenging task. Data mining could be utilized to almost all forms of data, be it is data warehouse, transactional databases, relational database, time-series database and worldwide web etc.

The researcher is utilizing data mining in this research work because data mining is an extremely application driven subject of the current era, as it has many outstanding achievements

in different disciplines, be it analytic, health, education or political forecasting. This has made prioritized usage of data mining more successfully and extensively into various applications like web-mining, business intelligence, load forecasting, diagnosis, marketing and sales, oil refinement and screening images etc. Data mining in election prediction is a growing area of high significance because it is providing accurate and early prediction of election outcomes before actual election results announced. The world is overwhelmed by gigantic amount of data which is produced by various heterogeneous sources like mobile phones, sensor network, social media etc at an alarming rate. These datasets are not only vast in size but are also complex in nature like structured, semi-structured and unstructured [6]. Such voluminous and metadata pose a severe threat to traditional database tools like RDBMS because these tools are unable to handle such massive amount of data. In order to prevail over such challenges, data mining techniques came into the limelight, which can easily handle, store and process such meta-data and derive informative analytics from such enormous data [7]–[9]. Today about 2.5 millions of data are being generated at every day and counts of data is raising continuously with every second [10]–[12]. Various researchers applied data mining techniques in their research work and predicted the election outcomes [13],[14]. Some politicians like Mr. Donald Trump even hired data scientist from the Cambridge Analytica organization during his election campaign in 2016 which led to his victory [15].

Data mining has a prodigious ability to examine the hidden patterns in the database of political domain chiefly election predictions. The highest challenges encountered during election analysis and predictions is to build an advanced technology that can present trusted hypothesis using specific measures, that can rely on political forecasting research and applied in political forecasting environments. The extraction or discovering of theoretical implications

from training datasets is a challenging process, and even the best-known domain experts were overwhelmed by such accumulated data. Hence data mining and machine learning technology came in this field because here it requires training machine learning algorithms only once then they will do necessary processing of their own, and this has abetted the political analyst in decision making more accurately. Data mining is changing the landscape of political environments. Now a days mostly winning political parties moves away from traditional methods to data mining and machine learning approach, because the traditional approaches are time-consuming and less effective[16]. But after entering the data mining into the field of political forecasting things start getting change as it attracts the millions of voters just broadcasting one message irrespective of religion, caste, region, gender and location etc. Using data mining with machine learning techniques political parties are able to know the views or demands of voters in the different constituency and then starts talking about the same things that was important to them on different locations [16]. A smarter voter approach means sending a message that is most relevant for people, it may be in any form like the text on social media or banner form. Data mining analytics enables the political parties to know their voter's views on a personal level. The advanced technology of data mining and hybrid tools like anaconda help the political parties to understand the needs and different ways the voters can be approached.

Traditionally, Political parties mainly relied upon that section of populations who whole heartedly votes them and neglects the remaining part of the population during their campaigns. But now, by applying the data mining and machine learning techniques, a good fraction of voters who really don't know much about politics can be targeted. Such voters are called as floating voters and data mining is very useful in smart targeting of such floating voters. These

voters are categorized into different groups based upon their interest and targeted accordingly. Therefore, converting such floating voters towards political parties may increase the chance of getting their votes which is impossible to get. In this research, the researcher applied the data mining techniques for accessing the data of previous elections and predicted the election outcomes. So, it is clear that data mining technologies have an indispensable effect on the data likewise stored data or daily generated data. Data mining with machine learning is all about collecting the data and making some informative decision that helps politicians to know the views or sentiments of people of their constituency and work accordingly. Advances in technology can benefit both the people and Politicians as peoples can express their thoughts and politician can easily know the problems faced by them and solve them. This also enables people to choose right politicians for them by looking at the back record of politicians. Thus, data mining can take elections beyond political campaigns to bring real change and win-win situations for whole nations.

1.4 Description of Research Work

In this work, data mining techniques are used for the development of election prediction model. This model is built in anaconda tool because of its versatility in library packages for data pre-processing, data manipulation and classification. In this research work three main feature selection techniques which consist of Filter Method, Embedded Method and Wrapper Method are applied to select the most significant parameters for early election prediction along with the mean weightage of election parameters. After selecting significant set of election prediction parameters, the researcher applied data mining classification algorithms like Decision tree, Support Vector Machine, Random forest and K-Nearest Neighbor to build the political forecasting model. After mining the election dataset with above mentioned

classification algorithms, it was observed that random forest model performed well as compared to other existing election prediction models. To improve the predictive efficiency of the developed election prediction models and to overcome the overfitting problems, hyper-parameter optimization was performed and examined the utilized machine learning models accuracy. The researcher examined the simulation results of the hyperparameter optimization election prediction models and it was observed that random forest model outperformed others prevailing election prediction models. To improve the efficiency of the election prediction models the researcher applied the soft voting ensembling technique so as to increase the model performance in terms of high accuracy and decrease the model misclassification rate. The soft voting ensembling technique is applied to give the probability means of all the used models. Finally, to check the significance level of the developed election prediction model the researcher applied statistical t-paired test using p value. From election prediction model results it is observed that there is significant difference between proposed election prediction and ensemble model. The model end-result reveals that the accuracy of ensemble model is better than others. Finally, the results of election prediction model are validated through various domain experts particularly from Jammu and Kashmir and generally from India.

1.5 Election Prediction Methods

From last decade prediction markets drew substantial compliments as a tool for predicting the future trends and political forecasting remained the central theme of these prediction environments [17]. Before the invention of scientific polling in the year 1936 political forecaster had a tuff time in gauging the public opinion and forecast the election outcomes [18]. Since 1936, election outcomes were predicted based on survey data or through direct communication with voters [17]. Now in recent time's prediction markets started gaining booming with the

Lowa Electronic Markets. However with an introduction of computational techniques mainly data mining techniques in political forecasting the whole scenario of election prediction become more interesting, because data handling becomes easy [19]. This has led political forecasting a big business, be it news channels for example exit pool, or Political forecasting organization, for example, FiveThirtyEight[20] etc. and so many research scholars throughout the world. Researcher has been using numerous techniques and methods for election predictions some of them utilizing economics parameter [21]–[24] while others include social media [25]–[27] as sole dimension for predictions with data mining techniques like deep learning etc. The era of election prediction happen to new prevalent when Barack Obama efficiently utilized Twitter in his election process [28]. Keeping under consideration his innovative procedure for election campaigning, different politicians and researcher throughout the world applied these techniques in election process and research work respectively.

The main problems with social media is that firstly, they have fake ids [29] and secondly, they are providing the views or thoughts of only those peoples who are accessing them and exclude the sentiments of peoples who are not accessing them [30]. So, in order to overcome such issues, the predictions of election outcomes should be based upon some parameters like Zolghadr et al. [31] predicts the election outcomes for USA based upon certain attributes, similarly other researcher like Singh et al.[32] predicts the election outcomes of Punjab (India) state based upon parameters. In the same way, numerous researcher like Hummel et al.[33], Arulampalam et al.[34], Singh et al.[35], Alam et al.[36], Gill [37], Sawant et al.[38], Jagdev et al.[14] and many others predicts the election outcomes of different areas based upon parameters with computational techniques.

In this work, election prediction model for Jammu and Kashmir is developed using data mining

techniques. The central theme for predicting the election outcomes utilizing the parametric approach is that in Jammu and Kashmir there are frequents internet blockade [39]–[41] so in such situations, people are unable to use social media and express their sentiments towards political parties. In this work, the researcher is predicting the election outcomes for Jammu and Kashmir based upon parameters like central government influence, hereditary etc, and these parameters are selected after consulting domain experts and cross-checked with numerous poll forecasters.

1.6 Objectives

Below is the list of the objectives for this research work:

i. To conduct a systematic literature review about election predictions using data mining and machine learning techniques.

ii. To identify the significant parameters that could accurately predict the election outcome.

iii. To develop the election prediction model using an appropriate data mining algorithm.

iv. To analyze the developed election prediction models efficiency using various metrics.

v. To validate the model using real-world datasets, expert domain knowledge and existing election prediction models.

1.7 Statement of the Problem

To accomplish the above research objectives, this work precedes the reviewed work and overcome the limitations and loopholes in the reviewed literature by incorporating most suitable parameters with advanced and efficient classification algorithms like Random Forest, K-Nearest Neighbor, Decision Tree and Support Vector Machine and finally ensemble them in one classifier. The existing election prediction systems are developed on insufficient parameters

which result erroneous predictions and the tools used doesn't handle large datasets because they are based on random sampling of surveys. Other limitation of the existing election prediction models is they produce biased results because it includes only that portion of population which are expressing their sentiments, taking part in exit poll and exclude the remaining portion of population which are not part of these methods.

The researcher developed an election forecasting model employing machine learning & data mining classification techniques. The prediction model is developed based on significant parameters with various techniques like Support Vector Machine, Decision tree, K- Nearest Neighbor & Random forest. Efficiency of the model is enhanced by applying ensembling technique and to validate the model the statistical t-paired test is used. The prediction model is developed in web application named Jupyter Notebook with Anaconda tool for execution and experimentation of data to predict the election outcome at constituency levels.

1.8 Thesis Outline

This thesis is outlined in following chapters:

- Chapter 1 is an introductory chapter that discussed the background of the research work. It starts with an overview of Election Process, Democracy, Election Process of Jammu and Kashmir, Election Predictions Techniques. This chapter also discussed data mining and its applications in election predictions, statement of the problem, and research objectives.
- Chapter 2 presents a detailed literature review about predictions of election outcomes by practicing data mining techniques. In this chapter researcher also explained the limitations in the previous literature and overcome them in this research work.

- Chapter 3 describes data mining tool and techniques for election prediction models. In this chapter researcher proposed different attributes selection Techniques. Further, researcher explained different classification techniques like support vector machine, decision tree classifier, K-nearest neighbors and Random forest classifiers; alongside with these methods investigators also described various model evaluation metrics.
- Chapter 4 describes the Knowledge Discovery Data (KDD) data mining methodology that demonstrates the thesis objectives. Researcher also proposed the research design which systematically explains each step to build the election prediction model for the accurate prediction of election outcomes. In this chapter, researcher also define exploratory data analysis, and pre-processing techniques to improve the data quality, which improves the accuracy of mining process.
- Chapter 5 describes the results of the developed election prediction models like Decision tree, K-Nearest Neighbor, Support Vector Machine and Random Forest. In this chapter researcher also performed evaluation metrices like sensitivity, specificity, F1 score, AUROC score, accuracy and misclassification rate of each developed election prediction model.
- Chapter 6 explains the hyperparameter optimization techniques and their application on the developed election prediction models. In this chapter researcher performed the soft voting ensembling of the developed models to obtain the optimal election outcomes. Finally, researcher used the statistical T-paired test to validate the model and check its level of significance.
- Chapter 7 describes the conclusion, limitations and future work of developed election prediction model.

CHAPTER 2

2. Literature Review

This chapter outlines the seminal contributions made by various researchers for the development of Election Prediction Model using different data mining techniques. This chapter also highlights the importance of early prognosis and identification attributes based upon which election prediction can be done. Finally, the research gaps found in the prevailing literature are discussed.

2.1 Election Prediction Using Various Data Mining Tasks and Techniques

The Election forecasting is an entangled procedure which is not liberated by wrong assumptions. To forecast the election results, various researchers applied numerous data mining techniques on raw poll data and predict the election outcomes. The conducted study is based on a critical literature review so as to make a precise impression of the potential and the value of imperative parameters with data mining technique for political forecasting purposes.

2.1.1 Election Prediction using Parametric Approach

The main objective of using parametric approach is to find the most significant parameters for election predictions. Different nations and societies have varied parameters for political forecasting.

Following Researchers applied distinctive parametric approach for election predictions using data mining techniques.

Erikson and Wlezien (2008) [42] presented an economic forecasting tool for predicting the US presidential election. The parameters selected for analysis were economic, non-economic (presidential approval or trial heats polls) and previous elections data. In this paper,

classification is applied with root mean square error, which revealed that if Leading economic indicator (LEI) increased by (0.1%) quarterly then chances of incumbent party will be more in the upcoming election. Hence from this finding it can be noted that the economic is most pivotal attribute in deciding the electorate fate.

Pavia et al. (2008) [43] predicted election results based on significant parameters like economy, development etc. with geographical factor. Two geo-statistical techniques like ordinary kriging & ordinary cokriging were used to forecast the election results based on the polling stations spatial locations. Experimental results show that both techniques predict election outcomes with high predictive capability. However spatial forecasts achieve optimal outcomes compared to temporal forecasts when polled stations are large in number and the locations are scattered.

Norpotha and Gschwend (2010) [44] took three graves and potent predicators like the fame factor, the long-term partisan balance and the cost of ruling to predict the election outcomes of Germany. Experimental results showed that the mentioned election factors play hefty and substantive role in election prediction with optimal election prediction results with an error rate of less than (1.3%).

Lewis-Beck and Nadeau (2011) [45] developed an economic model in forecasting U.S presidential election. A survey was conducted with three main dimensions of economics viz valence, position, and patrimony with sub-dimensions age, gender, race, education, income, class, patrimony. Logistic Regression equations applied on the conducted survey and asserted that the economic dimension played an imperative role in government formation and candidate selection. The proposed economic model depicts that Mr. Barak Obama had more chance of winning 2008 election, as economic parameter was concerned.

Toros (2011) [46] proposed an election prediction model based on the turkey data-set having

three theoretical premises. The author found that a vote share change depends on three other key factors -economic condition, local election success and political structure. Lewis-Beck tool was used to check the quality and derive the error rate of the proposed model. Experimental results showed that these factors are crucial for predicting election outcome results.

Singh (2012) [47] designed a fuzzy logic-based election prediction model based on nine divergent election parameters. They used Mamdani technique to determine the output of developed model. After the application of this technique it was analyzed that this method has significant importance in election outcomes.

Kodinariya (2012) [48] created a data warehouse of election data based upon some significant election prediction parameters like Candidate, Time, educational status of voter, religion, age, and Session. Proposed approach is implemented using Microsoft SQL Server 2000 with pixel-oriented technique for visualizations. After proper visualizations of dataset, they claimed that these techniques can be used for generating awareness among general public.

Singh et al. (2013) [32] developed a predictive model by using fuzzy cognitive maps. This election prediction model is based on ten distinctives election prediction parameters. After experimental result, they found that the developed model could be utilized for election prediction before the announcement of actual election results. It is also suggested to add more election prediction parameters so as to improve better predictability of model.

Dahlberg Stefan et al. (2013) [49] analyzed that the parameters like party and political system are highly interrelated and have direct impact on the voters. To prove this argument, they had conducted survey from thirty-two different countries with 86811 respondents. The experimental results showed that the variable like political system had less influence on voters however the party and the individual related parameters had the significant influence on voters.

Hummel and Rothschild (2014) [33] developed an election prediction model based upon different election parameters. The model was developed utilizing linear regression algorithm, which predicted the election with high predictive capability of 90% for president elections, 82% for senatorial elections and 79% for Governor Elections with minimum error rate.

Yu Wang et al. (2016) [50] developed an election prediction model based on significant election parameters. The developed model is trained on twitter data obtained online by using different online data collection techniques. The significant four election prediction parameters like social capital, gender, age, and race were used to train the model. They used the CNN along with Face++ API tool for classification and analysis purposes. Experimental results showed that the model outperform other existing developed model in accuracy.

Mohammad Zolghadr et al. (2017) [51] developed an election prediction model by using the SVM, ANN and linear regression algorithms. The performance of the model is validated through various parameters like mean absolute prediction error and root-mean-squared error. After proper scrutinizing the learning algorithms it was revealed that SVM have better prediction results as compared to ANN. The whole prediction was carried using some vital parameters like GDP, unemployment rate, personal income, etc. Experimental results showed that the model accuracy could be further improved by applying different algorithms on huge amount of dataset.

2.1.2 Election Prediction using Social Media

Social media is considered as a medium where every person expresses their emotions. The data brought about by social media platforms like facebook and twitter can be used to predict the election results. Following Researchers made significant contribution in predicting the election outcomes using data mining techniques with social media data.

Conover et al. (2011) [52] developed an election prediction model by applying classification algorithms. The applied election prediction model is trained on Twitter metadata. In this work semantic analysis is performed to obtain the hidden information generated by twitter users while network clustering algorithm was used for the classification and communication purposes. This method achieved 91% of accuracy rate for predicting election results.

Lei Shi et al. (2012) [53] designed an election prediction model based on Twitter data. Twitter data was collected through tweets for a time frame from September to February 2012. A lasso (Least Absolute Shrinkage and Selection Operator) regression algorithm was used for investigating the prediction problems. The forecasted results obtained from three states were compared with real clearpolitics website. After compiling the results of three states it was revealed that it is possible to predict the US Presidential elections.

Mahmood Tariq et al. (2013) [54] developed an election prediction model using twitter data by employing naive bayes, support vector machine & decision tree algorithms. The election prediction model was developed on rapid miner tool with classification algorithms. The result of which shows that decision tree classifiers strategically out performed to other poll forecasting models with optimal accuracy

Polykalas et al. (2013) [55] used Google trends to analyzed the Germany election outcomes among two famous political parties. After applying the data mining algorithms on data collected from google trends, it was claimed that this method predicts the election result accurately mostly suitably for 2013 and 2009 elections but for 2005 election performance was not upto mark. The reason behind is that in 2005 people were less aware about internet usage as compared to 2009 and 2013.

Fernanda et al. (2013) [56] analyzed the potential of social media for 2012 U.S Presidential pre-elections using four social media platform and seven Republican Candidates. The data was collected using their API's and Technorati, then the collected data was analyzed deeply to obtain the potential of social media about volume, attention and popularity metrics. The final results were evaluated with Galluppoll results which illustrated that the proposed approach revealed significant relationship with 2012 primary outcomes.

Song et al. (2014) [57] developed an election prediction model on the basis of Twitter data through different data mining techniques like network analysis, multinomial topic modeling and co-occurrence retrieval techniques in order to acquired relevant information from Twitter. Experimental results showed that these techniques could be applied on Twitter to extract social trends with admirable accuracy.

Ceron et al. (2014) [58] built a predictive model for two nations namely US Presidential and Italy Primary election held in 2012. Supervised sentimental analysis using Hopkins and King with Mean absolute error was performed on twitter data. The predicted results were compared with Traditional polls survey for both USA and Italy nations respectively. Furthermore, there predicted result performed far better accuracy for both the nations with less Mean Absolute error (0.02% for America), and (1.96% for Italy) respectively when compared with other traditional poll survey methods.

Anjaria et al. (2014) [59] predicted election outcomes using supervised machine learning techniques based on Twitter data. An election prediction model was developed by adopting Naïve Bayes classifier, Support Vector Machine, Neural Network and Maximum Entropy with different feature selection techniques. Experimental results showed that Support Vector

Machine election prediction model performed well when compared with other models with maximum accuracy of 88%.

Wani and Alone (2014) [60] developed a prediction model based upon social media data that is generated in our day to day lives. Twitter data was used for the survey purpose in order to get the user opinion about political parties. The election prediction model was build using the K-NN algorithm.

Nam et al. (2015) [61] proposed an election prediction model using network analysis and linear regression techniques. Data was obtained from heterogeneous social media sites and found that Twitter generated prejudice outcomes however online news remained less prejudice. Two main political parties were selected for social network analysis in which the candidates namely Park Geun-hye remained the central Figures in all four social media environment and out performed exceptionally good then his rivalry candidates.

Kagan et al. (2015) [62] developed an election prediction model based on Twitter data. It utilized diffusion estimation model & sentiment analysis algorithms. The Indian Election Tweet Database (IET-Db) was used to train the model and predict the election outcomes. This election prediction model was found best when compared to the existing election prediction models on the basis of experimental results.

Ullah and Irfan (2015) [63] forecasted the Pakistan local Government election held in Islamabad 2015 using Twitter data. Data is collected in the form Positive, Negative and Neutral tweets employing twitter API for two political parties of Pakistan. After analyzing 2588 tweets, it was predicted that PML-N may win 24.11 seats and PTI may win 25.89 seats, but in actual result PML-N won 21 seats and PTI won 17 seats. Total accuracy of this model was 0.33%. The main problem with this research work is that it used twitter data set for election forecasting

which is only used by 10% population of Islamabad, hence this model may produce biased result as it left 90% population views who don't use twitter.

Conway et al. (2015) [64] developed an election prediction model based on data collected from diverse social sites like Twitter, Facebook and news articles by employing the ADA miner and word state techniques to collect tweets and authentic published articles. Results showed that there exists a non-random relationship among these data. Authors mentioned that although twitter is an essential way for campaigning however traditional techniques also play significant role in it.

Tsakalidis et al. (2015) [65] developed an election prediction model based on Twitter data. The twitter data is obtained using Twitter API and performed Lexicon based approach to know the people's sentiments. Three algorithms used for analysis and comparison are linear regression, Gaussian process, and sequential minimal optimization for regression, with Weka tool, and for detecting error rate Mean Absolute Error (MAE) was employed. After proper mining the dataset, it was acknowledged that Gaussian process achieved the lowest MAE (1.31), followed by sequential minimal optimization (1.35) and linear regression. This seems that Gaussian performed well as compared with this proposed work (Twitter based prediction).

Singhal et al. (2015) [66] built an election prediction model for Delhi (India) held in 2014 using semantic and context-aware rule methods. Lexicon based and rule-based hybrid methods were used for performing sentimental analysis. Further, Stanford parser tool was applied for removing extra words. Experimental outcomes revealed that this method is best to forecast the election results. However, this method can be well suited for urban areas as number of twitter users are more as compared to rural areas.

Jadhav and Deshmukh (2016) [67] built a mechanism for predicting Bihar state election for

the year 2015. Tweets were collected from Twitter API for analysis work. Further tweets were classified as Positive, Negative and Neutral. Naïve Bayes and R Tools were used for sentimental analysis and classification with Mean Absolute error for error detection. Results showed that National Democratic Alliance had maximum Positive Tweets as that of Grand Alliance; however Grand Alliance actually formed government in Bihar. This is because the survey was carried out in urban areas where BJP is the favorable party.

Ismail (2016) [68] developed a model to predict U.S Presidential candidate elections that were conducted in the year 2016. Two contestants - Donald Trump and Hillary Clinton were chosen for analysis work. The data for analysis was collected from Twitter API and 10,000 tweets were collected and analyzed for both candidates. The processing of model was carried out using Polarity Lexicon mode and R Tool. Rigorous user sentiment analysis was performed which showed that Trump have more chances than Clinton.

Oliveira et al. (2016) [69] proposed a method for election prediction by using Rapid miner tool along with natural language processing techniques to extract significant information from twitter datasets. The proposed election prediction model is trained on twitter and traditional data. Experimental outcomes revealed that the developed election prediction model has an efficient accuracy in predicting election outcomes with less error rate.

Wang and Lei (2016) [70] developed a hybrid election prediction model using peer-to-peer ratings, candidate mentioning volumes & sentiment scores. The performance of this election prediction model is evaluated using time series and regression analysis respectively. Experimental results showed that the developed model performs well as compared to existing election prediction models with optimal accuracy.

Sharma and Moh (2016) [71] predicted election outcome of India nation using Hindi Twitter

data. The election prediction model was developed based on the support vector machine and Naïve bayes algorithms alongside with Hindi-sentwordnet tool. After applying data mining techniques, it was predicted that the developed election prediction model can be used for political forecasting.

Maurice Vergeer (2017) [72] developed an election prediction model using Twitter data of five varied countries from the years 2010 to 2012, by employing the negative binomial and regression techniques with R mass package. The experimental results obtained from the election prediction model are optimal however additional improvements is required.

Mellon and Prosser (2017) [73] developed an election prediction model on the basis of twitter and facebook data by using the linear regression algorithm. The data is obtained from heterogeneous data groups like demographics, political attitudes etc. After proper scrutinized the data set, it was observed that the users of Facebook and Twitter have distinct opinion and viewpoints regarding various political systems; which includes age, education, gender, vote choice and turnout. Experimental results of this election prediction model shows that the model is ideal however further refinements is required to get optimal accuracy.

Ranjan and Reza (2017) [74] predicted election outcome of Gujarat state (India) using Twitter data by employing deep learning approach and decision trees for sentiment analysis, with word2vec model for processing purpose. Two main political parties namely Congress and BJP were selected and Tweepy with NLTK (Natural Language Toolkit) in python library were used. After mining the twitter data, it was observed that this method can be used for political forecasting.

Jain and Kumar (2017) [75] built an election prediction model to predict the Delhi election for the year 2015. The model was developed by using the Support Vector Machine, Naive Bayes

Classifier, Decision tree, and Random Forest algorithms. Support vector machine was well performed as compared to others proposed classifier with 79.4% accuracy. The developed model anticipated a landslide triumph for Aam Aadmi Party (AAM) which shows that this model can be used for prediction works.

Navya et al. (2017) [76] proposed a model for predicting the Indian Parliamentary election using twitter data. Three classifiers namely ARIMA, CVAR and improved CVAR with Java language were used. Twitter data was collected and analyzed using the said three classifiers to improve the ability of model in terms of prediction.

Goyal (2017) [77] developed an election prediction model utilizing K Nearest Neighbor and Naive Bayes algorithm. The proposed model was trained on twitter data and on different writers' comments. After data analysis it was found that the social media data can be used to predict the sentiments of people. The model is good enough to predict the election outcomes however further enhancements may be requisite.

Safiullah et al. (2017) [78] used the regression analysis techniques for data analysis of twitter data. The developed model performance is checked by using the root mean squared error. It was demonstrated from experimental results that social media Buzz could be availed as a hefty tool for political forecasting with optimal accuracy.

Gaurav et al. (2017) [79] developed an election prediction model by using the text and mining methods on the online twitter data. The data analysis was carried out on R studio with Natural Language Processing (NLP) and twitter API for early predictions of election results.

Suarez-Hernandez et al. (2017) [80] developed an election prediction model based on the mood analysis methodology. The proposed methodology is applied to predict the social sentiments revealed on the twitter. Naive Bayes algorithm was used to classify the user tweets

into positive and negative labels. The experimental results of proposed model strategical proved that it can be used as an excellent model for observing online behaviors of users towards various political issues especially at polling times.

Singh and Sawhney (2017) [81] built a political forecasting model using Twitter data in the form of tweets obtained from various countries. After proper scrutinized it was found that the model provides optimal results in those countries in which internet users are above 80%, however for countries with less internet usage this model is less optimal.

Narwal Neetu (2018) [82] predicted the Delhi MCD election held in 2018 using Twitter data in the form of tweets and analyzed them using R Tool and K-means clustering algorithms. Three main political parties were chosen for analysis and predictions were AAP, BJP and Congress. After proper analysis it was revealed that BJP has got more positive comments and tweets as compared to other rivalry parties, hence BJP has more chance of winning elections.

Hasan et al. (2018) [83] developed an election prediction model using weka tool with classification algorithms viz support vector machine and Naïve Bayes. Sentimental analysis of twitter data was performed using three different sentimental lexicons like (W-WSD, SentiWordNet, TextBlob). Finally, it was analyzed that out of all the three sentimental analyzers, TextBlob has more accuracy rate as compared to other sentimental analyzers.

Imane El Alaoui et al. (2018) [84] built an election prediction model by employing big data techniques with Twitter data. Twitter data was collected using apache Kafka and stored in HDFC (Storage) and spark was employed for processing. Sentimental analysis of tweets were performed for two candidates which shown that the developed model has higher accuracy of election prediction.

Mazumder et al. (2018) [85] proposed an election prediction model to predict the election

outcomes. The model is developed on the twitter data, twarc library and twitter API. The proposed election prediction model is developed based on ANFIS (Adaptive Neuro Fuzzy Inference System) algorithm. RMSE (Root Means Square Error) was used to calculate the error rate which is 16% of error rate and 84% of accuracy.

Thampi et al. (2018) [86] proposed a model for predicting the election results of Indian sub-continent for 2014 using sentimental analysis. Two politically parties were chosen namely Congress and BJP with Twitter data for their analysis work. Three main classifiers were used in this work were Naive Bayes (NB), Support Vector Machine (SVM) and Maximum Entropy (ME) classifiers. After Proper classification of tweets into positive, negative and neutral, BJP was predicted as winner with an accuracy of about 60–65%.

2.1.3 Election Prediction using Previous Election data

This section deals with the research work based on the past election dataset which is used for making prediction of future elections results [87]. Following is the list of work done on the basis of past election dataset of numerous nations with different time span.

Gill (2008) [88] developed an election prediction model for Indian Lok Sabha elections. Neural network algorithm was applied on survey data that was obtained from general public of India. Two-layer neural network was applied which comprised of input from layers and fixed rate was used for this purpose. The model is trained on nine different datasets in such a way that 8 datasets are used for training purposes and the remaining dataset is used for the test purpose of the model. After Applying Neural Network, it was claimed that this model predicted six elections correctly and three wrongly. Hence it can be used as forecasting tools with further improvements.

Arulampalam et al. (2008) [34] designed a theoretical election prediction model by incorporating the linear regression algorithm. The model is trained on data set of different states from 1975 to 1996. Experimental results showed that those states which have strong relationship with Centre government receive higher grants while other states which are non-aligned or partially aligned to the union government receives less grants.

Campbell and Lewis-Beck (2008) [89] designed an election prediction model based on a systematic literature review from years 1979 to 2008. The model was used for election prediction and analysis based upon its experimental outcomes. However, the designed phase of model requires further improvements with refinements in data, parameters and technologies.

Murray et al. (2009) [90] proposed two variables for U.S Presidential elections to developed an election forecasting model. Two survey were conducted which consisted of variables, vote intend and previous presidential voting for 1964, 1976, 1980 and 1988 elections. After applying Iterative Expert Data Mining likely voter 2 (IEDM LV2) technique with decision tree the authors asserted that this model predicts the election result with an accuracy of 78%.

Munzert (2017) [91] proposed time-series method to forecast 2013 German election based on survey polls and previous election results. After properly mining the authors claimed that this model predicts 299 constituency seats correctly with 2.5% mean absolute error.

Dassonneville et al. (2017) [92] built an election prediction model for Netherland held in 2017 using Least Square Regression algorithm. The model was trained on eighteen different election datasets with varying number of election parameters like GDP, Unemployment, previous success and time duration in office. Experimental results showed that this model can be used for political forecasting with minimum least Square error.

2.1.4 Election Prediction using Hybrid Approaches

The combination of several methodologies within a design of single system results into a hybrid system. The Hybrid systems extract the best from all the methodologies and provide an optimal solution for the election predictions. The following work utilized numerous data mining techniques for forecasting of election results.

Draper and Riesenfeld (2008) [93] developed an election prediction model with sophisticated user interface that permits users to build queries on metadata and visualize results in minimum time requirement. The model is novel in its nature because an interactive visualization allows the user to query and analyze tabular demographic data so that naive and professional user ascertains huge amount of data. After complete analysis it was revealed that the designed method plays significant role in analyzing opinion poll of data.

Rigdon et al. (2009) [94] developed an election prediction model for United States held in 2004 based on Bayesian algorithm. It acknowledged that to obtain the optimal election outcomes with high predictive capability the dataset should be pre-processed with ultimate care because the noisy dataset degrades the performance of the proposed election prediction model.

Vreese (2009) [95] analyzed the expanded scope of political election campaigns of eight European countries for 2004 elections. After analysis it was observed that nations varied in term of campaigns, strategies for election, Peoples are less active in second order election in Europe than then first order elections.

Armstrong et al. (2010) [96] utilized facial expression to forecast the outcomes of U.S presidential election. For analysis, facial competence ratings of 11 candidates from Democratic Party and 13 from Republican Party were selected respectively. Photographs of candidates were

used for election forecasting and took decision on the basis of facial competence. This model claimed that Mr. Obama will have maximum rating among all, so his chances are more.

Lewis-Beck and Stegmaier (2011) [97] forecasted British elections based upon people's opinion collected through surveys. Two methodological approaches for this task were applied- (i) statistical models and (ii) opinion polls. Four elections from 1992 to 2005 were tested with the obtained survey, which showed that the proposed method accurately predicts the election results. Finally, predictions for 2010 election were carried out by merely adding an online survey from internet and telephone, which predicted that the conservative party may win the elections. The prediction was found accurate when conservative party won the elections.

Ford et al. (2015) [98] developed a method which consist of three stages for predicting the parliamentary election outcomes from voter preferences in British opinion polls. The first stage was to collect data from various polling agencies then secondly forecasting the voter intention for political parties then finally simulates the result. Overall, this method combines national, local and regional data and forecast the election results.

Khatua et al. (2015) [99] developed an election prediction model to predict elections outcome of India. Heterogeneous data was collected from different sources and techniques like Lexicon method for sentimental analysis and OLS Regression model for vote swing were employed. The Mean absolute error technique was used to detect the error of the model.

Rallings et al. (2015) [100] proposed an election prediction model in which local election results were used to predict the national level results for UK parliamentary elections. The main aim of this forecasting is to recognize various constituencies where local level pattern supports to which main political parties namely Conservative party, Labour party or Liberal Democrat. After properly scrutinizing, the model revealed that there is maximum chances for Labour party

to won 280 seats as compared to Conservatives party with 275 seats and Liberal Democrats with 22 seats respectively. Hence this model can be used for political forecasting.

You et al. (2015) [101] built a poll forecasting model for the prediction of election outcomes. The Competitive Vector Auto Regression (CVAR) model was proposed for prediction purposes. The flicker data which comprises of textual and information data was used for data analysis. The experimental results obtained were compared with different election prediction forecasting models like VAR and CR and concluded that CVAR performed well as compared to other models with optimal election results.

Jagdev et al. (2016) [9] applied the Hadoop and map reducing techniques to analyze the data. The vast amount of heterogeneous data is stored using the Hadoop tool and the map reduce algorithm was used to sort and process the relevant data. Big data techniques are used for extracting relevant information in much shorter time with very low cost in contrasts to traditional tools. Experimental results showed that after proper data mining, the big data techniques were found efficient in predicting the election outcomes with optimal accuracy.

Xie et al. (2016) [102] predicted presidential election outcomes of Taiwan during 2016 year by collecting the online and offline public opinions. A model was proposed by utilized Kalman Filter techniques and Moving average model for real time burst detection on heterogenous data with less than 3% error rate. The obtained result seems promising and can be used for election predictions.

Xu and Liu (2017) [103] developed an election prediction model for 2016 Hong Kong legislative council based on data mining and machine learning techniques. The model was developed using Apriori algorithm and its accuracy is developed using delegating and

weighting operations. Researcher collected the data from general survey and opinion polls. Experimental results showed that the model accuracy is 82.5% and can be further improved.

Sathiaraj et al. (2017) [104] developed a hybrid machine learning approach using campaign specific voter share algorithms (CVS) to predict election outcomes for three Louisiana elections. The proposed election prediction model produced an outcome with optimal accuracy and minimum error rate. The developed model is validated with different existing models and from the experimental results it was derived that this methodology can forecasts the polls results more accurately.

Cristina et al. (2018) [105] proposed a technique for predicting the Brazilian 2016 municipality elections. Sentimental analysis of online news through likes dislikes and comments from four different newspapers was performed. Weka tool was used for building model with multivariate linear regression techniques for cross validation and M5 methods for attribute selection. For measuring the error rate, mean absolute error rate was practiced. After proper mining it was revealed that most positive comments and likes means candidates have more chance of winning elections and hence this method can be used as an accurate technique for predicting polls result.

Sharar and Abd-el-Barr (2018) [106] used the online survey data to predict the election outcomes. The online questionnaire survey was distributed between the months of March & April 2016. A total of 637 records were collected to build the model. SPSS tool was used to analyze the survey data. The experimental results showed that social media data especially twitter data play crucial role in developed countries for the election outcomes.

Pranay Patel (2018) [107] described the vital role being played by psychometric big data analytic (third party Cambridge Analytica) for election analysis and predictions. This Cambridge Analytica collect the record of general public from Facebook, election commission,

automotive store and then advertise the campaigning team of Donald Trump accordingly so as to target the voter as per their psychology wishes. This technique plays well for Trump as he targeted the masses with his speeches and campaign both online and offline as per voter wishes, which efficiently leads to Trump's Victory.

2.2 Research Gaps

Election outcomes can be predicted using several methods; however, the most cost-effective and reliable methods are based on the assessment of election prediction attributes. Various researchers predicted election outcomes before actual result announced using different data mining techniques, but a closer look at the reviewed literature reveals several shortcomings which are described as:

i. Most of the developed Election Prediction models lack generalization capability.

ii. Mostly the election prediction models were based upon the data collected from social media site like Twitter and Facebook which may produce inaccurate results because of the presence of large number of Fake ids.

iii. The data collected from Twitter or Facebook for making election prediction can be done only in developed nations because in developed nations peoples have more access to internet as compare to developing nation. For developing nations, it may predict biased results as only small portion of population using it.

iv. By using bots in Twitter one can post as many tweets as he/she wants in supporting any political parties to measure the public's sentiments hence could not be a great technique to predict election outcomes.

v. The internet gap is the significant problem and prevents significant samples from being gathered especially in developing areas.

vi. The existing election prediction model can be applicable only in developed areas but in case of developing areas like Jammu and Kashmir (J&K) it cannot be used because in J&K there are frequents electricity cut and internet blockade and one cannot use internet to express their sentiments towards political parties.

vii. Different tools were used for experiments and simulation purposes; however, each tool has complications accompanied by it. There are many mechanisms for prediction but all have limitations like documentation for GUI is limited, scaling is a problem, Big Data cannot be handled, etc.

viii. Election Prediction evaluations performance measures like sensitivity, specificity, accuracy, precision, etc. were used; however, the model measures like computational complexity, scalability, robustness, and comprehensibility and most important statistical test are not used by the researchers.

ix. Most of the researchers used only a single feature selection technique to get the significant attributes; however, there are no investigations of using multiple feature selection techniques to derive the significant attributes with their mean values for election prediction.

2.3 Proposed Solution to Overcome the Limitations

Keeping under consideration the above research gaps, the researcher in this research work proposed an election Prediction model to overcome such research gaps as described follows:

i. The researcher identified new parameters like central govt. influence, caste factor and sensitive areas etc. and by adding these election parameters the level of predicting election results increased.

ii. The researcher applied three different feature selection methods in association with four different classifications algorithm and finally performed soft voting ensembling to build a

robust election prediction model. Till date we did not find any such research work that incorporated these techniques for feature selections and election result predictions.

iii. The proposed model is built upon different set of significant parameters instead of social media parameters because J&K witnessed frequent electricity cut and internet blockade.

iv. This research work is novel in its nature because researcher did not find any such research work pertaining to constituency wise election result prediction for the area of Jammu and Kashmir.

2.4 Summary

This chapter presented a detailed review of literature for election prediction methods using data mining techniques. Mostly the developed election forecasting models used parameters like gross domestic products, previous term record of governments, unemployment rate, education qualifications, and president approval rates etc. However, they are missing some vital parameters like central government influence, caste factor and sensitive areas etc. which plays a pivotal role during the election process in developing nations. The main limitation of the reviewed research work is that most of them used only one algorithm like neural network, etc. or combinations of more than one classifier like Artificial neural network and support vector machine in their poll predictions models. The parameters and techniques adopted by them for political forecasting are not strong enough to make accurate predictions in developing nations. In order to overcome these issues, the researcher built an election prediction model on significant election parameters like central government influence, sensitive area, and hereditary factor etc with advanced computational techniques like Support Vector Machine, Random forest, Decision Tree and K-Nearest Neighbor algorithms and finally ensemble them using soft voting.

CHAPTER 3

3. Research Tools and Techniques for Election Prediction

3.1 Introduction

Most of the raw data in election databases is unprocessed, incomplete, and noisy and to transform this raw data into useful knowledge, data mining tools and techniques are used. In this chapter, the researcher transforms the raw election data into an appropriate form and applied data mining algorithms to predict election outcomes. This research identified a significant subset of election attributes for early prediction of election outcomes. The researcher used the feature selection techniques to get optimal features for political forecasting. In this chapter the researcher also discusses the model validation techniques and finally wrap up by chapter summary and conclusion.

3.2 Data Mining Tools

A number of available data mining tools can be utilized for model building; however, this research discusses only those tools which were used by researchers in forecasting the election outcomes.

3.2.1 WEKA

WEKA (Waikato Environment for Knowledge Analysis) is a tool of machine learning algorithms that contains tool for pre-processing of data, classification, regression, clustering, association rules and visualization [108]. It can be accessed through GUI, standard terminal application or java API. It is widely used for teaching standard machine learning tasks however due to its out of Memory error it cannot be used for high dimensional datasets [109].

3.2.2 Rapid Miner

RapidMiner is a data science framework which provides data preparation, machine learning and predictive model deployment [110]. It comes up with a unified platform for deep learning, text mining, machine learning and predictive analysis[111]. RapidMiner slows down the system due to excessive memory usage hence is less efficient for research purposes and real-time systems [112].

3.2.3 Orange

Orange is a non-propriety machine learning and data visualization toolkit, written in python programming language [113]. Orange is component-based software called as widgets which range from data visualization & pre-processing to an assessment of algorithms and predictive modelling. Orange help end-users to spawn smart decisions in short time by quickly comparing and analyzing the data [114].

3.2.4 Matlab

Matlab (Matrix Laboratory) provides easy access to matrix software [115]. It is a language for technical computing that combines computation, visualization & programming environment with good throughput. MATLAB has a delicate data structures, having implicit editing and debugging mechanisms and abetted object-oriented programming. Hence, the said parameters makes the MATLAB as an efficient tool for teaching and research [116], [117].All the described tools have some limitations which are mentioned in below table 3.2.

Table 3.2: Limitations of Existing Tools

Tools	Limitations

WEKA	No proper documentations with limited storage problem. It has a bad connectivity with excel spreadsheet and non-Java based databases.
Rapid Miner	This tool is less eco-friendly while using multimedia like video, pictures and audio. Rapid Miner took lot of processing time even on small datasets, especially when the user is optimizing manually different attributes based on the results.
Orange	Orange supports limited machine learning algorithms. Orange can compute basic statistical operations however it is weak in classical statistics. Orange provides no widgets for statistical testing.
Matlab	As it is an interpreted language tool and its execution speed is very slow. So, Matlab does not support asynchronous events.

To surmount the limitations and pitfall of above-mentioned tools as shown in table 3.2 the researcher will utilize anaconda tool (Package) and Jupyter notebook with in python language in this research work for developing election prediction model.

3.2.5 Anaconda

In this research work the researcher used anaconda that is an open source tool with inbuilt data processing and manipulation facilities. Anaconda is basically a data science stack that consists of more than 1000 powerful libraries based upon python and other programming languages [118]. The researcher is using ‘conda’ a package manager, which consists of hundreds of packages of python language and utilized these packages in this research work to perform data pre-processing, classification and validation. Because of the inbuilt machine learning algorithms, anaconda help us in getting an easily manageable environment setup which can deploy to our election prediction model with single click [119].

To develop the election prediction model, researcher used Jupyter notebook web application. The front end of the election prediction model is developed using flask [120], [121]. The

developed model is uploaded on Heroku that is a container-based cloud Platform as a Service (PaaS) [122].

3.3 Machine Learning Techniques

Machine learning is a field of data mining that gives the computer a capability to automatically learn from previous data or information without being explicitly trained[123]. The basic premise of machine learning is to build a computer programme to access the data and learn for themselves. It then uses this data (labelled or unlabelled) for forecasting the futuristic outcome [124]. Machine learning techniques are most widely used techniques of the present era and are estimated to be the effective tools for future predictions. Machine learning techniques are mostly categorized into four different types as shown in Figure 3.3.

i. *Supervised learning:* It is a type of learning in which machine can learn from labelled data that means some data is already tagged with the correct answer, for example image classification problems, in which the objective is to find either the image is of car or bike [125]. Supervised machine learning techniques are categorized into two types i.e classification and regression.

ii. *Unsupervised learning:* It is a type of learning in which machines are being trained upon information that is neither labelled or classified [126]. Here the task of machine is to categorized the unlabelled data into groups or patterns based upon similarity or dissimilarity measure without any prior training of data. Unsupervised machine learning techniques are of two types i.e. clustering and associations.

iii. *Semi-supervised learning:* It lies between supervised learning and unsupervised learning. In this type of machines learning, machines are trained on both labelled and unlabelled data. The goal is to combine both the labelled and unlabelled data that can change the learning behaviour of computer and design of an algorithm for such type of combinations. Hence, in semi-

supervised learning, the algorithm is trained on both the labelled as well as non-labelled data. Typically, the combinations contain a huge chunk of unlabelled data with a less amount of labelled data. For example speech analysis & internet content classification etc [127].

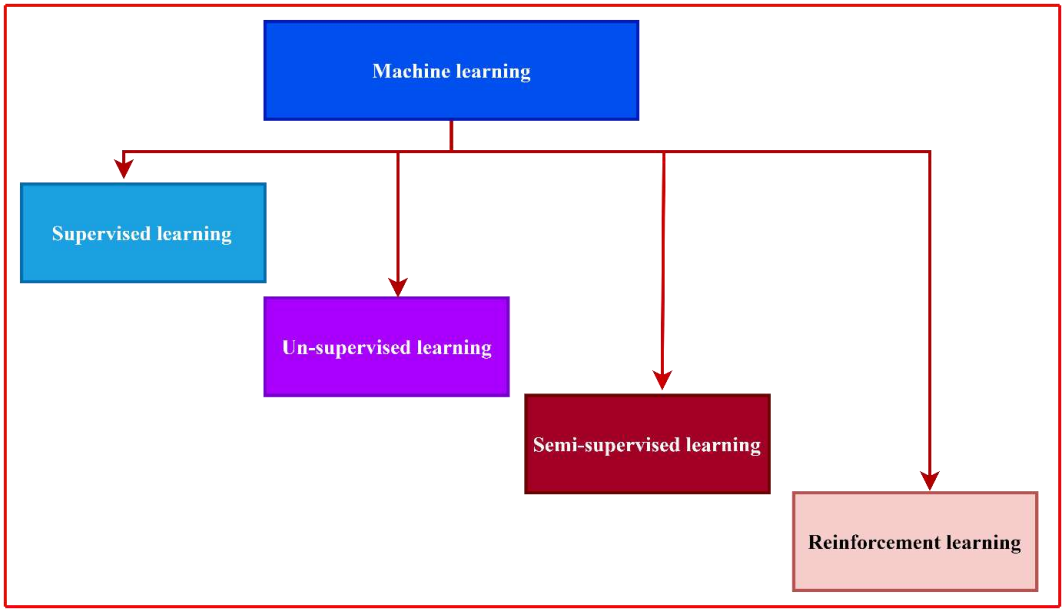

Figure 3.3: Machine Learning and Its Types drawn from [128]

iv. *Reinforcement learning:* It is also known as reward-based learning. In reinforcement learning agents learn by interacting with the environments [126]. The agents received awards for every correct step and penalty for every false step. Reinforcement learning works in contrast to other machine learning that it is not told how to work on any problems, but it works the problem and comes up with solutions through its own [129]. For example, self-driving car and chess game etc

In this research work, the researcher used the supervised classification machine learning techniques for the development of election prediction model.

3.4 Feature Selection Techniques

Features selection method is a technique for selecting the attributes that are most relevant for model construction. It chooses subset of appropriate features according to specific relevance

evaluation condition that provides optimal learning accuracy, lower computational cost, and better model interpretability. These are mostly classified as follows

a) supervised techniques

b) unsupervised techniques

c) semi-supervised techniques.

Supervised feature selection methods are further classified into filter methods, wrapper methods and embedded method as shown in the below Figure 3.4.

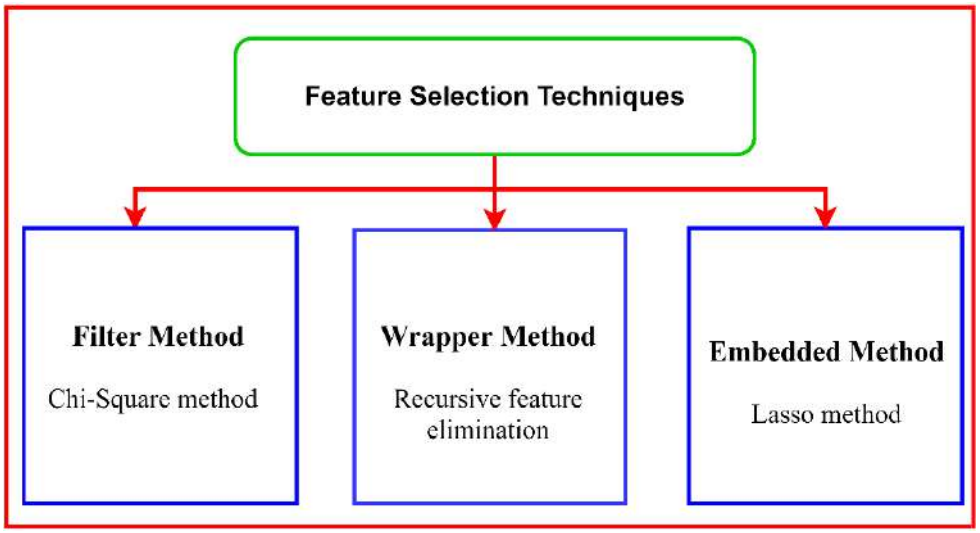

Figure 3.4: Categorization of Feature Selection Techniques

These techniques reduce unrelated attributes that diminish the running time of the learning algorithm. In this research work, the researcher applied filter, wrapper and embedded feature selection methods.

3.4.1 Filter Method

Filter methods assign some score to each attribute based upon some statistical test. The features are ranked according to the score obtained in the dataset and based upon the obtained score the attribute are removed or kept [130]. Filter method consists of two steps, it grades features on some conditions and then chooses highest ranking features to induce classification

models [131]. In this research work Chi-square statistical test is applied in filter method to decide the dependency of two variables [132]. Below Figure 3.4.1 shows the working of filter method for providing the weightage to the attributes.

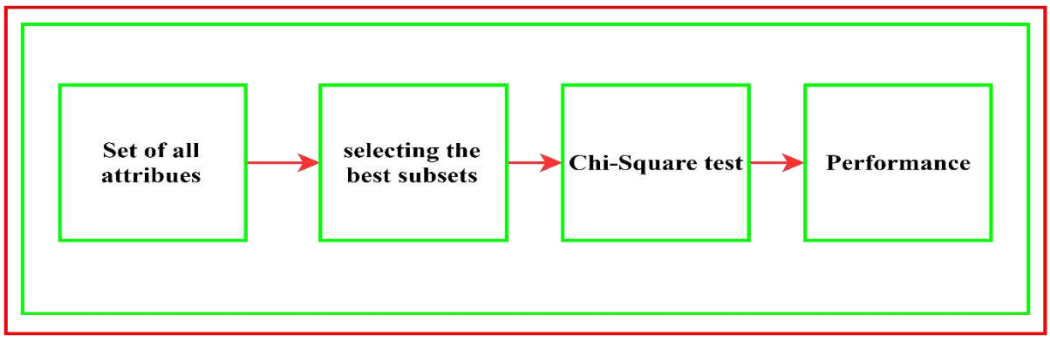

Figure 3.4.1: FilterMethod for Election Parameter Subset Selection drawn from [133]

In this research work the researcher calculated chi-square statistics among all dependent attributes and independent attribute. Finally, a relationship among dependent and independent attributes is calculated. If they are independent, we can discard them else the features are kept.

3.4.2 Wrapper Method

Wrapper methods are based upon the greedy search algorithms (i.e it uses the results of a classifier to calculate the goodness of feature). This technique evaluates all the feature combinations and select the combination that produces the best result for machine learning algorithms. Wrapper method applies a particular algorithm to calculate optimal attributes, and provide an easy way to deal feature selection problems, regardless of the chosen machine learning classifiers [134]. Given a predefined algorithm, a classic wrapper model will perform the following steps:

a) Searches a significant subset of attributes

b) Evaluates the selected subset of features by algorithm

c) Repeats Step I and Step II till the required quality is achieved.

In wrapper methods the researcher applied recursive feature elimination method that fits a model and eliminates irrelevant feature still the required features are obtained [135]. Cross validation is applied in RFE to get relevant number of features[136], [137]. The working of wrapper methods is shown in Figure 3.4.2.

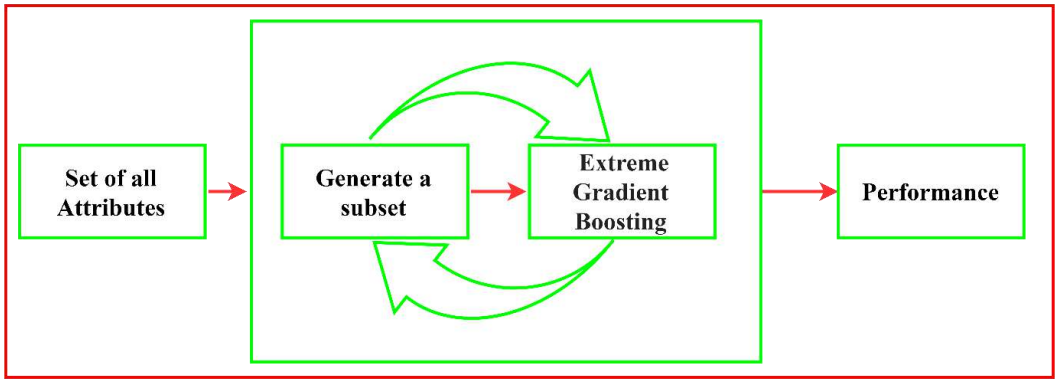

Figure 3.4.2: Wrapper Method for Election Parameter Selection drawn from [133]

3.4.3 Embedded Method

An Embedded Method (or hybrid method) is a combination of filter method and wrapper method in which a feature selection method is incorporated into learning algorithm and is then optimized. Embedded feature selection methods decreases the computation time taken up for reclassifying different subsets [133].

In this research work the researcher applied lasso feature selection method. Lasso regression is an embedded method that selects subset of attributes to minimize prediction error [138]. Lasso works by applying a condition on the model attributes that causes the regression coefficients for some variables to shrink toward zero. Attributes with regression coefficient equal to 0 are excluded from the model and non-zero regression coefficients are included in model [139].

3.5 Data Mining Tasks

The main goal of data mining is to learn from the data. Data mining tasks are utilized to identify the novel patterns found in the data mining process. Data mining tasks are usually classified in two types: Predictive Tasks and Descriptive Tasks as shown in below-given Figure 3.5.

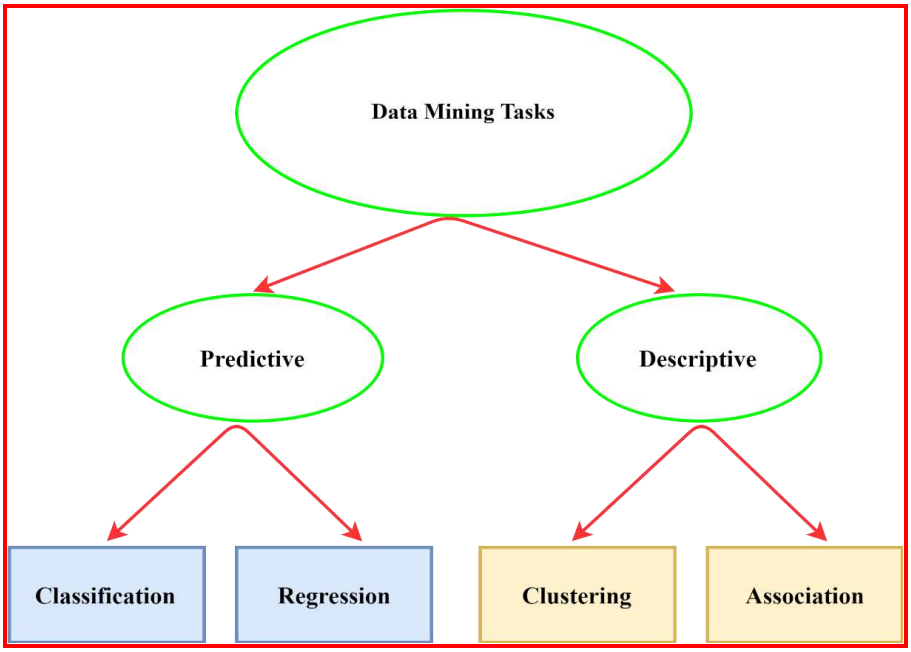

Figure 3.5: Categorization of Data Mining Tasks drawn from [140]

In predictive tasks, the value of dependent feature is predicted on the basis of exploratory features however in descriptive tasks the novel patterns are derived which explain the cardinal association in data. Descriptive data mining tasks are frequently exploratory in nature which needs post-processing methods to evaluate the outcomes/results.

3.5.1 Predictive Data Mining Tasks

Predictive modelling can be defined as the process of development mechanism for the desired variable as a function of the explanatory. Classification & Regression/Prediction are the two kinds of predictive modelling processes.

i. Classification: Classification data mining applications are predictive (supervised) learning data mining tasks that predict *discrete values of the target attribute*. If the target attribute values are Boolean types (*yes or no*), then it is called as the *binary classification*. But if the target attribute has multiple possible values then that is called as the *multi-class classification* [140].

ii. Regression: Regression modelling can be defined as the process of making a system to address the regular interval variables as a function of the explanatory variable. For instance, to predict the cost of an item for three months from now as the cost is considered as a continuous-valued attribute [141].

The aim of classification and regression is to have a system which reduces the error amid the predicted and true values of the desired variable.

3.5.2 Descriptive Data Mining Tasks

Descriptive modelling means the way of deriving the patterns which explains the fundamental relationships in the data. There are two types of descriptive modelling tasks: Clustering, and Association.

i. Clustering: The clustering data mining applications are descriptive (unsupervised) data mining tasks that used to search groups of inter-related observations so that observations that belong to the similar cluster which are equivalent to one another, than the observations that belong to other observations [140].

ii. Association: Association analysis is applied for the discovery of patterns which defined firmly related properties of the data. Implication rules or features subsets are used to typically designate discovered patterns [140].

3.6 Data Mining Techniques

Election prediction is a challenging task, as things got changed before voting nights. There are numerous ways to predict the election outcomes, someone is forecasting the election outcomes using social media like Twitter or Facebook data, but the main problems with such social media sites are the presence of fake ids. Some others are doing random sampling and predicts the elections outcomes but the main problems with such sampling is that it may produce biased results because it includes only small portion of survey or it includes only those people's views which are taking part in the survey and exclude the views of peoples which are not taking part in the survey.

In order to overcome such issue, the researcher is utilizing data mining and machine learning technologies to build an election prediction models which is purely based upon the parameters and these parameters are selected after consulting with domain experts.

To extract knowledge from election data-set different data mining techniques are utilized. The main aim of practicing data mining for political forecasting is not to make survey or exit poll methods obsolete. But to come up with new solutions or methods so as to ease the predictions process. There are several core data mining techniques available but the focus of this research is the application of classification techniques that would assist political forecaster to identify the attributes and predict the election outcomes constituency-wise for Jammu and Kashmir.

3.6.1 Decision Tree

A Decision Tree is a supervised machine algorithm applied to solve regression and classification problems. It is basically a tree like structure which is having root node on the top denoting a test on attribute, the root node correspondence to branch denoting results of that

attributes and finally it leads to leaf node that hold the class labels of that attributes [142]. The Decision trees algorithm follows a greedy (i.e. non-backtracking) approach and are built up or formed in a top-down recursive divide-and-conquer manner. It begins with training data set that have class labels associated with them. While building decision tree many branches show noise or outliers during training data and to overcome this problem tree pruning is used for removing anomalies from such branches, in order to improve classification accuracy on unseen data [140].

Various form of decision tree classifiers are available, the main distinction between them is the mathematical model based upon which the attribute splitting process done [143]. The most common measure for features or attributes selection are Gini index, Information Gain and Gain ratio [144].

The information gain is performed when the attributes are categorical in nature while Gini Index is performed when attributes are continuous in nature. The Information Gain attribute selection is a measure that expresses how well an attribute best partitions the tuples into distinct classes or groups. The working of a decision tree to build an election prediction model is shown in below-given Figure 3.6.1.

The principal objective of utilizing decision tree classifier for building an election prediction model is that it can predict the value or class of target variables; based upon decision tree rule gleaned during training phase of data. The primary aim of Information Gain approach is to selects the splitting attribute in such a way that it maximizing the Information Gain. The Information Gain for each attribute is calculated using *Equation (3.1)*

$$Gain(A) = \quad Info(D) - Info_A(D) \tag{3.1}$$

Where *Info (D)* is the average amount of information needed to find the class label of a tuple in *D* and is calculated using *Equation (3.2).*

*Info$_A$ (D)*means the expected information that is needed to classify a tuple from *D* based on the partitioning by *A and is* calculated in *Equation (3.3)*.

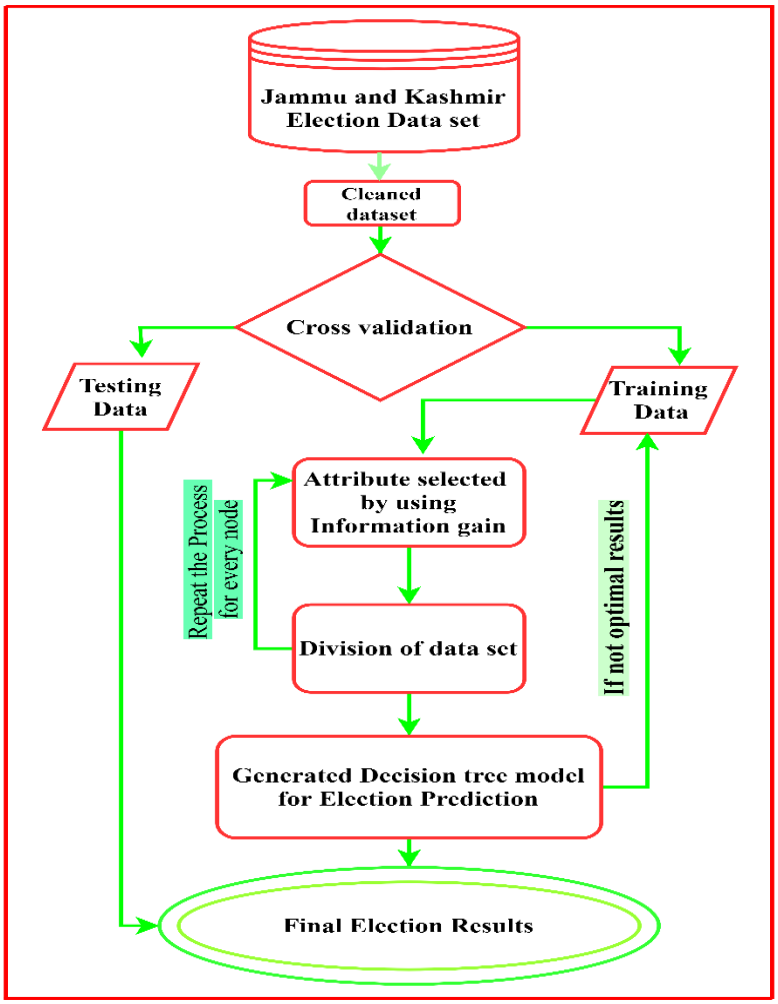

Figure 3.6.1: Decision Tree Model for Election Prediction

$$Info(D) = -\sum_{i=1}^{m} pi \ log_2 \ (pi) \qquad (3.2)$$

Where *pi* is non-zero probability that an arbitrary tuple in *D* belongs to class *Ci* and is estimated by ***|Ci, D| / |D|***. A *log* function to the *base2* is utilized as the information is encoded in bits.

$$Info_A(D) = -\sum_{j=1}^{v} \frac{|\ Dj\ |}{|\ D\ |} * Info(D_j) \qquad (3.3)$$

The term ***| D_j | /| D|*** acts as the weight of the j^{th} partition.

The experimental results obtained from the decision tree are explained in chapter IV.

3.6.2 K-Nearest Neighbors (K-NN)

K-Nearest Neighbors is most delicate classifier that saves all the available cases & classify new instances based on nearest majority neighbours (*for example distance function)* [145]. K-NN has been adopted in statistical estimation, pattern recognition and was already considered as a non-parametric tool [146]. K-NN assumes that identical classes exist in shut proximity i.e entities which are similar, exist together. In the name of K-NN algorithm the alphabet 'K' means the number of nearest neighbours for determining the class of an instance, as shown in Figure 3.6.2.

The K-NN algorithm falls under "Lazy learner" classifier as it saves the training dataset and waits until it is provided with test dataset, then it performs the required operation for classifying the dataset based on available majority neighbors [147].

The K-NN is considered to as a 'instance-based learner'. It perform lesson training tuples and perform well while working with classification and prediction, hence makes it computational expensive, unlike the eager learner that when given a training tuple has the capability of constructing an classification model before receiving the test tuples to classify, so K-NN remain ready and eager for classifying the unseen tuples[146].

The good value of 'k' can only be determined experimentally by setting the value of k to 1 and then increment 'k' to allow for new more neighbours. The 'k' value that gives the minimum error rate is selected. The test set is used to estimate the error rate of the classifier. While describing the class of new instance K-NN measure the distance function using Euclidean or Manhattan distance measure then classified the new instance based upon closeness to the neighbors [148]. K-NN have many distance measures that can be used, such as (Euclidean,

Manhattan and Minkowski) but in this research, Minkowski measure is used because of the properties of the election data.

The Euclidean distance between two points is the length of the path connecting them. It may be determined as the square root of the sum of the squared differences between i= (xi_1, xi_2 ...xi_p) and j= (xj_1, xj_2... xj_p) across all input attributes p.

$$\boldsymbol{d(i,j) = \sqrt{(xi1 - xj1)^2 + (xi2 - xj2)^2 + \cdots + (xip - xjp)^2}} \qquad (\mathbf{3.4})$$

Before using Euclidean distance measure the values of each attribute is normalized so as to prevent attributes with initially large ranges from outweighing attributes with initially smaller ranges.

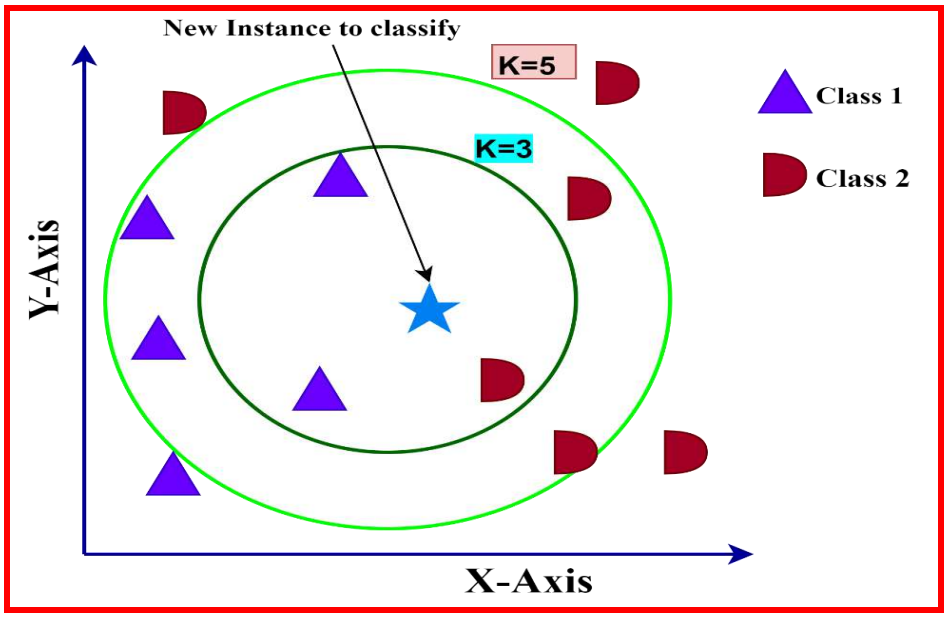

Figure 3.6.2: Example of K-Nearest Neighbor (KNN) classification drawn from [149]

The Min-Max normalization is used to transform a value **V** of numeric attribute **A** to **V** in the range [0, 1] by computing

$$V = \frac{V \ - \ min\, A}{max\, A \ - \ min\, A} \qquad (3.5)$$

Where $min_{\mathbf{A}}$ and $max_{\mathbf{A}}$ are the minimum and the maximum values of attribute A

In this research, the K-NN technique is used to predicts the election outcome and the experimental results obtained from the classifier are discussed in chapter IV.

3.6.3 Support Vector Machine (SVM)

Support vector machine is one of the most powerful classifiers that is being utilized for regression as well as classification rationales [150]. The main task of Support Vector Machine is to find the optimal hyper-plane that divide the data-set into different classes based upon their characteristics [151]. The purpose of SVM is to select the hyperplane that have maximum margin at-least between two hyperplanes in such a way these hyperplanes segregating various classes correctly. However, if there is no clear hyperplane it is necessary to move to a higher dimension view called as kernelling in SVM. The motive of utilizing the Kernel trick in SVM is that it can transform the input data into the appropriate form *for example* from one dimensional input data to two-dimensional output data. This process is continued into higher dimensions unless a hyperplane is obtained to segregate it into different classes [152]. The *optimal* hyperplane for an SVM means the one whose distance with each nearest class is maximum. SVM finds out this hyperplane by using margins and support vectors. It is to note that the support vectors are the data points that are nearest to the segregating hyperplane and are considered critical elements of the dataset, and the margin means the maximal width of the slab parallel to the hyperplane that has no interior data points. The discriminant function $f(T)$ is a linear combination of support vectors for a test sample T and is constructed as follows:

$$f(T) = \sum_{n=1}^{\infty} \alpha_i y_i (X_i \ . \ T) \ + b \qquad 3.6$$

Here the X_i are support vectors, Y_i are class labels of vectors X_i. (X_i.T) is the dot product of the test

sample T for one support vectors $X_{i.}$ $_{i}$ and b are parameters(numeric) to be determined by learning algorithm. The Figure 3.6.3 illustrates the linear support vector machine where brown-red circles represent data points of class X_1 and sky-blue, indicating data points of X_2. In case of non-linear separation, the training data will be mapped into a higher-dimensional space *H* and an optimal hyperplane will be constructed there. Different mappings construct different SVMs. When there is a mapping, the discriminant function is given like:

$$f(T) = \sum_{i=1}^{n} \partial_i y_i K(x_i, T) + b \qquad (3.7)$$

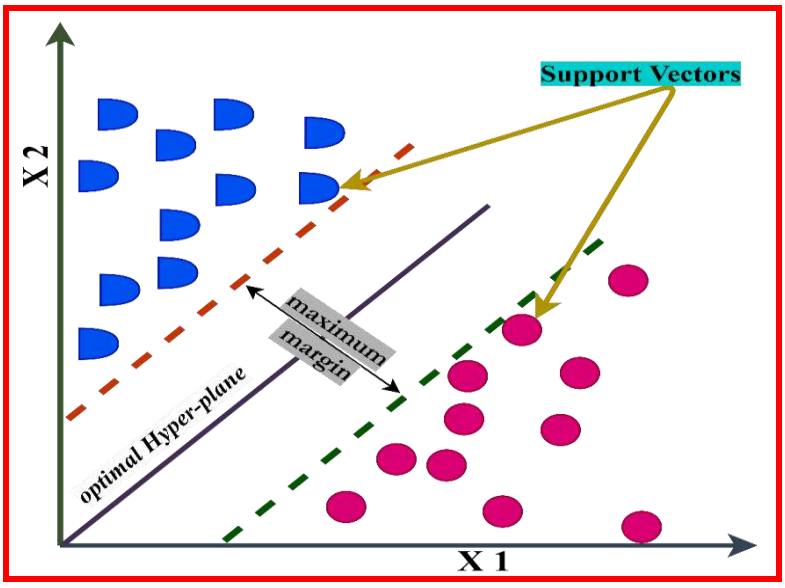

Figure 3.6.3: Linear SVM Classifier for Two class representation drawn from [153]

SVM is largely characterized by the choice of its kernel function used *for example* polynomial kernel and Gaussian radial basis kernel function. However, besides these kernel functions, there are other kernel functions. In order to determine parameters and y in the equation, the construction of the discriminant function finally turns out to be a constrained quadratic problem on maximizing the Lagrangian dual objective function:

$$maxW(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(X_i, X_j) \quad (3.8)$$

Under constraints

$$\sum_{i=1}^{n} \alpha_i y_i \; = \; 0, \; \alpha_i >= 0, (i = 1,2,\ldots,n) \quad (3.9)$$

Where *n* is the number of samples in training data. However, the quadratic programming problem in equation (3.8) cannot be solved easily via standard techniques. The SVM technique is used in election prediction because its accurate rules which are important to help poll forecasters. The election prediction results obtained from the SVM model are discussed in chapter IV.

3.6.4 Random Forest

Random Forests are an ensemble of large numbers of decision tree which can be used to predict the new data or class by aggregating the average majority vote for classification and regression [154]. It builds forest based on large set of decision trees that are randomly selected from the training dataset. The main aim is to overcome the over-fitting problem of each decision tree and also to handle meta dataset with higher dimensionality. In random forest classification task, individual decision tree votes and on the basis of summarized votes final class is decided however in regression the mean of every tree is calculated and finally majority votes become model predictions [155].Random forests (Breiman, 2001) are trained by bagging methods that builds a large collection of de-correlated trees and then averages them [156].

In random forest algorithm, each tree is grown on a different sample of original data and there is no need for cross-validation to get an unbiased estimate of the test set error because, in each of k iterations, about 1/3rd of the samples are left out of the new bootstrap training set and are

not used in the construction of the tree. In this way, a test set classification is obtained for each sample in about one-third of the constructed trees. Figure 3.6.4 shows the working of the random forest algorithm. The final class so obtained from the sample is the class that have highest majority votes associated with it. The random forest algorithm used in election prediction and forecasting. The election prediction results obtained from the random forest model are discussed in chapter IV.

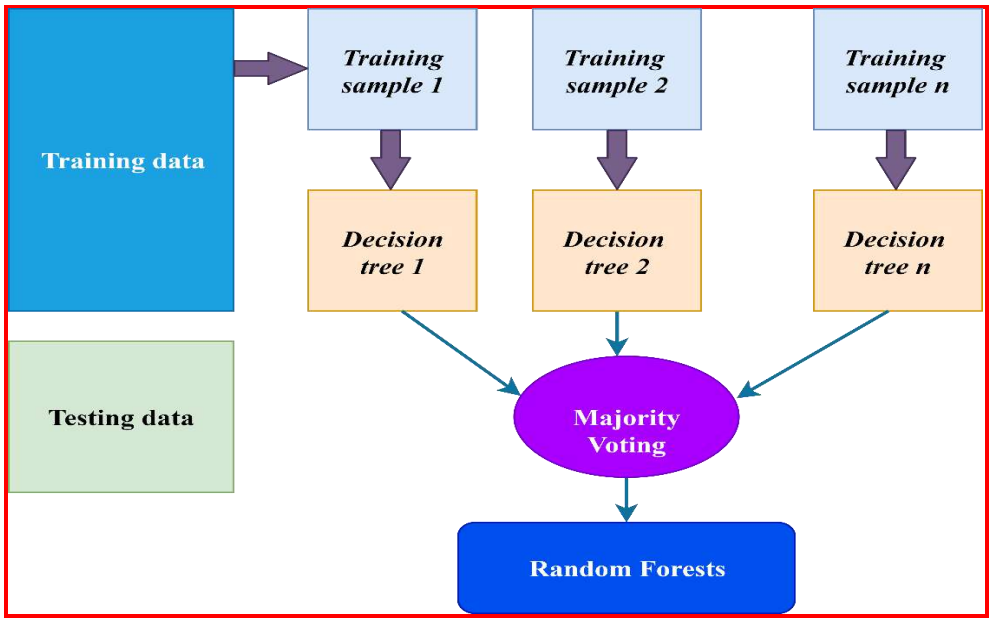

Figure 3.6.4: Random forest algorithm working drawn from [156]

3.7 Model Evaluation Techniques

To check the performance of the model one need systematic ways to evaluate how data mining techniques work. For classification task, it is important to measure the effectiveness of classifiers in term of the confusion matrix, cross-validation, error rate, sensitivity, specificity, accuracy, precision and ROC curves, which are discussed as follows:

3.7.1 Confusion Matrix

Confusion matrix is mainly used for measuring the performance of machine learning

classifiers on test data whose true values are already known [157]. Below given table 3.7.1 shows the two-class confusion matrix which provides insights into the types of errors being made by a classifier. The Positive tuples that are correctly labelled by the machine learner classifier are known as True Positive (TP). Whereas the negative tuple which are correctly labelled as negative by the classifier are known as True Negative (TP). False Positive (FP) are the negative tuples that are wrongly classified as Positive tuple. Whereas False Negative are the positive tuples but are misclassified by classifiers as positive tuple [159].

Table 3.7.1: Confusion matrix for two class classification drawn from [158]

		Predicted Cases	
		Negative	**Positive**
Actual Cases	**Negative**	True Negative	False Positive
	Positive	False Negative	True Positive

i. **Sensitivity:** when it is actual positive how many times the classifier recognised it as positive. It is also known as True Positive Rate/ Recognition/ Recall rate.

$$Sensitivity = \frac{TruePositive}{TruePositive + False\ Negative} \qquad (3.11)$$

ii. **Specificity:** when it is actually negative, how many times classifiers predict it as negative. it is also known as True Negative Rate.

$$Specificity = \frac{TrueNegative}{\text{TrueNegative} + FalsePositive} \qquad (3.12)$$

iii. **Accuracy:** is the overall percentage of cases that are correctly classified by an algorithm (i.e. overall how much classifier is correct).

$$Accuracy = \frac{True\ Positive\ +\ True\ Negative}{TP\ +\ FP\ +\ TN +\ FN} \qquad (3.13)$$

iv. **Precision:** when classifier predicts yes, how often it predicts correctly (i.e. upto how much the percentage of tuples labelled as positive are actually true).

$$\boldsymbol{Precision = \frac{True\ Positive}{TruePositive\ +\ FalsePositive}} \qquad (\mathbf{3.14})$$

3.7.2 AUROC (Area under the Receiver Operating Characteristics)

The AUROC curve is universally accepted methods for measuring the performance of machine learning classifiers over certain threshold in-between true positive and false positive errors rate [160]. ROC is a probability curve and AUC represent the degree or measure of separability [161]. It specifies the efficiency of the classifier in distinguishing between two classes in case of binary classifications as shown in the Figure 3.7.2 [162].

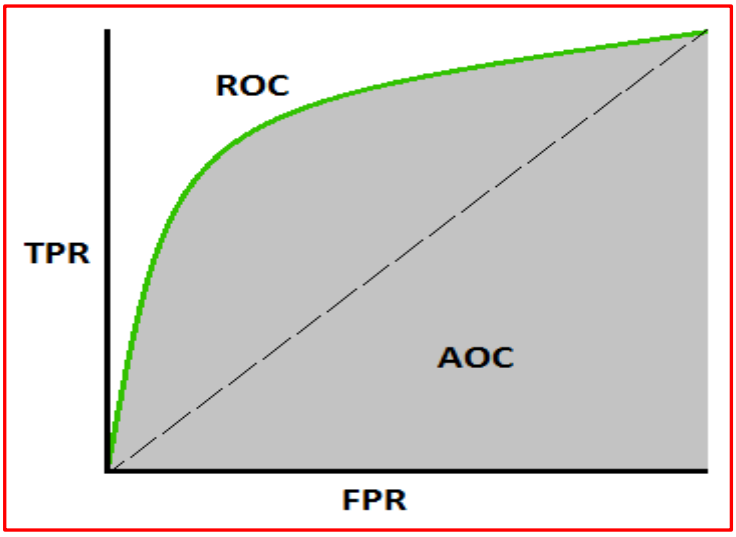

Figure 3.7.2: AUROC Curve Representation

True Positive Rate (TPR) and False Positive Rate (FPR) is used for plotting ROC Curve in such

a way; that TPR depicts on y-axis and FPR depicts on x-axis respectively. An intelligent models AUROC score is equal to 1, which means model distinguish class between positive and negative very well. When the model has AUC near to 0, it then reciprocating the results. In such situations models is predicting 0s as 1s and 1s as 0s further, when AUC of model is close to 0.5 then that model has no discrimination capacity in separating between positive class and negative class.

3.7.3 Cross-Validation

Cross-validation is basically a statistical technique that is used to measure the skill of classifiers in term of error rates on a particular dataset. The training dataset allows data mining techniques to learn from this data. The testing dataset is used to evaluate the performance of the data mining technique in relation to what is learned from the training dataset [163].

3.7.4 Misclassification Rate

The errors committed by a classification model are generally divided into two types: training errors and generalization errors. Training errors; also known as resubstitution error or apparent error, is the number of misclassification errors committed on training records, whereas generalization error is the expected error of the model on previously unseen records. A good classification model must have low training error as well as low generalization error. The classifier predicts the class of each instance: if it is correct, that is counted as a success; if not, it is an error. The error rate is the proportion of errors made over a whole set of instances and it measures the overall performance of the classifier [164] .

$$Error\ Rate\ = \frac{False\ Positive\ +\ False\ Negative}{Positive\ +\ Negative} \tag{3.15}$$

3.8 Ensemble Techniques

Ensemble methods merge numerous machine learning classifiers for classification into one predictive classifier in order to achieve better accuracy and less variance [165]. It is well known fact that ensemble methods generate more accurate outputs with higher accuracy then single classifier, and this has led the usage of ensemble methods in different fields of computer science.

It is of two types- *Hard voting* and *Soft Voting.*

Hard voting is where model is selected by majority vote. For example, if researcher have four classifiers with outputs 1,0,1,1 respectively, then researcher select the output as 1 because of majority rule [166].

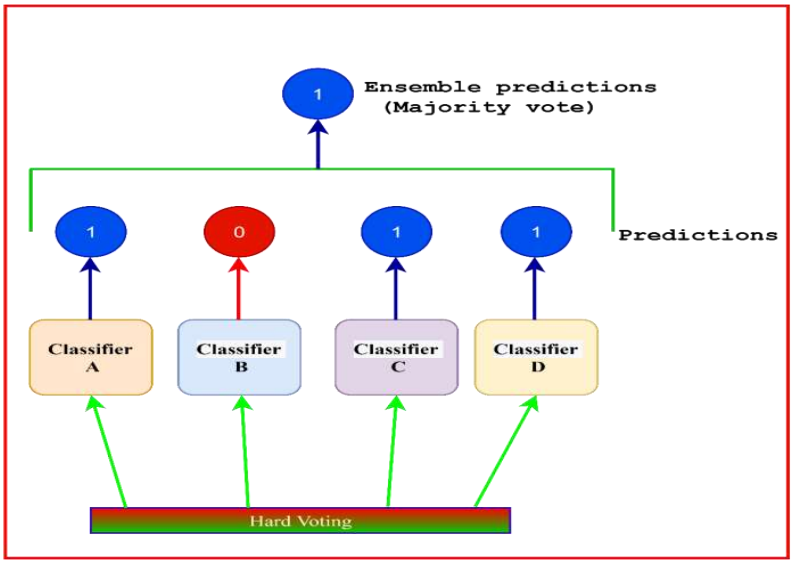

Figure 3.8.1: Hard voting drawn from [167]

The Figure 3.8.1 explained working of hard voting by which researcher can compute a weighted majority vote by associating a weight wj with classifier Cj:

$$y\hat{} = \text{argmaxi} \sum_{j=1}^{n} \text{mwj}\chi A(\text{Cj}(x) = i) \qquad (3.6.1)$$

where χA is the characteristic function [Cj(x)=i∈A], and A is the set of unique class labels.

The description of Figure 3.8.1 is explained as below

Classifier A predict outputs 1

Classifier B predict outputs 0

Classifier C predict outputs 1

Classifier D predict outputs 1

As we noticed that 3/4 classifiers predict the 1 as output, so 1 is hard voting or ensemble decision.

Soft Voting is done by calculating the mean probabilities of all the classifiers used for outcomes [168]. In soft voting, the researcher predicts the class labels based on the predicted probabilities p for classifiers as shown in the Figure 3.8.2

$$y\hat{} = \text{argmaxi} \sum_{j=1}^{n} \text{mwjpij} \qquad (3.6.2)$$

where w_j is the weight that can be assigned to the jth classifier.

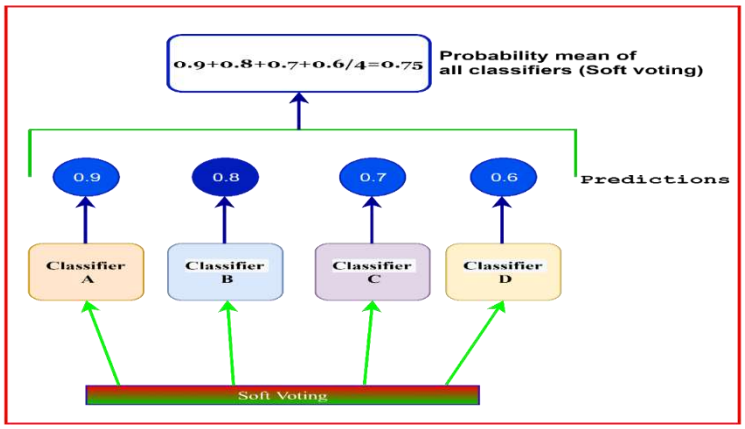

Figure 3.8.2: Soft voting drawn from [169]

In soft voting average probabilities mean of all classifiers taken into consideration for making final outputs. In soft voting the outputs of all classifiers are dividends between class1, and class 0, then the class that have highest outputs is considered as final class as shown below in Figure

3.8.3. Thus soft-voting classifier performs better than hard-voting as it gives more weight to highly confident votes.

In the Figure 3.8.3 there are four classifiers with two class (i.e class1 and class0) and after calculating the probability mean of two class and it was noticed that class 1 have highest probability mean then the class 0, thus in this research work the researcher choose class1 as the final class for making final outputs. The election prediction results obtained from the ensemble learning are discussed in chapter V.

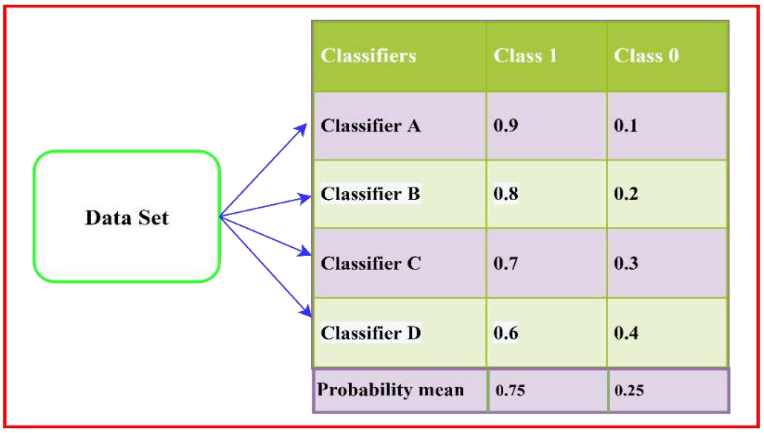

Figure 3.8.3: Soft Voting Classifiers Working

3.9 Statistical Test

The statistical tests are performed to find the either the model is significant or not using p value, such that if $p<0.05$ then we reject the null hypothesis and this prove that the comparing models are performing different, and the model upon which comparison made is significant. In this research work the researcher are performing statistical T-Paired test.

*T-paired test: "*A popular test used to compare two learning algorithms is the paired-difference t-test, this test uses Student's t distribution to estimate a p-value representing the probability that the mean of the differences observed occurred randomly. If the resulting p-value is very low (usually below 0.05), it can be concluded that the observed difference is more than can be

explained by random chance, and is therefore statistically significant. In the context of machine learning, this amounts to comparing how well algorithm A does compared to algorithm B on a particular learning problem characterized by a data set. If the mean difference between algorithm A's performance and algorithm B's performance using data set is statistically significant, there is support to prefer using algorithm A for that particular learning problem. For this reason, tests comparing two or more algorithms are important in validating the utility of machine learning algorithms and comparing them to each other in different problem domains [170]. A paired t-test is used for measuring the observation means of two models in term of samples using p value.

3.10 Data Collection and Research Methods for Election Prediction Model Development

The Considerable challenge to collect a dataset is to get the quality and relevant data. The Primary data is obtained from different heterogeneous data sources like election commission of India and the election commission of Jammu and Kashmir websites and some data from every district election cell. This election dataset comprises 3770 records accompanied by twelve attributes, however the attributes which are used for model building are shown in table 3.10.

In this research work Descriptive research methodology is followed to develop political forecasting model. The election prediction model is developed in python programming language and data cleaning, statistical modelling, data visualization, and machine learning operations are done in Jupyter web application.

The Exploratory Data Analysis (EDA) is performed on the Jammu and Kashmir election dataset to get different insights from the data. After EDA it is found that the election dataset is noisy and consist of several missing attribute values represented with interrogation mark (?).

Data imputation is performed (for *numeric* missing values *mean imputation* data cleaning technique is used to fill in the missing attribute values and for the *categorical attributes, Mode* method for filling in the missing values is used).

The attributes of election dataset are organized into two types nominal and numeric. For example, the Gender attribute values as Male and Female represent nominal attribute and the Age attribute values as 50 years represent a numeric attribute. Furthermore, the nominal data attributes correspond to binary and ordinal variables and continuous data attributes correspond to integer, interval-scaled and ratio-scaled variables.

Table 3.10: Parameters and their Description

Parameters	**Descriptions**
Central Government Influence	The Party which formed the government in India has the maximum chance of forming the government in the Jammu and Kashmir. Like in 2008 Congress formed coalition govt. with Jammu and Kashmir National conference (JKN) in Jammu and Kashmir also, 2014 BJP formed coalition govt. with Jammu and Kashmir People Democratic Party in J&K.
Religion Followers	The religion factor dominated the politics of Jammu and Kashmir. Religion is the primary reason for BJP as it could not win even single seats in Kashmir province from three last elections because the majority of religion followers are Muslim.
Party Wave	It means the Party which gained positive waves at election times has the maximum chance of winning enormous numbers of seats in elections like BJP Party wave during 2014 India Parliamentary elections.
Party Abbreviations	It specifies the name and type of parties like INC (Indian National Congress), JKN (Jammu and Kashmir National

	Conference) and central Party, state party or independent. If a candidate is contesting the election from a party than that candidate has maximum chance of winning election in contrast to candidates who contested election as an Independent candidate.
Sensitive Areas	This phenomenon is very prevalent in Kashmir province especially at election times, peoples boycott the poll and do not cast their votes, which affects the election process in J&K.
Vote Bank	Vote bank means either people vote for the Party or candidates.
Hereditary	Hereditary implies if a person's contest elections and some-body had already contested election from his/her family or his/her heir, then the winning chance of that person is more as compared to his opponent's candidates. Like Abdullah family, Mufti family etc.
Incumbent Party	It employs the winning Party of every constituency. The sitting Party may not have a bright chance of winning election repeatedly because of voter inclination.
Caste Factor	Caste plays a predominant role in some areas of Jammu and Kashmir like Twin district of Poonch, Rajouri, Kupwara, Gool Arnas and whole of Ladakh region. Peoples voted for their caste candidates instead of party or religion.

3.11 Summary

This research investigates developing an election prediction model using data mining analysis. The primary data is collected from different heterogeneous data sources like election commission of India and Election commission of Jammu and Kashmir websites through quantitative data collection methods while some data of it is collected through election cell of each district. This research work follows the descriptive research methodology and used python programming language and Jupyter web application to develop the election prediction model.

In this research work different feature selection techniques are applied over the Jammu and Kashmir election dataset to select the significant subset of features for the early prediction of election outcomes. This investigation applies different data mining techniques like Decision Tree, K-NN, SVM and Random Forest and finally ensemble them into one classifier to see whether these techniques will help poll forecaster in early predictions of election results. Various model evaluation techniques and statistical methods techniques are used to measure the performance of the developed election prediction model. Finally, the chapter is concluded by discussing the significance of election prediction attributes for Jammu and Kashmir.

CHAPTER 4

4. Proposed Methodology

In, this chapter the researcher discusses the knowledge Discovery in Data mining (KDD) methodology. The researcher used the proposed data mining methodology to build an efficient election prediction model for the early prediction of election outcomes before actual results announced. The Jammu and Kashmir dataset is mined to derive knowledge for the early prediction of election outcomes. In this chapter, the research design is formulated to simplify the research activities and make the research productive by the statement of objectives. In this chapter the researcher also proposed the research methodology, which will be utilized in the different systematic research phases and steps. This study is exploratory that attempts to explore the main components, barriers, issues and requirements to build an election prediction model so as to help political forecasters for the accurate predictions. Finally, an election prediction model is developed that helps the user in identifying the winning or losing status of any political parties constituency-wise for Jammu and Kashmir. The model is built in such an efficient design that naïve as well as professional user can handle and understand it easily.

4.1 Data Mining Methodology for Election Prediction

A data mining methodology is a technique for applying alternative methods to take raw data and transformed it into an understandable format so as to generate knowledge for users. There are two eminent prevailing data mining methodologies for the "Knowledge Discovery from Data" process: CRISP-DM and SEMMA[171]. "The CRISP-DM (Cross-Industry Standard Process Model for Data Mining) was developed by an industry-led consortium in the year 1996 [172]. CRISP-DM is defined as a process model which provides a frame work for carrying out

data mining projects which is independent of both industry sector and technology used [173]. The SEMMA (Sample Explore Modify Model Assess) is a data mining methodology derived from the Statistical Analysis Software Institute (SAS, 2008) [171].” These two methodologies are not advisable for this research work because they are too hefty and too complicated to use. Hence for this research work, KDD methodology is followed, as shown in Figure 4.1. The reason to apply this specific methodology is that it demonstrates our research objectives precisely. This methodology contains the following steps:

Step I. Data Selection: In this phase, the relevant election data from various heterogeneous sources is selected and then stored in the standard database.

Step II. Data Preparation: In this phase election dataset is analyzed and prepared into an appropriate form for the data mining algorithms to derive meaningful insights from it and to get the optimal output.

Step III. Data Task Filter: In this step, the heuristic decision rules are employed to determine expected results for election prediction in later steps. The selected dataset is then stored in the “Data Mining Task Warehouse.”

Step IV. Data Mining Techniques: In this step, an appropriate algorithm is selected with a suitable dataset for the task requested in step3.

Step V. Comparison and Evaluation: In this phase, the classified outcomes are contrasted and estimated based on different data mining evaluation measures.

Step VI. Building New Models: In this phase, the developed supervised classification models are stored in the data mine warehouse for the next prediction problems. For new prediction tasks, the process is repeated from step 3 to step5.

Hence it is clear that data mining is becoming more popular day by day because of its spacious

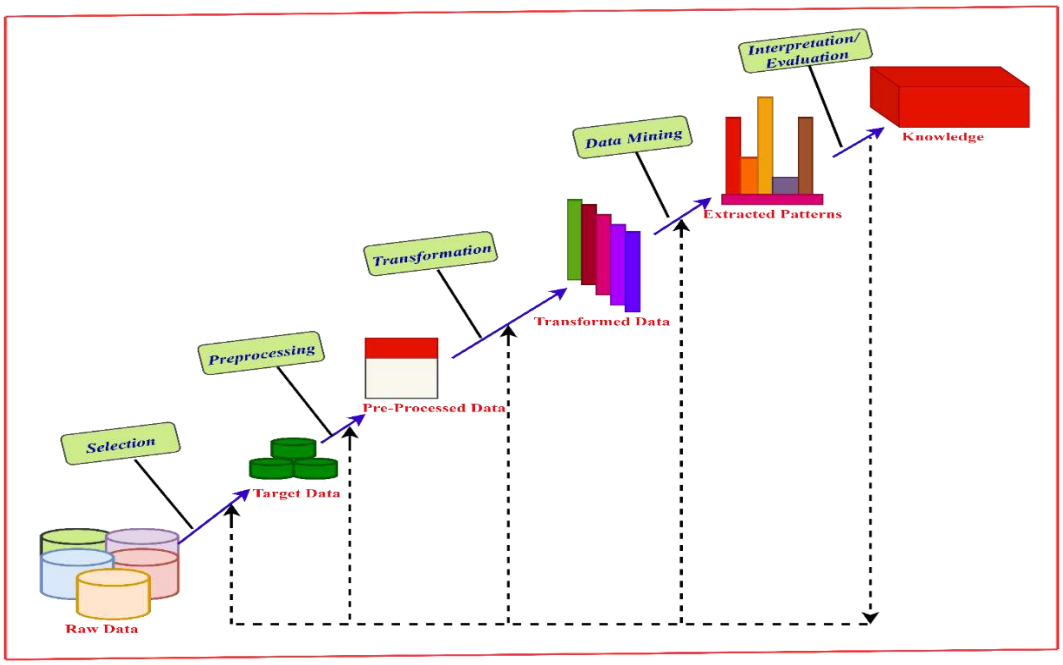

Figure 4.1: Election Prediction Model Methodology

application that enables the researcher and information industry to eliminate the randomness and discover the hidden patterns from large database. Because of its widespread application in every aspect of human life. Data mining is anticipated to be the utmost revolutionary advancement for next decade.

4.2 Research Design for Election Prediction Model

The research design is followed to simplify the research activities and make research productive by the statement of objectives. It provides information about the data inputs required, the methods of analysis used, and a statement of objectives of the study to solve the research problem. The proposed research design is followed step by step to build the election prediction model for Jammu and Kashmir (constituency wise) as shown in below given Figure 4.2. The research design consists of three main phases having eight steps. The phases of the research design with their respective steps are explained as follows:

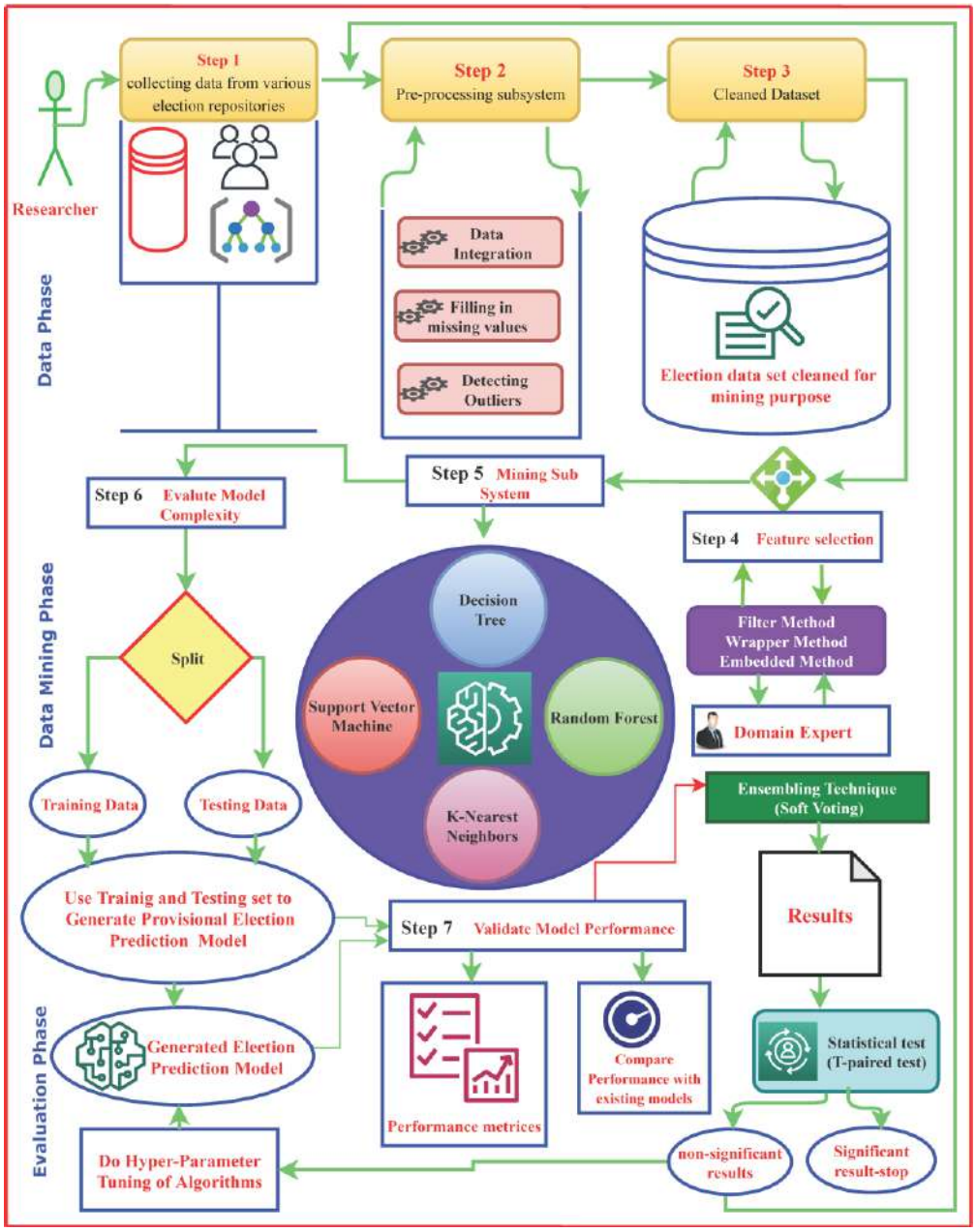

Figure 4.2: Research Design for Election Prediction

Step I. Data Phase: The data phase contains the whole process from data collection to feature engineering. This phase includes the qualitative data collection step, the pre-processing subsystem step, the cleaned data set storage step, and finally, the feature selection step.

Step II. Data Mining Phase: The data mining phase includes Machine Learning classifiers like K Nearest Neighbor, Decision Tree, Support Vector Machine, Random Forest, and finally ensemble them into one classifier which would be employed to develop the election prediction model.

Step III. Model Evaluation and Validation Phase: The model evaluation and validation phase calculate and endorse the evaluation model efficiency using different data mining techniques. The election prediction model would be evaluated through the train-test split of the election data using 10-fold cross-validation. The model would be then validated using different performance measures by comparing the results with the existing models and various model metrics (Sensitivity, Specificity, Accuracy, Misclassification Rate, and ROC Score). Then finally, a statistical test like t-paired test will be performed to check the level of significance.

Step IV. Knowledge-Base Phase: The Knowledge-based phase includes the steps to store and retrieve knowledge about election prediction. The generated election prediction model rules would be stored in the knowledge base and cross-checked as per domain expertise for poll predictions. In this chapter first step (qualitative data collection, pre-processing subsystem, and feature selection) is discussed and the rest of the steps will be discussed as demanded by the research for the development of the election prediction model.

4.3 Exploratory Data Analysis (EDA) Process

The basic statistical description is performed to learn about each attribute value of the Jammu and Kashmir election dataset. Knowing such fundamental statistics about each attribute helps to smooth noisy values, spot outliers, and fill in the missing values. The election prediction data consists of a combination of nominal and numeric attributes. The missing numeric values are

removed through the simple mean imputation method, and categorical missing values are filled by mode imputation technique.

4.3.1 Checking Class Imbalance and Data Distribution Problems in Dataset

Before performing any operations on the election dataset, it is required to test class balance because highly imbalanced data makes the machine learning algorithms biased.

The dataset contains 3776 records, of different political parties which has contested election in Jammu and Kashmir from the year 2002 to 2014. So, keeping in views of assembly election results from 2002 election to 2014 state assembly election the researcher is building the election prediction model for J&K so that it predicts accurate outputs instead of biased outputs. The researcher collected the data from the election commission of India and election commission of Jammu and Kashmir websites for the past three terms of data spanning from 2002 to 2014 results. Some dataset is collected as field survey for the different constituency of Jammu and Kashmir. The dataset was collected with various features includes State, constituency, Account No, Gender, Caste, Party wave, Religion followers, Caste factor, Hereditary, Vote-bank, Central Government Influence, Party Abbreviations, Sensitive areas, Votes polled, Votes majority and finally Party Won. The dataset that is obtained from the https://eco.Gov.in.

The researcher utilized the election dataset in this research work from 2002 assembly election to 2014 assembly election because before 2002 assembly election in J&K, Jammu and Kashmir National Conference (JKN) won maximum number of assembly election in J&K [174] and hence it leads the model may produce biased or vague results. But after 2002 more party like Jammu & Kashmir Peoples Democratic Party (JKPDP) and Bharatiya Janata Party (BJP) started forming governments in Jammu and Kashmir along with Jammu and Kashmir National Conference (JKN) and Indian National Congress (INC) as shown in table 4.3.1

Table 4.3.1: Performance of the main political parties from 2002 to 2014.

S.No.	Years	Name of Party	Seats won	Governments formed
1	2002	JKN	28	INC+ JKPDP
2	2002	JKPDP	16	
3	2002	INC	20	
4	2002	BJP	01	
5	2002	Other	22	
6	2008	JKN	28	INC+ JKN
7	2008	JKPDP	21	
8	2008	INC	17	
9	2008	BJP	11	
10	2008	Other	10	
11	2014	JKN	15	JKPDP+BJP
12	2014	JKPDP	28	
13	2014	INC	12	
14	2014	BJP	25	
15	2014	Other	07	

CHAPTER 5

5. Implementation and Results

In, this chapter the researcher discusses the results and implementation of election prediction model. Researcher used the Pearson coefficient correlation for finding the relationship between different election predictions attributes. Then stimulate the results of applied attributes selection techniques which consists of filter method, wrapper method and embedded methods to develop an intelligent election prediction model for early prediction of election outcomes before actual results are announced.

5.1 Finding Correlation among Different Election Prediction Attributes

In any dataset, there could be multifaceted and strange relationships among the attributes; hence, it is imperative to determine and measure the degree to which attributes in the dataset are related to each other. This process of finding the degree of relationship amongst the dataset attributes is known as correlation. The knowledge of the correlation between the attributes helps to prepare the data so as to meet the expectations of machine learning algorithms. Pearson's correlation is applied to check the mutual relationship among the election prediction attributes. A correlation could be positive (which means all the related attributes move in the same direction), negative (which means all the related attributes move in opposite directions) or neutral (which means that the attributes are not related to each other). The result of the applied Pearson's correlation coefficients among the election prediction variables is shown in the below-given Figure 5.1 in the form of heatmap representation. The heatmap grid represents the correlation between the election attributes with their corresponding coefficients. The

Symmetrical heatmap matrix represents all attributes across the top and down the side, to give a correlation between all pairs of features.

Figure 5.1: Correlation in Election Prediction Attributes Through Heatmap Representation

The diagonal line across the matrix from bottom-right corner to the top-left represents a perfect correlation of each attribute with itself. The value 1 represents perfect positive correlation among the attributes and value -1 describes perfect negative correlation among the election prediction attributes; a correlation coefficient close to zero indicates weak dependency among the election prediction attributes.

After analyzing the heatmap correlation results, it is found that the independent attributes from Jammu and Kashmir election dataset are strongly correlated with one another. However, if the

attributes in a dataset are firmly correlated then change in one variable can lead to change to another. Correlation among the attributes does not mean causation hence, the strong relationship among attributes should be evaluated significantly. Mostly, a relationship among attributes may look causal through strong correlation because of some overlooked factors.

5.2 Feature Selection Techniques for Election Prediction

The feature selection techniques are applied to choose the significant and most appropriate subset of attributes for accurate prediction of election outcomes at constituency level. Feature selection helps to reduce an inappropriate and redundant attribute, which often decrease the performance of classifiers. In this research work, filter, wrapper, and embedded feature selection techniques are applied so as to get an appropriate feature subset for election prediction model. These feature selection techniques weight each attribute as per their role in election prediction. The applied feature selection techniques provide weight in between the scale of 0 to 1 for every attribute of election prediction. After weight assignment to every attribute by the separate feature selection technique, the overall mean of all the applied weights to every attribute by these methods is taken as the final weight. The feature with the mean value near to 1 is considered imperative in predicting election outcomes, and those attributes whose associated values are near to 0 are considered less significant in predicting election results for Jammu and Kashmir constituency wise.

The table 5.2.1 shows the different election prediction attributes with their respective weights assigned by different feature selection techniques and the last column in table shows the overall Mean of all the techniques. These election prediction attributes were identified by poll analytics 1. *Ranjeet Singh Karla (Assistant Professor)*, 2. Mr. *Assad Noomani* (Social Activists), 3.

Zulifkar Malik (Advocate and Social Activists) and many other general experts who are working in the Poll prediction at various level across, INDIA.

Table 5.2.1: Feature Selection Techniques Providing Weight to Each Attribute

	Filter_Method	Embedded_Method	Wrapper_Method	MEAN
PARTY_ABBREVIATIONS	0.72	0.50	0.50	0.57
INCUMBENT PARTY	0.23	0.77	0.88	0.63
PARTY WAVE	0.98	0.83	1.00	0.94
VOTE BANK	0.00	0.75	0.62	0.46
SENSTIVE AREA	1.00	0.00	0.25	0.42
RELIGON FOLLOWERS	0.99	0.84	0.12	0.65
CASTE FACTOR	0.39	0.84	0.38	0.54
CT_GOVT_INFLUENCE	0.69	1.00	0.75	0.81
HEREDITARY	0.36	0.86	0.00	0.41

The assigned weights to each election prediction attribute are validated and approved by different domain experts like *Dr. Afroz Alam (HOD Political Science Manuu Hyderabad India)*, etc. meanwhile these poll prediction domain experts gave their respective opinions to include some vital attributes also like (development agenda and gender factor) for early election prediction.

After analyzing the results, it is derived that the attributes [Party wave, Religion Followers, Incumbent Party, Vote Bank, Central Government influence, Hereditary, Party Abbreviations, Sensitive Areas and Caste Factor as shown in table 5.2.1] are the most important features for the early prediction of the election outcomes because their numeric

Figure 5.2: Election Prediction Attribute Hierarchy by Feature Selection Techniques.

values (corresponding) are high and are also validated and approved by different poll predictions domain experts. The pictorial representation of attribute hierarchy with their respective weights is shown in the Figure 5.2. The attributes with the highest weight are crucial, and the attributes with lower values are less significant in predicting election results.

The highly weighted significant subset from selected attributes are used to develop the election prediction model as shown in table 5.2.2.

Table 5.2.2: Mean Ranking of Election Prediction Attributes by Feature Selection Techniques

	Feature	Mean Ranking
1	PARTY WAVE	0.94
2	CT_GOVT_INFLUENCE	0.81
3	RELIGON FOLLOWERS	0.65
4	INCUMBENT PARTY	0.63
5	PARTY_ABBREVIATIONS	0.57
6	CASTE FACTOR	0.54
7	VOTE BANK	0.46
8	SENSTIVE AREA	0.42
9	HEREDITARY	0.41

5.3 Experimental Results of the Proposed Data Mining Techniques

It is found that the prevailing election prediction models are not free from limitations because they show varying results across different datasets that greatly reduces the effectiveness of the system. In this research, the election dataset of Jammu and Kashmir is mined through various classification algorithms like Decision tree classifier, Random forest classifiers, K-Nearest Neighbor classifiers with support vector machine classifiers and finally ensemble all of the classifiers into one classifier using soft voting technique. Various prediction domain performance metrics including sensitivity, specificity, accuracy, precision, AUROC score, and misclassification rates, also the model measures like statistical techniques are calculated to check the level of significance. Below given sub-sections explain the experimental results obtained by different election prediction models.

5.3.1 Decision Tree Model Experimental Results

The rationale to apply a decision tree is to develop an election prediction model that can predict election outcomes by learning decision rules deduced from training dataset. The performance results of the decision tree model are shown in the confusion matrix Figure 5.3.1.1 From the decision tree model's confusion matrix (Figure 3.7.1) the sensitivity, specificity, accuracy, precision, and misclassification rates are derived that are described as follows. The percentage of Political Parties and Independent candidates that were recognized accurately to have won the election (i.e. True Positive) upon the total number of Political Parties and Independent candidates who actually have the won the election is known as Sensitivity.

Putting the derived sensitivity values of the confusion matrix as shown below Figure 5.3.1.1 in equation (3.11) the sensitivity of 92% is obtained.

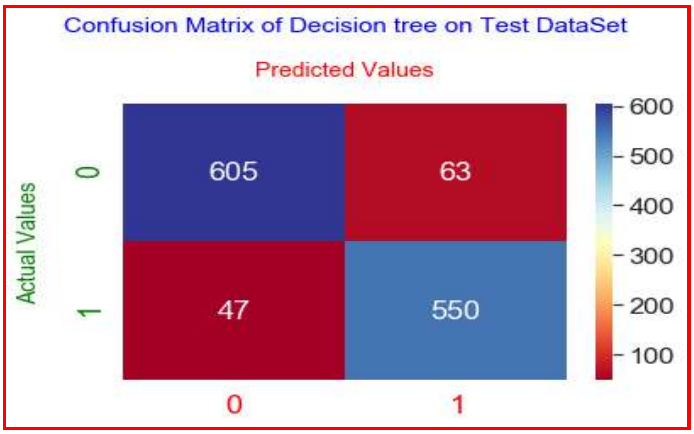

Figure 5.3.1.1: Decision Tree Model Confusion Matrix

The closer the value for this measure is to 1, the better the rules are at identifying those political parties or independent candidates who have won the elections. Similarly, the percentage of Political Parties and Independent Candidate that were recognized correctly to had not won the elections (i.e., True Negative) upon the total number of Political Parties and Independent Candidate who do not have won the elections is known as Specificity." Putting the derived specificity values of confusion matrix from Figure 5.3.1.1 in equation (3.12) the specificity of 90% is obtained which means the decision tree model can recognize the Political parties or independent candidates who don't win the elections with an accuracy of 90%.

The nearer the value for this measure is to 1 the best the rules are at identifying those Political parties or independent candidates without much errors. The overall accuracy of the decision tree model is obtained by using the equation (3.13) in Figure 5.3.1.1which is equivalent to 0.91% which represents that the decision tree election prediction model's overall performance in predicting both the won and lost election is 91%. The higher the accuracy percentage, the more the accuracy of the model. Similarly, putting the values of the confusion matrix from Figure 5.3.1.1 in equation (3.14), a precision of 0.89% is obtained. The closer the value for this measurement is to 1, the greater the chance that those with a positive outcome will actually have won the elections. If a high precision rate of the decision tree model is obtained, then it

means that the model will obtain a low false-positive rate. The error rate of the developed decision tree model is obtained by putting the values of the confusion matrix Figure 5.3.1.1in equation (3.15), which is equivalent to 0.08%. The lower the percentage of misclassification rate of the model, the more accurate the model is in identifying the winning or losing election for each constituency of Jammu and Kashmir. The AUROC performance measurement is used to check the "probability curve and measure of separability obtained by decision tree algorithm." AUROC demonstrates how efficiently the model can differentiate among the winning and losing the election by political parties or independent candidates' constituency wise. Below given Figure 5.3.1.2 is the AUROC of the decision tree algorithm with an AUROC score of =0.91%. "The area under a correlation curve plotting true positive against false positive is higher for models best able to correctly identify positive and negative cases.

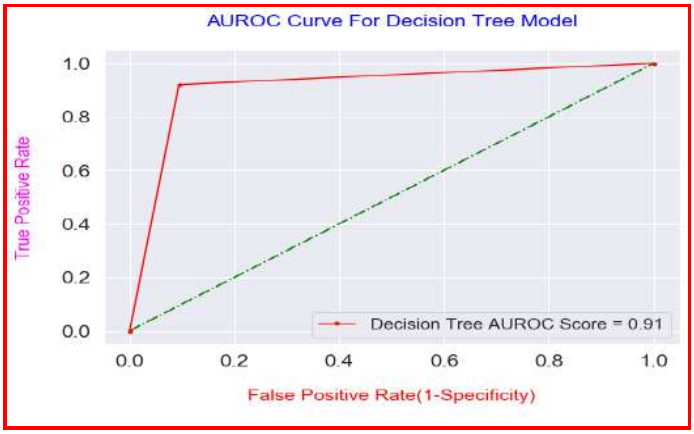

Figure 5.3.1.2: AUROC of the Decision Tree Model.

The researcher simulates the accomplished experimental results of the developed decision tree election prediction model with the prevailing research; the results obtained are greater than the published results.

5.3.2 K-Nearest Neighbor Model Experimental Results

The rationale to apply the K Nearest Neighbor algorithm is to develop an election prediction

model to predict election outcomes at its earliest. The 10-fold cross-validation is used on training dataset to get optimal and unbiased results. The performance results like sensitivity, specificity, accuracy, precision, and the misclassification rates of the K-NN classifier are derived from the confusion matrix Figure 5.3.2.1. The sensitivity of 0.85% is obtained by using equation (3.11) in Figure 5.3.2.1, which means the K-Nearest Neighbor model can recognize the winning Political parties and Independent Candidates with an accuracy of 85%. Similarly, the amount of election lost by different Political parties and Independent Candidates that were correctly recognized as defeated is 0.86% by using the specificity equation (3.12) in Figure 5.3.2.1; this means the K Nearest Neighbor model can recognize the loosing of election with an accuracy of 86%.

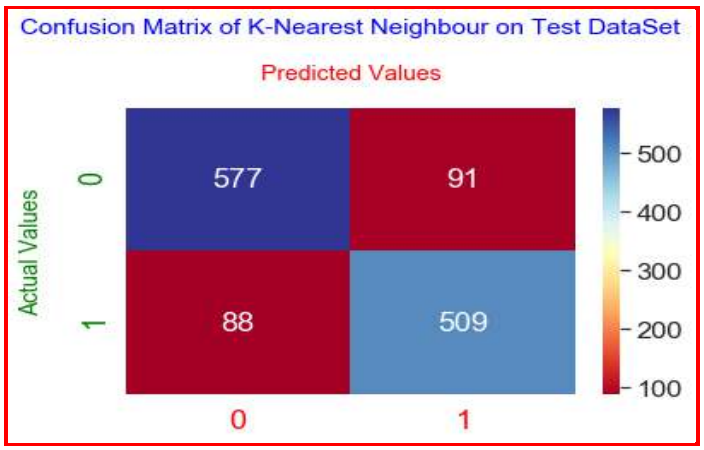

Figure 5.3.2.1: K-Nearest Neighbor Confusion Matrix on the Test Data

The overall accuracy of the K-Nearest Neighbor model is obtained by putting the confusion matrix Figure 5.3.2.1into the equation (3.13) which is equivalent to 0.85% this means that the K-Nearest Neighbor election prediction model overall performance accuracy in determining both winning and losing cases is 85%. Similarly, using equation (3.14) precision of the K Nearest Neighbor model is calculated that is equivalent to 0.84%.

If a high precision rate of the K-Nearest Neighbor model is obtained, it means the model will

obtain the low false-positive rate. The error rate of the developed K Nearest Neighbor model is obtained using the equation (3.15) in the confusion matrix Figure 5.3.2.1, which is equivalent to 0.14%.

The AUROC performance measurement is used to see whether the predictive classification model can accurately differentiate between the won and lost cases; however, the poor models will have difficulties in distinguishing among them. Below given Figure 5.3.2.2 shows the AUROC curve obtained from the K-Nearest Neighbor algorithm with an AUROC score of 0.86%.

The researcher simulates the accomplished experimental results of the developed K-Nearest Neighbor election prediction model with the prevailing research; results of which shows that the K-Nearest Neighbor model is optimal for election prediction because the misclassification rate is low.

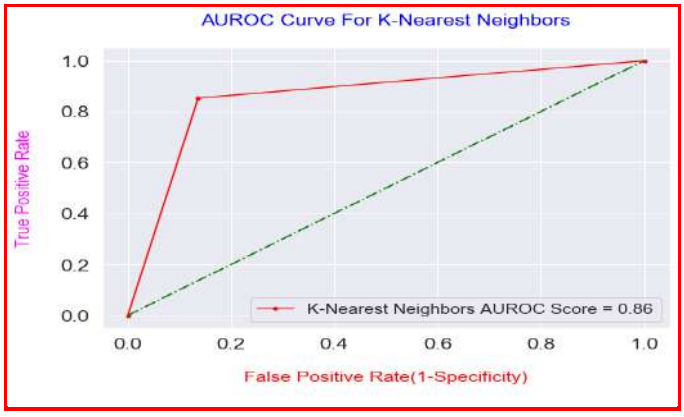

Figure 5.3.2.2: AUROC of K-Nearest Neighbor Model.

5.3.3 Support Vector Machine Model Experimental Results

Support Vector Machine separates data into classes based on a hyperplane with maximum margin and searches for the optimal hyperplane between classes to create the support vectors [175],[176].In this research, the Support Vector Machine is used to develop an election

prediction model that can predict election outcomes in its early stages. The performance results of the Support Vector Machine model on the Jammu and Kashmir election dataset are shown in confusion matrix (Figure 5.3.3.1) and the sensitivity, specificity, accuracy, precision and misclassification rates are derived from it which is described as follows:

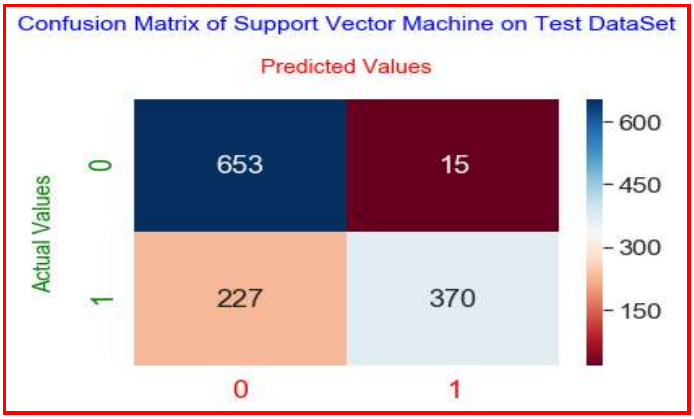

Figure 5.3.3.1: SVM Confusion Matrix on Test Dataset

Using equation (3.11) in confusion matrix Figure 5.3.3.1 the sensitivity of the Support Vector Machine (SVM) model is calculated as 0.61%; hence the SVM model can recognize the Winning cases with an accuracy of 61%. Similarly, by using equation (3.12) in Figure 5.3.3.1 the specificity of 0.97% is obtained, which means the SVM model can recognize the losing parties or independent candidates' cases with an accuracy of 97%. Similarly, the overall accuracy of 0.80%, is obtained by putting the values of the Figure 5.3.3.1 in equation (3.13), this means that the SVM election prediction model's overall performance in diagnosing both the winning and losing the Assembly elections at constituency wise is 80%. Similarly, the precision of 0.96% is obtained from Figure 5.3.3.1 using equation (3.14). The misclassification rate of the developed SVM model is obtained using the equation (3.15) in Figure 5.3.3.1, which is equivalent to 0.19%. The AUROC performance measurement is used to check the probability curve and measure of separability obtained by the SVM model. Below given Figure 5.3.3.2 depicts AUROC derived from the SVM model with an AUROC score of =0.80%.

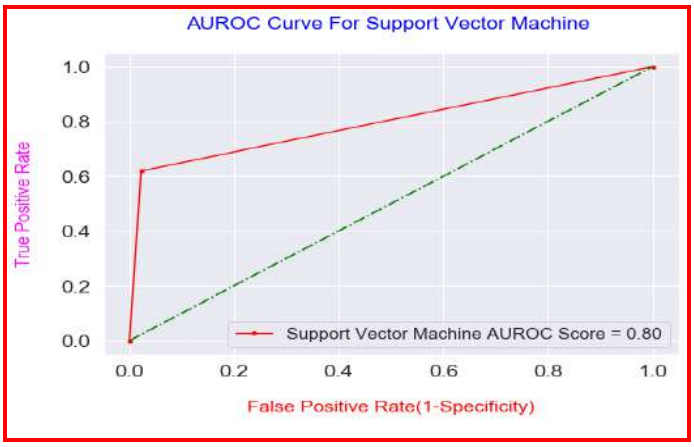

Figure 5.3.3.2: AUROC of Support Vector Machine Model

The researcher simulates the accomplished experimental results of the developed Support Vector Machine election prediction model with the prevailing research; the results obtained are to best of our knowledge. Hence the developed SVM model can be used for its practical implementation. However, further improvements are required in the SVM model which the researcher will do in chapter v using hyperparameter tuning.

5.3.4 Random Forest Model Experimental Results

The predictive result of Random Forests model on the Jammu and Kashmir Assembly dataset are shown in the confusion matrix Figure 5.3.4.1. The sensitivity, specificity, accuracy, precision, and misclassification rates derived from the Figure are explained as follows:

The random forest model can recognize the positive (winning) cases with a sensitivity of 94% by putting values of Figure 5.3.4.1 in equation (3.11). “Similarly, the number of political parties and Independent candidates that had actually not won the election is equivalent to 95% that is obtained by using equation (3.12) in Figure 5.3.4.1. The closer the value for this measure is to 1, the better the rules are at identifying election outcomes.

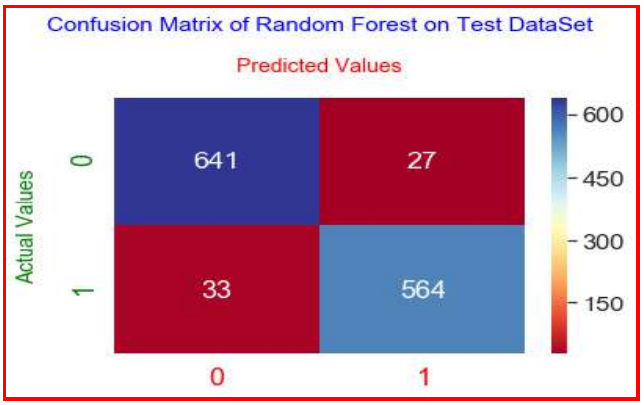

Figure 5.3.4.1: Random Forest Model Confusion Matrix on Test Dataset

The total accuracy of 0.95% is achieved by using the equation (3.13) in Figure 5.3.4.1 this means that the random forest election prediction model's overall accuracy (in diagnosing both the winning and losing the Assembly election cases) is 95%, the higher the accuracy percentage, the excellent the model is. Similarly, the precision of 0.95% is obtained using equation (3.14) in Figure 5.3.4.1. The closer the value for this measurement is to 1, the greater the chance that those with a positive result will have won the election. The error rate of 0.04% is obtained by putting the values of Figure 5.3.4.1 in equation (3.15). The lower the percentage of misclassification rate of the model, the more accurate the model is in identifying the election outcomes.

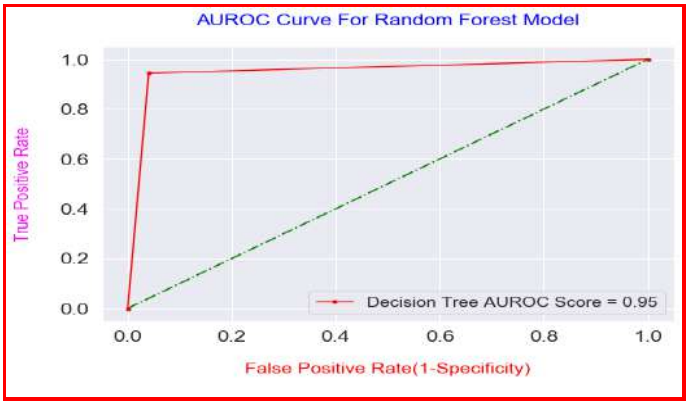

Figure 5.3.4.2: AUROC of Random Forest Model

The AUROC score is calculated to check the probability curve and measure of separability obtained by the random forest model." The AUROC tells how good the model can differentiate

among a won and lost case of Assembly election at constituency level. Above Figure 5.3.4.2 shows AUROC obtained from the random forest model with an AUROC score of =0.95%.

The researcher simulates the accomplished experimental results of the developed random forest election prediction model with the prevailing research; the results obtained are optimal. Hence the proposed random forest model is used for the early prediction of the election outcomes for Jammu and Kashmir. However further improvements is required in the Random Forest model which will be done in chapter v using hyperparameter tuning.

From the results of all derived models we came to know that in most of the models hyperparameters tuning is prerequisite for optimal results. Hence, in next chapter we will do hyperparameters tuning of all the derived models.

5.4 Performance Comparison of The Developed Election Prediction Models

This part shows the performance and comparison of the Decision Tree, K Nearest Neighbor, Support Vector Machine, and Random Forest election prediction models through different measures as described in the table 5.4.

Table 5.4: Performance Measures of Election Prediction Models

Performance Measures of The Models						
Models	**AUROC Score**	**F1 Score**	**Classifiers Accuracy**	**Recall Score**	**Precision Score**	**Miss-Classification Score**
Decision Tree	0.9134%	0.9090%	0.9130%	0.9212%	0.8972%	0.0869%
Random Forest	0.9521%	0.9494%	0.9525%	0.9447%	0.9543%	0.0474%

K - Nearest neighbors	0.8581%	0.8504%	0.8584%	0.8525%	0.8483%	0.1415%
Support Vector Machine	0.7986%	0.7535%	0.8086%	0.6197%	0.9610%	0.1913%

Experimental results demonstrate that the Random forest model performs most excellent in comparison to other predictions model. The results of the developed election prediction model are tested with the prevailing evaluation tools which demonstrate that the results are exceptionally encouraging with outstanding predictive accuracy. After the basic assessment of experimental results, it is imperative to cautiously check and assess the data to derive useful insights, design optimal models, and find out significant factor settings.

The results show that the Random forest model outperforms other election prediction models with an optimal accuracy of 95%, the specificity of 95%, the sensitivity of 94%, precision of 95%, AUROC score of 95% and with less misclassification rate of only 04%. The accuracy obtained by the Random Forest model is highest for forecasting election Prediction and is not achieved by previous studies.

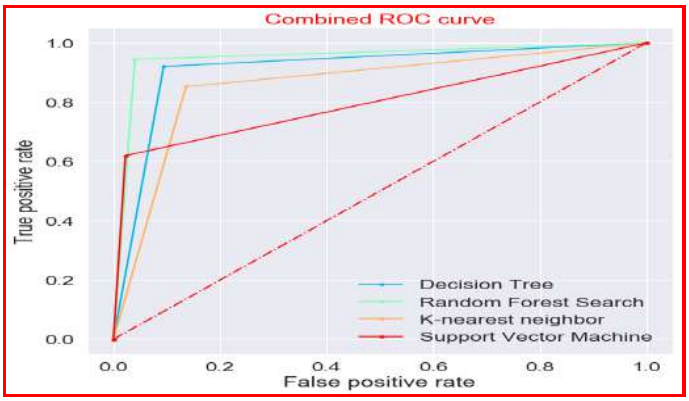

Figure 5.4 :Combined AUROCs of the Election Prediction Models

The Figure 5.4 shows the combined AUROC curves of different developed election prediction

models. Figure 5.4 also shows that Random forest model outperforms other election prediction models with the highest AUROC score of 0.95%, which means the model is highly skilful in predicting the winning and losing of political parties and independent candidates.

5.5 Summary

In this research work, KDD data mining methodology is followed to design an election prediction model. The Jammu and Kashmir assembly election dataset is mined using various feature selection methods to select the optimal set of attributes. These significant attributes are used for data mining techniques for the prediction of election outcomes constituency wise. The data mining methods like Decision Tree, Support Vector Machine, K-Nearest Neighbor and Random Forest, are applied for the early prediction of election outcomes. Experimental results show that the Random Forest model surpass other existing models with the highest model accuracy, low misclassification rate. However, the main problems which the researcher encountered in the utilized classifiers are overfitting and will try to reduce the problem of overfitting in chapter V, so as to develop an accurate election prediction model. Then the developed model will help user in early prediction of election results hence would reduce progression to severe complications.

Chapter 6

6. Results Discussion and Validation

6.1 Introduction

Election prediction is a complicated process because of its underlying complications. To predict the election results more precisely and efficiently with minimum errors rate, the researcher optimizes the hyperparameters of the proposed models in this research work. Hyperparameter optimization is the process of tuning the most favorable hyperparameters for a learning model [177]. It essentially means searching through an enormous universe of possible combinations of hyperparameters for the set that optimizes the desired Figure of merit [178]. The model parameters are learned through the training phase however the hyperparameters are optimized separately [179].

This chapter describes how to optimize the hyperparameters of the proposed models build in chapter IV. In this chapter, the researcher also explains the various types of hyperparameter optimization techniques and the comparison of the election prediction evaluation models with ensemble model. Finally, discuss the different combinations of election prediction analysis, generated election outcomes rules, election prediction evaluation model components, the developed election prediction model and end up with the chapter summary and conclusion.

6.2 Hyperparameter Optimization Techniques

Hyperparameters are those parameters whose value one set at training phase before developing a machine learning model [180]. Hyperparameters are optimized by analyzing different settings to analyze which values give optimal accuracy [181]. Hyperparameter optimization frequently involves fine-tuning parameters that reside outside the model, but that

can profoundly influence its behavior [178],[182]. The pictorial representation of hyperparameter optimization is shown in below-given Figure 6.2 [183].

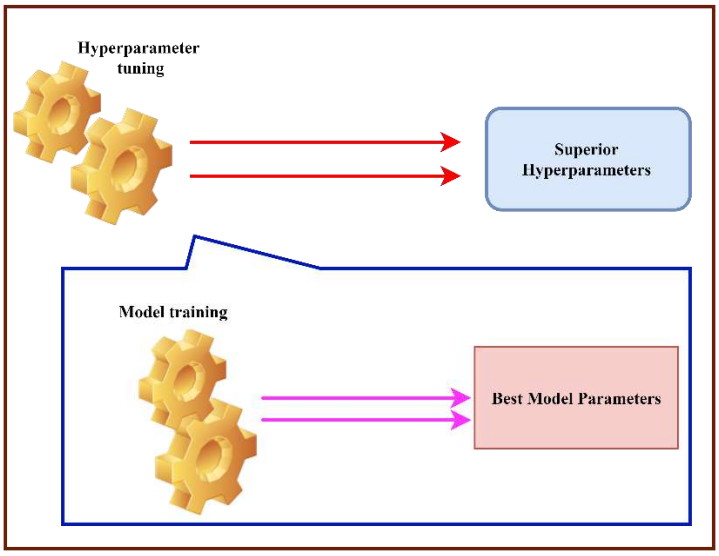

Figure 6.2: The Model and Hyperparameter Optimization Representation

Hyperparameter tuning for performance optimization is an art and choosing appropriate hyperparameters will generate the most accurate outcomes and will give us highly valuable insights into our data [184]. The hyperparameters help us to control the behavior of the machine learning models when optimizing for the performance and when finding the right balance between bias and variance [185]. Below given equation 6.1 shows the representation of hyperparameter optimization:

$$x^* = arg\min_{x \varepsilon X} f(x) \qquad (6.1)$$

In above equation f(x) means an objective score; x* is the set of hyperparameters which provides the lowest value of the score. Actually, the focus is to find the hyperparameters which give the optimal results on the validation set metric. There are various hyperparameter techniques defined in the literature, but in this research work only the most efficient techniques are being discussed.

6.2.1 Grid Search Hyperparameter Optimization

Grid search hyperparameter optimization is a comprehensive search of candidate parameter values over all feasible values in the defined search space [186], [187]. The Figure 6.2.1[188] shows the grid search method layout. It trains algorithm for every combination by using learning rate and measures the performance using cross validation technique [187]. Through this validation method one obtains the maximum valid patterns from the dataset. Although grid search technique is easy to use however it creates problems due to the *curse of dimensionality*. Because of this limitation it is only feasible to use Grid search on small number of configurations.

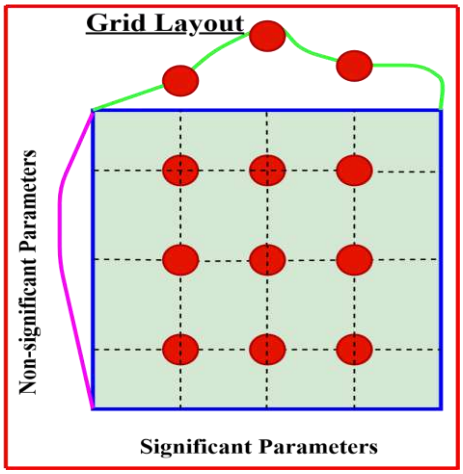

Figure 6.2.1: Grid Search Layout

6.2.2 Random Search Hyperparameter Optimization

In a random search, hyperparameter values are randomly selected from the defined search space[188]. Random sampling allows the search space to include both discrete and continuous hyperparameters. Random search uses previous knowledge and specify the distribution from were to sample [189]. Figure 6.2.2 [188] explains the random search optimization layout. Grid and Random search hyperparameter optimizations are unaware about the previous

evaluations, which results in evaluating “bad” hyperparameters hence is a time consuming process [178].The Grid and Random search optimization techniques are based on experienced machine learning practitioners and are the heuristic technique [182].

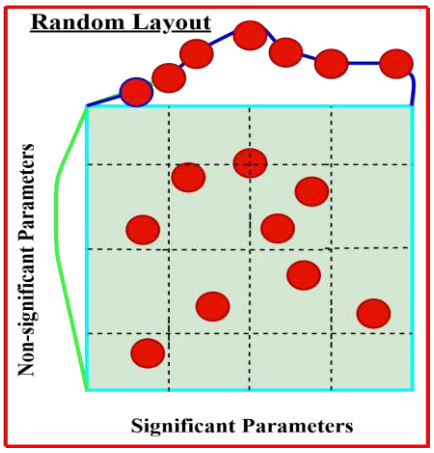

Figure 6.2.2: Random Search Layout

In these techniques, the human expertise will not get optimum setting of hyperparameters due to mishandling of high dimensional data and can easily misinterpret when trying to tune multiple hyperparameters [142]. Due to these drawbacks associated with the Grid and Random search hyperparameter optimization techniques, the researcher used the Bayesian hyperparameter Optimization for the development of the election prediction model for Jammu and Kashmir.

6.2.3 Bayesian Hyperparameter Optimization

Bayesian optimization builds a probabilistic model of the function mapping from hyperparameter values to the objective evaluated on a validation set [182]. In Grid Search and Random Search, the investigators try the configurations randomly and blindly and the new trial is independent of all the trials done before [191]. In contrast, automatic hyperparameter tuning forms knowledge about the relation between the hyperparameter settings and model performance in order to make a smarter choice for the next parameter settings [192].

Typically, it will first collect the performance at several configurations, then make some inference and decide what configuration to try next. The purpose is to minimize the number of trials while finding a good optimum. There are two major choices that must be made when performing Bayesian optimization.

1. Select prior over functions that will express assumptions about the function being optimized. for this, researcher choose the *Gaussian Process* prior.

2. Next, the researcher must choose an *acquisition function* which is used to construct a utility function from the model posterior, allowing us to determine the next point to evaluate.

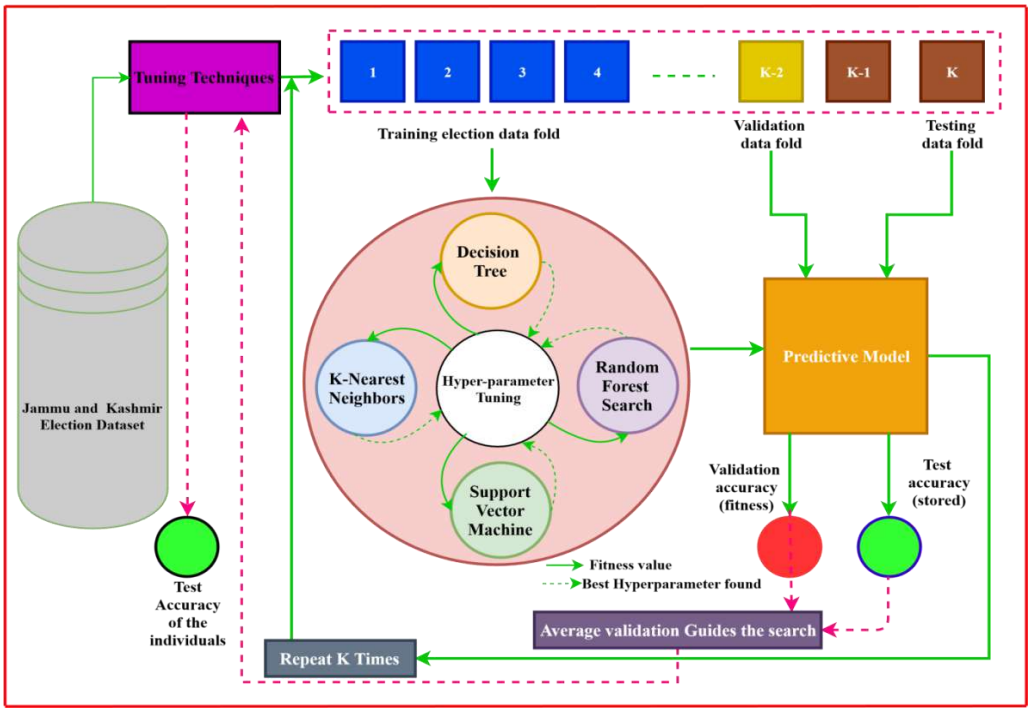

Figure 6.2.3: Single Cross-Validation Experimental Methodology for Hyperparameter Tuning

3. In Bayesian Optimization with each tuning technique, one uses the single Cross-validation (S-CV) methodology depicted in Figure 6.2.3.

When a hyperparameter tuning is executed, the election data-set is divided into k stratified folds. Algorithms like Decision Tree, Random Forest Search, K-NN and SVM are trained with k–2

partitions (training folds) for each candidate solution found by the technique. One set is used to validate the model and other fold is used to test. “The test and validation accuracies are assessed through the model induced with the training partitions and the values of the hyperparameters found by the optimization technique. This process is repeated for all k permutations in single cross-validation. The average validation accuracy is then used as the fitness value, which will guide the search process at the end, the individual with the highest validation accuracy is returned (with its hyperparameters values), and the technical performance is considered the average test accuracy of the individual.

6.3 Optimizing the Election Prediction Model

The hyperparameters are used to evaluate model parameters that are unused by the trained model for prediction purposes. Machine learning algorithms consist of several hyperparameters whose values play significant role in performance of the induced models in complicated ways [185]. Because there are large numbers of promises of hyperparameter settings, the researcher lack knowledge into how to intelligently analyze this huge space of configurations.

The researcher set the random state as “42” in all the machine learning classifiers that are being used in building the model, because if researcher don’t set the random state then every time, when researcher run the model a new random value is being generated and the test and trained dataset of the model has different values each time. Hence by mentioning the random state as “42” every time the researcher run the model and get the same results (i.e same value in train and test datasets). There are tons of potential parameters to tune on every model and all the parameters are valuable but researcher need to select the significant subset of parameters to tune.

6.3.1 Decision Tree Hyperparameter Optimization Model

To obtain the optimal accuracy the researcher tunes the most important hyperparameters of Decision tree model however researcher should be careful to validate them on test data fold in order to avoid overfitting. The significant hyperparameter subsets that progress the performance of the decision tree model that need to be tuned are as follows:

Table 6.3.1: Hyperparameter Optimization Results of the Decision Tree Model

Max depth	Min samples leaf	Min samples split	Random state	Accuracy
10	55	30	42	76%
10	55	30	42	82%
10	130	200	42	80%
5	20	42	42	64%
10	20	42	42	73%
8	5	15	42	78%
9	55	90	42	81%
9	30	183	42	82%
9	30	183	42	73%
6	55	30	42	74%
7	55	30	42	75%
7	30	50	42	77%
8	30	50	42	78%
8	30	150	42	77%
8	90	150	42	79%
8	70	150	42	78%
8	70	110	42	78%
8	10	150	42	77%
8	10	180	42	76%
9	10	183	42	82%
10	20	70	42	82%

1. Max Depth: this hyperparameter shows how deep the tree can be. If the tree is deeper then it means it has large number of splits and can achieve maximum information from the data [144]. In this max depth hyperparameter the researcher set the range of trees from 1 to 32 and plot both training and testing AUROC scores. The decision tree election prediction model overfits when value is set high. This model fails to generalize the findings for new data.

2. Min Samples Leaf: this hyperparameter shows the less number of samples needed to be at a leaf node [193]. Increasing this value causes underfitting.

3. Min Samples Split: it represents the total minimum number of records needed to split an internal node [194]. The value varies from one record to including all the samples at each node. If the value of this hyperparameter is increased the tree becomes more constrained. When all the samples at each node are included the model underfits.

After tuning the hyperparameters of the decision tree model the researcher obtained the results as shown in above table 6.3.1.

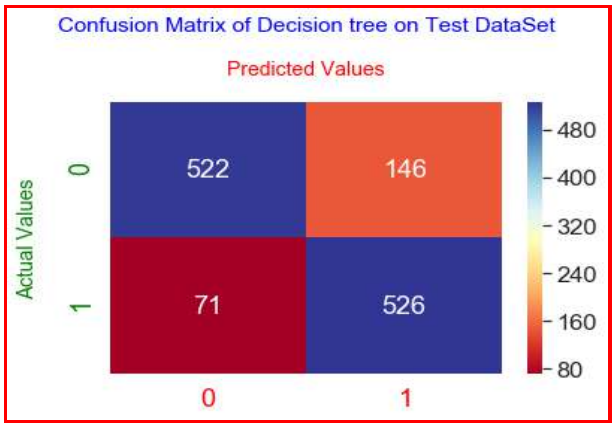

Figure 6.3.1.1: Confusion Matrix of the Hyper-parameterized Decision Tree Model

The permutations and combinations of the hyper parameterized decision tree model shows different results only those combinations which provide the highest accuracies are shown.

The researcher applied the optimized decision tree algorithm for prediction of election outcomes and the performance results are depicted in confusion matrix Figure 6.3.1.1. From

the Figure 6.3.1.1, the researcher derived Recall, Specificity, Recognition Rate, Precision and Misclassification Rate which are described as follows:

Using equation (3.11), researcher obtained the True Positive Rate of the decision tree model as $\frac{(526)}{(71+526)}$which is equivalent to 0.88% which means our decision tree model can recognize the positive election won cases with an accuracy of 88%. Similarly, by using equation (3.12) researcher get the True Negative Rate of the decision tree as $\frac{(522)}{(522+146)}$ which is equivalent to 0.78% that means our decision tree model can recognize the election lost cases with an accuracy of 78%. The accuracy of the decision tree model is obtained by using the equation (3.13) that is $\frac{(522\ +\ 526)}{(522+146+71+526)}$ after calculations which is equivalent to 0.82% which means that the decision tree model's overall performance in diagnosing both the won and lost case of election results is 82%. Similarly, to obtained the Precision of the hyper-parameterized decision tree model the equation (3.14) is used as $\frac{(526)}{(526+146)}$ that after an evaluation is 0.78% this means that our hyper-parameterized decision tree model has a low false-positive rate. The misclassification rate of the proposed decision tree model is obtained using the equation (3.15) $\frac{(146+\ 71)}{(522+146+71+526)}$ which is equivalent to 0.17%. The researcher also used the AUROC performance measurement to see distinguishing ability of the model among won and lost cases of election outcomes. Intelligent models distinguish among these winning and losing cases however; the poor models will have difficulties in distinguishing between the two. The below given Figure 6.3.1.2, shows the AUROC curve obtained from the decision tree model with an AUROC score of =0.83%. The researcher simulates the accomplished experimental outcomes of the optimized election prediction model with the prevailing research; the results obtained are greater than the published values.

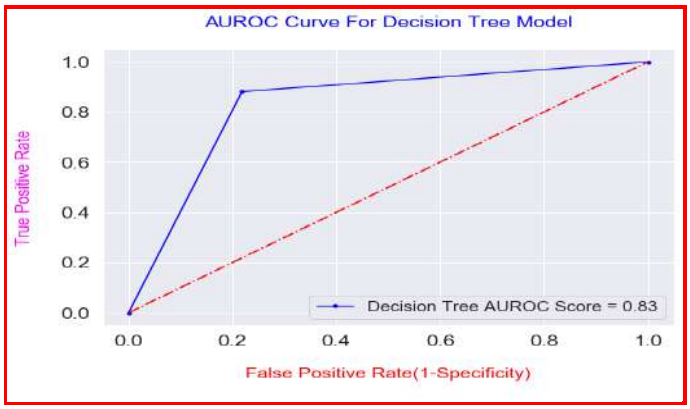

Figure 6.3.1.2: Optimized Decision tree AUROC Curve

Hence, this result generated the proposed optimized decision tree model for predicting the election outcomes constituency wise of Jammu and Kashmir quite easily however, further improvement in model performance is required.

6.3.2 K-Nearest Neighbor Hyperparameter Optimization Model

K-NN classifies an unknown instance on the basis of majority votes [195]. Each instance is provided with equal based on the distance [196]. The researcher run a hyperparameter tuning job for K-NN to obtain the optimal model which shows highest accuracy and lowest error on the test election dataset. The investigators search the significant parameters of K-NN and check how they influence model in terms of total error. There are *three primary hyperparameters* that one can tune:

1. The number of neighbours' ***k.***
2. The distance parameter.
3. The weight parameter is used to describe the neighbors weights.

All of these hyperparameters play significant role in affecting the accuracy of K-NN classifier. The optimal k is one that gives minimum error and in order to find that researcher perform repeated measurements of the test error for different values of K. Different validation approaches

are used in practice, but in this work, researcher explore the 10-fold cross-validation. Another essential thing to note is that the K-NNs time complexity is enormous because it measures the individual distances for each point in the test set. Optimization Search CV functions by training model multiple times on a range of attributes that the researcher mention and check the best values to obtain optimal accuracy. The experimental results of the hyper-parameterized K-NN model are shown in below-given table 6.3.2.

The researcher used the different permutations and combinations of the K-NN parameter combinations best score functions to attain the maximum accuracy of our model because the 'best score' outputs the mean accuracy of the scores obtained through cross-validation. When the value of k is small then one gets the low bias and high variance however, when k is large then high bias and low variance is obtained. So, the researcher in such combinations set the value of k parameter in sweet position and using such optimization search to improved our model accuracy up to 81%. The parameter combinations which result in the highest accuracies are described in the table 6.3.2. The experimental results showed that when the hyperparameter combinations are [Leaf Size= 25, N neighbours= 11, Weights=uniform, Metrics= minkowski, random state=42] the highest accuracy of 81% is achieved.

Table 6.3.2: Hyperparameter Optimization Results of the K-NN Model

Leaf size	**N neighbors**	**weights**	**Metric**	**Random State**	**Accuracy**
30	11	uniform	Minkowski	42	81%
50	13	uniform	Minkowski	42	81%
30	13	uniform	Minkowski	42	80%
20	13	uniform	Minkowski	42	81%
20	13	uniform	Euclidean	42	81%
20	19	uniform	None	42	80%
20	15	uniform	Euclidean	42	80%

20	19	None	Minkowski	42	80%
25	17	uniform	Minkowski	42	79%
20	19	uniform	None	42	80%
20	19	uniform	Euclidean	42	80%
25	17	uniform	Euclidean	42	79%
20	19	None	Minkowski	42	80%
10	11	uniform	Minkowski	42	81%
15	11	uniform	Minkowski	42	81%
40	11	uniform	Minkowski	42	81%
50	11	uniform	Euclidean	42	81%
25	11	uniform	Euclidean	42	81%
25	15	uniform	Euclidean	42	80%
25	15	uniform	Minkowski	42	80%
20	19	uniform	Minkowski	42	80%
30	19	uniform	None	42	80%
25	11	Uniform	Minkowski	42	81%

The performance results of the proposed K-NN model are shown in confusion matrix Figure 6.3.2.1. From the Figure 6.3.2.1, the researcher derived the Recall, Specificity, Recognition Rate, Precision and Misclassification Rate which are described as follows:

Using equation (3.11) the researcher obtained the True Positive Rate of the K-NN model as $\frac{(504)}{(504+93)}$ which is equivalent to 0.84% hence our K-Nearest Neighbor model can recognize the winning party or independent candidate's cases with an accuracy of 84%. Similarly, by using equation (3.12) the researcher gets the True Negative Rate of the K-Nearest Neighbor as $\frac{(529)}{(529+139)}$ which is equivalent to 0.79% that means our K-NN model can recognize the party or independent candidates that don't won the elections with an accuracy of 79%. The overall accuracy of the K-NN model is obtained by using the equation (3.13) that is $\frac{(529 + 504)}{(529+139+93+504)}$

after calculations which is equivalent to 0.81% which means that the K-NN model's overall performance in diagnosing both the election won and lost cases is 81%.

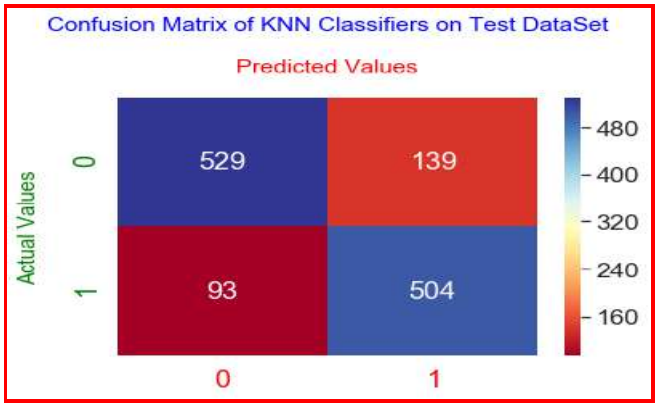

Figure 6.3.2.1: Hyper-parameterized K -NN Confusion Matrix

To obtain the Precision of the hyper-parameterized K-NN model the equation (3.14) is used and the results are evaluated as $\frac{(504)}{(504+139)}$ that is equal to 0.78% this means that the hyper parameterized K-NN model has low false positive rate. The misclassification rate of the proposed K-NN model is obtained using the equation (3.15) $\frac{(139+\ 93)}{(529+139+93+504)}$ which is equivalent to 0.18%. The researcher also used AUROC to check how efficiently the model can differentiate among the election won and election lost by party or candidates at constituency levels.

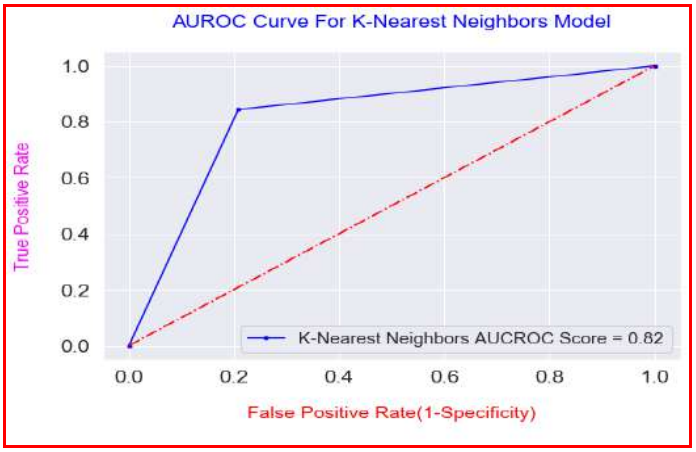

Figure 6.3.2.2: AUROC Curve of Hyper-Parameterized K NN Model

Intelligent models differentiate among the election winning or lost cases however the poor models will have difficulties in distinguishing between them. Figure 6.3.2.2 shows AUROC curve obtained from the K-NN model with an AUROC score of =0.82%. The researcher simulates the accomplished experimental results of the optimized K-NN election prediction model with the prevailing research; the results obtained are optimal than the published values in literature.

Hence, the result generated proposed optimized K-NN model for predicting the Assembly election outcomes constituency wise of Jammu and Kashmir efficiently. However, further improvement in model performance is required.

6.3.3 Support Vector Machine Hyperparameter Optimization Model

To obtain the highest accuracy for the early prediction of the election outcomes, the researcher tunes the most significant hyperparameters of the SVM model; however, researcher should be careful to validate them on the test dataset. The hyperparameters of the SVM model that the researcher optimized are as follows:

1. Kernel: this hyperparameter selects the type of hyperplane used to separate the data [185]. The main function of the kernel is to transform the given input data into the required form. The values for this hyperparameter could be linear, polynomial, sigmoid and radial basis function [198].

2. Regularization: This hyperparameter is incorporated to preserve regularization. It is a penalty parameter, which represents misclassification or error term [199]. A lesser value to regularization parameter creates a small-margin hyperplane and a higher value of its creates a larger-margin hyperplane [200].

3. Gamma: Gamma is a parameter for non-linear hyperplanes. A lower value of Gamma will loosely fit the training dataset, whereas a higher value of gamma causes over-fitting [201]. Researcher can see that increasing gamma leads to overfitting as the classifier tries to fit the training data perfectly.

4. Probability: The behaviour of the model is very sensitive to the gamma parameter. If the researcher set the value of gamma too large it results in overfitting and when the value of gamma is small, the model underfits [202]. For intermediate values, researcher can see that a good model can be found on a diagonal of nu and gamma.

Below table 6.3.3 shows the different accuracies obtained after tuning the various hyperparameters of SVM. The "Kernel", "nu", "Gamma", and probability hyperparameters of the Support Vector Machine model are used to obtain the optimal accuracy.

Table 6.3.3: SVM Hyperparameters Optimization with their Accuracies

nu	***Gamma***	**Kernel**	***Probability***	**Accuracy**
0.1	0.1	Rbf	True	81%
0.6	0.01	Rbf	True	78%
0.2	0.01	Rbf	True	83%
0.8	0.01	Rbf	True	76%
0.6	0.01	Rbf	True	78%
0.7	0.01	Rbf	True	76%
0.7	0.001	Rbf	True	78%
0.5	0.0001	Rbf	True	83%
0.6	0.0001	Rbf	True	81%
0.7	0.0001	Rbf	True	79%
0.8	0.0001	Rbf	True	78%
0.9	0.0001	Rbf	True	77%
0.9	0.001	Rbf	True	75%
0.8	0.001	Rbf	True	77%
0.7	0.001	Rbf	True	78%

0.6	0.001	Rbf	True	79%
0.5	0.001	Rbf	True	83%
0.5	0.0001	Rbf	True	83%
0.6	0.1	Rbf	True	78%
0.7	0.1	Rbf	True	76%
0.8	0.1	Rbf	True	76%

Researcher found that when the **Kernel** hyperparameter values are set to (***linear*** or ***sigmoid)*** models time complexity increases. Experimental outcomes showed that when the SVM hyperparameter combinations are [nu=0.5, Kernel=rbf, Gamma=0.0001, *Probability* =True] the highest accuracy of 83% is achieved. The performance results of the model are shown in confusion matrix Figure 6.3.3.1.

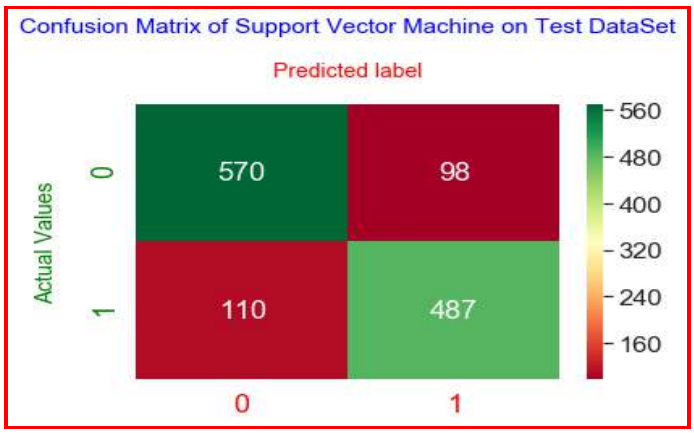

Figure 6.3.3.1: Hyper-Parameterized SVM Confusion Matrix

From the above Figure 6.3.3.1, researcher derived the Recall, Specificity, Recognition Rate, Precision and Misclassification Rate which are described as follows:

Using equation (3.11) researcher obtained the True Positive Rate of the SVM model as $\frac{(487)}{(487+110)}$ which is equivalent to 0.81% hence the SVM model can recognize the positive election won cases with an accuracy of 81%. Similarly, by using equation (3.12) researcher got the True Negative Rate of the SVM model as $\frac{(570)}{(570+98)}$ which is equivalent to 0.85%, that means our

hyper-parameterized SVM model can recognize the election lost cases with an accuracy of 85%. The accuracy of the SVM model is obtained by using the equation (3.13) that is $\frac{(570 + 487)}{(570+98+110+487)}$ which is equivalent to 0.83%, which means that the SVM model's overall performance in diagnosing both the election won and election lost cases is 83%. To obtain the Precision of the hyper-parameterized SVM model the equation (3.14) is used and the values are evaluated as $\frac{(487)}{(487+98)}$ which show the results as 0.83%, this means that the hyper-parameterized SVM model has low false positive rate. The misclassification rate of the proposed SVM model is obtained using the equation (3.15) $\frac{(110+ 98)}{(570+98+110+487)}$ which is equivalent to 0.16%. The researcher also used AUROC performance measurement to see how efficient the model is in distinguishing between them (*e.g. If a party or independent candidate won or lost the election*).

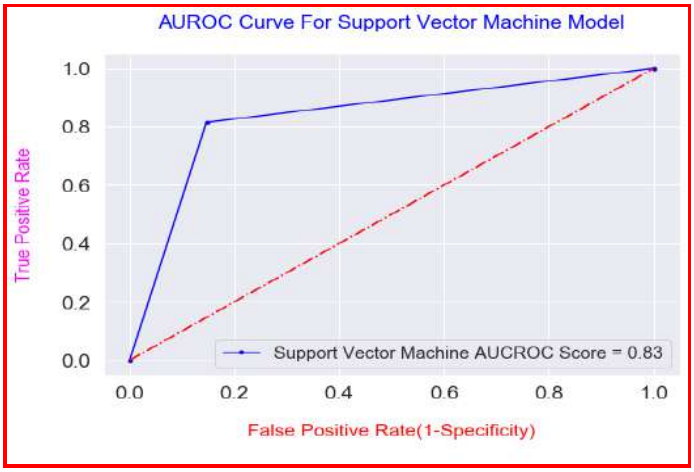

Figure 6.3.3.2: AUROC Curve of Hyper-Parameterized SVM Model

Figure 6.3.3.2 shows AUROC curve obtained from the SVM model with an AUROC score of =0.83%. The researcher simulates the accomplished experimental results of the optimized SVM election prediction model with the prevailing research; the experimental outcomes of the SVM model are excellent for predicting election outcomes.

6.3.4 Random Forest Hyperparameter Optimization

Random forest fits several decision tree algorithms on various subsamples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting [203]. Researcher explored the most critical parameters of the random forest algorithm which are discussed below:

1. Bootstrap: It is a method for sampling data points.

2. Max Depth: this hyperparameter specifies maximum depth of each tree. The default value for max depth is None, which means that each tree will expand until every leaf is pure. A pure leaf is one where all of the data on the leaf comes from the same class

3. N Estimators: N estimators represent the number of trees in the forest. If there are maximum numbers of trees then the model learns efficiently from the data. However, adding trees can slow down the training process considerably, therefore researcher do a parameter search to find the best spot.

4. Min Samples Split: this hyperparameter shows the minimum number of samples required to split an internal node. Its value varies from one sample to all of the samples at each node. If researcher consider all instances at each node, this may lead to an underfitting problem.

5. Min Samples Leaf: this hyperparameter represents minimum number of records needed to be at a leaf node. This parameter is similar to Min Samples Split; however, this describes the minimum number of samples at the leaf, the base of the tree. Increasing this value can cause underfitting.

After hyper parameter tuning of the random forest model, researcher obtained the results as shown in below-given table 6.3.4. The permutations and combinations of the hyper-parameterized random forest model shows different results; however, researcher described only those parametric combinations which provide the highest accuracies. The experimental results

shows that when the hyperparameter combinations are [Boot-strap = True, Max Depth= 100, Min samples leaf=50, Min samples split=110, N Estimators=1000, Random state=42,] the highest accuracy of 85% is achieved.

Table 6.3.4: Random Forest Hyperparameters Optimization with their Accuracies

Boot-strap	Max depth	Min samples leaf	Min samples split	N estimators	Random state	Accuracy
True	200	10	150	1000	42	85%
True	500	10	150	1000	42	85%
True	100	10	150	1000	42	85%
True	100	5	150	1000	42	85%
True	100	20	150	1000	42	84%
True	100	20	170	1000	42	83%
True	100	20	170	100	42	83%
False	100	20	270	100	42	83%
False	100	40	270	100	42	83%
False	100	50	270	100	42	82%
True	100	50	150	1000	42	83%
True	100	50	130	1000	42	84%
True	100	50	110	1000	42	85%

Researcher applied hyper-parameterized random forest model for the prediction of election outcomes constituency wise of Jammu and Kashmir. The performance results of the model are shown in confusion matrix Figure 6.3.4.1. From the Figure 6.3.4.1, researcher derived the Recall, Specificity, Recognition Rate, Precision and Misclassification Rate which are described as follows: Using equation (3.11) researcher obtained the True Positive Rate of the random forest model as $\frac{(505)}{(505+92)}$ which is equivalent to 0.84% hence our random forest model can recognize the positive election winning cases with an accuracy of 84%.

Similarly, by using equation (3.12) researcher get the True Negative Rate of the random forest model as $\frac{(581)}{(581+87)}$ which is equivalent to 0.86% that means the random forest model can recognize the election lost cases with an accuracy of 86%.

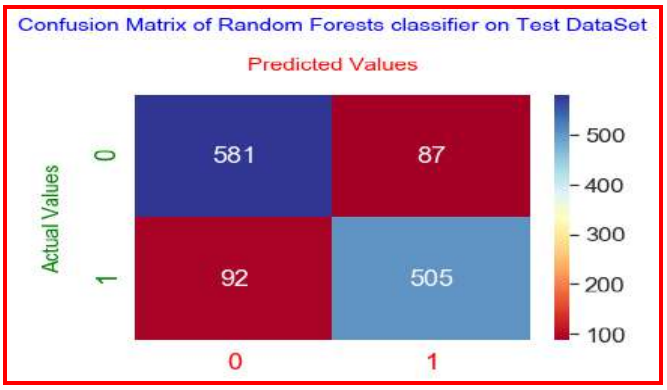

Figure 6.3.4.1: Hyper-Parameterized Random Forest Confusion Matrix

The accuracy of the random forest model is obtained by using the equation (3.13) that is $\frac{(581 + 505)}{(581+87+92+505)}$ after calculations which is equivalent to 0.85%, which means that the hyper-parameterized random forest model's overall performance in predicting both the election won and lost cases is 85%. To obtain the Precision of the hyper-parameterized random forest model the equation (3.14) is used and the results are obtained as $\frac{(505)}{(505+87)}$ after evaluating the equation the result is 0.85%, this means that our hyper-parameterized random forest model has low false positive rate. The misclassification rate of the proposed random forest model is obtained using the equation (3.15) $\frac{(87+ 92)}{(581+87+92+505)}$ which is equivalent to 0.14%. Researcher also used another performance evaluator called AUROC to check the probability curve and measure of separability achieved by random forest model. AUROC describes how efficiently the model can differentiate among the election won or lost cases by the party or independent candidates. Efficient models exactly differentiate among them; however, the poor models fail in so. Above

Figure 6.3.4.2 shows AUROC curve obtained from the random forest model with an AUROC score of 0.86%.

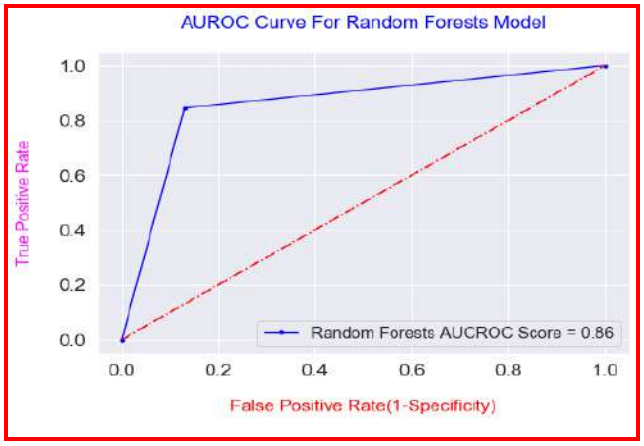

Figure 6.3.4.2: AUROC Curve of Hyper-Parameterized Random Forest Model

Researcher simulate the accomplished experimental results of the optimized random forest election prediction model with the prevailing research; the results obtained are higher than the published values however further improvement in model performance is required. Hence, researcher used the proposed optimized random forest model for predicting the election outcomes of Jammu and Kashmir constituency wise effectively.

6.4 Performance Comparison among Hyperparameterized Models

In this section, researcher present an evaluation and comparison of the hyper-parameterized models. To check the performance of the developed hyper-parameterized election prediction model, researcher used the different performance metrics which are described in the table 6.4. The experimental results show that the hyper-parameterized Random Forest model outperforms other models. The performance of the proposed election prediction model is checked with prevailing models which show that the results are very promising with excellent predictive power. The results show that the Random Forest model obtains the optimal accuracy of 85%, with a minimum misclassification rate of only 0.14%.

Table 6.4: Performance Metrics of the Different Hyperparameterized Election Prediction Models

Performance Measures of The Models						
Models	**AUROC Score**	**F1 Score**	**Classifiers Accuracy**	**Recall Score**	**Precision Score**	**Miss-Classification Score**
Decision Tree	0.8312%	0.8289%	0.8284%	0.8810%	0.7827%	0.1715%
Random Forest	0.8578%	0.8494%	0.8584%	0.8458%	0.8530%	0.1415%
K -Nearest Neighbors	0.8180%	0.8129%	0.8166%	0.8442%	0.7838%	0.1833%
Support Vector Machine	0.8345%	0.8240%	0.8355%	0.8157%	0.8324%	0.1644%

Figure 6.4 show the combined AUROC curves of different hyper-parameterized election prediction models. The Figure 6.4 shows that Random Forest election prediction model has highest AUROC score of 0.85% which means the model has the best ability to distinguish between the election won or lost by the political parties or independent candidates at constituency levels quite effectively however, further improvement is needed.

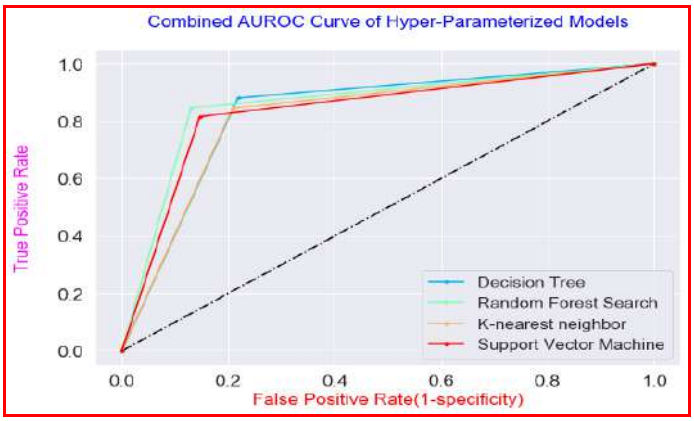

Figure 6.4: Combined AUROC Curves of the Proposed Election Prediction Models

6.5 Ensemble Methods

As discussed in (chapter 3rd) about the ensemble techniques (i.e Hard voting and Soft voting), the researcher applied soft voting in this research work because soft voting gives the probabilities mean of all the classifiers utilized. In this research work researcher utilized four different machine-learning classifiers like decision tree, random forest, K-nearest Neighbors and support vector machine.

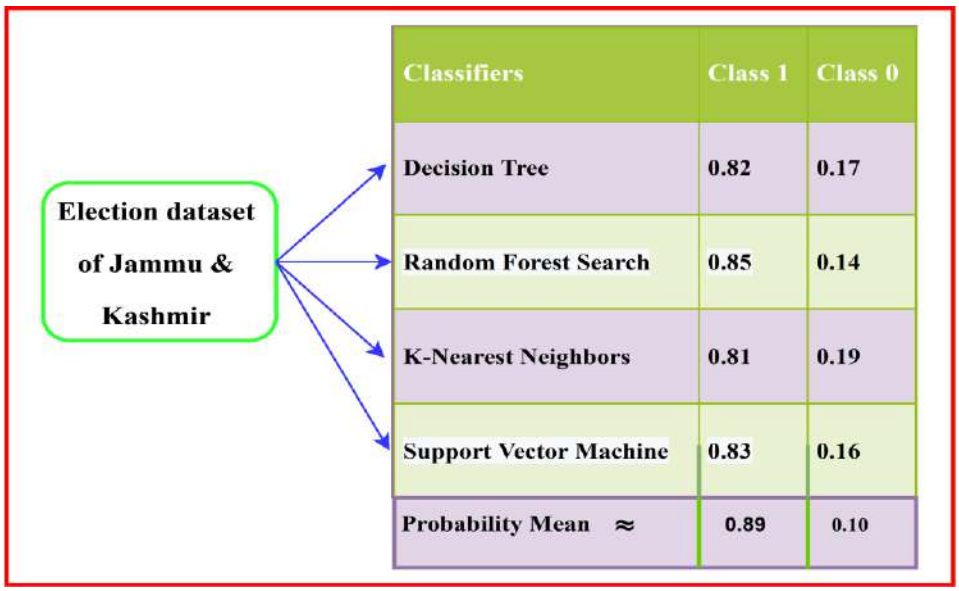

Figure 6.5: Ensemble (Soft voting) Election Prediction Model.

Now the question arises which classifiers researcher used in making the final outcomes of the election prediction model, mostly researcher applied only that classifiers which gives maximum accuracy. But in this research work researcher are taking the average probability mean of all the four machine learning classifiers by using an ensemble method (soft voting).

The overall accuracy of ensemble model is obtained by using the equation (3.6.2) in Figure 6.5 which is equivalent to 0.89% which represents that by applying soft voting in ensemble model, the election prediction model overall performance increases in diagnosing both the winning and losing election by the party or independent candidates.

6.5.1 Ensemble Model Experimental Results

The predictive outcomes of the ensemble model on the Jammu and Kashmir Assembly dataset are shown in the confusion matrix Figure 6.5.1.1. The sensitivity, specificity, accuracy, precision, and misclassification rates derived from the Figure 6.5.1.1 are explained as follows:

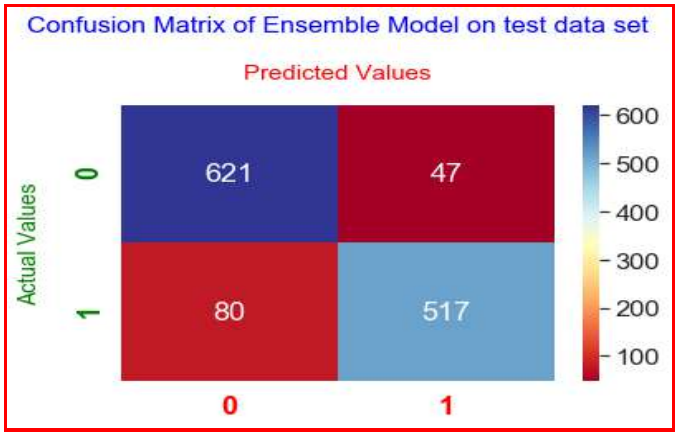

Figure 6.5.1.1: Confusion Matrix of Ensemble Model on Test Dataset

From the Figure 6.5.1.1, researcher derived the Recall, Specificity, Recognition Rate, Precision and Misclassification Rate of ensemble model which are described as follows: Using equation (3.11) researcher obtained the True Positive Rate of the ensemble model as $\frac{(517)}{(517+80)}$ which is

equivalent to 0.86% hence our ensemble model can recognize the positive election winning cases with an accuracy of 86%. Similarly, by using equation (3.12) researcher get the True Negative Rate of the ensemble model as $\frac{(621)}{(621+47)}$ which is equivalent to 0.92% that means the ensemble model can recognize the election lost cases with an accuracy of 92%. The accuracy of the ensemble model is obtained by using the equation (3.13) that is $\frac{(621 + 517)}{(621+47+80+517)}$ after calculations which is equivalent to 0.89% which means that the ensemble model's overall performance in predicting both the election won and lost cases is 89%. To obtain the Precision of the ensemble model the equation (3.14) is used and the results are obtained as $\frac{(517)}{(517+47)}$ after evaluating the equation the result obtained is 0.91%, this means that our ensemble model has low false positive rate. The misclassification rate of the proposed ensemble model is obtained by using the equation (3.15) $\frac{(47+80)}{(621+47+80+517)}$ which is equivalent to 0.10%.

Researcher also used another performance evaluator called AUROC to check the probability curve and measure of separability achieved by ensemble model. AUROC describes how efficiently the model can differentiate among the election won or lost cases by the party or independent candidates. Efficient models exactly differentiate among them; however, the poor models fail in doing so.

The Figure 6.5.1.2 shows AUROC curve obtained from the ensemble model with an AUROC score of =0.90%. Researcher simulated the accomplished experimental results of the developed ensemble election prediction model with the prevailing research; the outcomes obtained are greater than the published results in the literature. Hence this ensemble model is used for the early prediction of the election outcomes for Jammu and Kashmir. Figure 6.5.1.2 demonstrated

the predicted result of this method which shows that the applied technique is efficient in forecasting the assembly election outcomes of previous three elections.

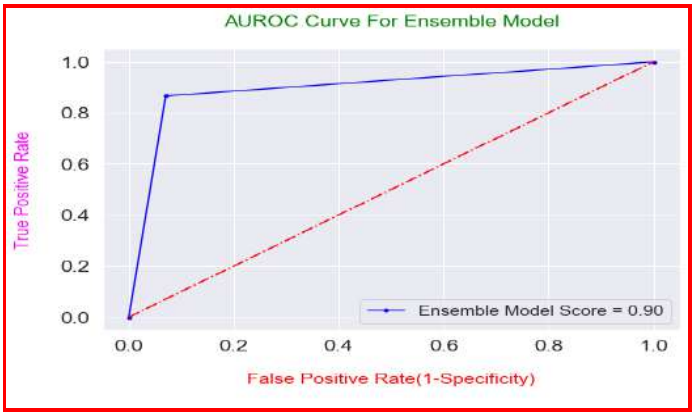

Figure 6.5.1.2: AUROC by Ensemble Model

6.6 Performance Comparison of Different Proposed Election Prediction Models

Below given table 6.6 demonstrate results calculated through different metrics. Experimental results show that the ensemble model gives higher accuracy and lower miss classification error values as compared to the other four selected classifications model. This indicates that the ensemble methods outperform other four selected classifications model.

Table 6.6: Performance Comparison of Different Proposed Election Prediction Models

	Performance Measures of The Models					
Models	**AUROC Score**	**F1 Score**	**Classifiers Accuracy**	**Recall Score**	**Precision Score**	**Miss-Classification Score**
Decision Tree	0.8312%	0.8289%	0.8284%	0.8810%	0.7827%	0.1715%
Random Forest	0.8578%	0.8494%	0.8584%	0.8458%	0.8530%	0.1415%
K -Nearest Neighbors	0.8180%	0.8129%	0.8166%	0.8442%	0.7838%	0.1833%

Support Vector Machine	0.8345%	0.8240%	0.8355%	0.8157%	0.8324%	0.1644%
Ensemble Models	0.9018%	0.8906%	0.8996%	0.8659%	0.9166%	0.1003%

6.7 T-Paired Test

In this statistical test researcher compared all the four machine learning models viz (Decision Tree, Random Forest, K-Nearest Neighbors and Support Vector Machine) with ensemble model one by one by using p-value criteria. If the p-value of models is less than 0.05, then the ensemble model is significant and this will be the evidence that ensemble model performs differently from selected machine learning models, hence researcher reject the null hypothesis**.** Since the null hypothesis is that there is no dissimilarity between the performances of two Machine Learning models, a p-value smaller than the considered significance level refuses the null hypothesis in favour of the alternative hypothesis, which assumes the Ensemble model perform differently to Machine Learning models. In addition, a p-value greater than the significance level represents that investigators fail to reject the null hypothesis. The above procedure is used to compare the performance of proposed optimized models with ensemble model.

The p-value for all the optimized models applied in this research work is less than the considered significance level (0.05%). Hence, researcher reject the null hypothesis. Therefore, the results statistically provide convincing evidence that hyper-parameterized decision tree, random forest, K- nearest Neighbors and SVM perform differently than ensemble model. On average, the mean accuracy of the ensemble model is more than that of all the utilized machine learning models.

Hence ensemble model is significant than the four utilized machine learning models in performance and accuracy. Therefore, researcher used ensemble technique in building the election prediction model for Jammu and Kashmir constituency wise. The results of which are enlisted in the below table 6.7.

Table 6.7: Comparison among proposed models with ensemble model using T-Paired test

S.No	Comparison of models with ensemble model	p value
1	Decision tree	0.006
2	Random Forest	0.001
3	K-Nearest Neighbors	0.009
4	Support Vector Machine	0.014

6.8 Rule Generation for Election Prediction Assessment

Election Prediction is considered to be a nonlinear problem that shows the complex causal relationship between the variables. Different researcher used numerous data mining techniques to assist poll forecaster in predicting election outcomes. In this work, Decision Tree, K-Nearest Neighbor, Random Forest, Support Vector Machine classifiers and finally ensemble model are proposed to determine the attributes which contribute more towards the political forecasting. The political forecasters are factor that influencing the rate at which the prediction progresses, and therefore predicts either the political parties or independent candidates can win or loses the election at different assembly constituency. Party wave, Central Government Influence, Religion Followers, Party Abbreviations, Sensitive Areas, Vote Bank, Hereditary, Incumbent

Party and Caste Factor are the most prominent factors for election prediction as per this research is concerned. The election prediction rules are extracted to help professional as well as naïve user in understanding how these attributes combinations are used by these proposed models in predicting the election outcomes. Figure 6.8 shows a subset of the rules extracted from Jammu and Kashmir election dataset. All rules are presented in Appendix A.

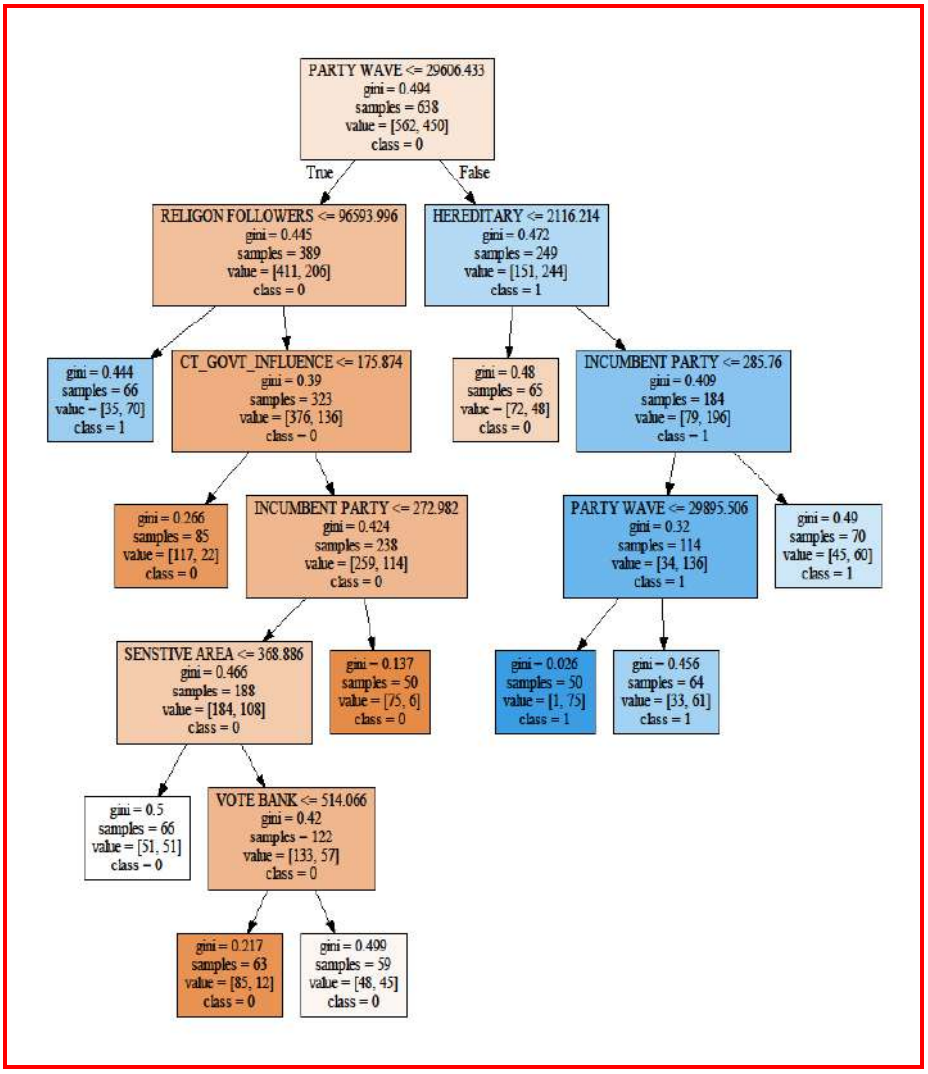

Figure 6.8: Election Prediction Decision Tree Using Nine Attributes

The generated rules are selected, pruned, evaluated and validated by different poll forecaster domain experts. Thus, the extracted decision tree rules can be used in political forecasting

environments to help general poll analysts establish an early prediction of election outcomes using nine attributes with a low-cost, reliable and effective evaluation model.

6.9 Election Prediction Expert System Evaluation Model Components

The proposed election prediction model is innovative because it identifies most accurately the chance of a political parties or independent candidates for winning or losing the elections using only the election data attributes, thus supporting its application as a public screening test. For simplicity, researcher have called this model as JKEPM (Jammu & Kashmir Election Prediction Model). Below given Figure 6.9 shows the three main components of JKEPM: the knowledge base; inference engine; and the interface.

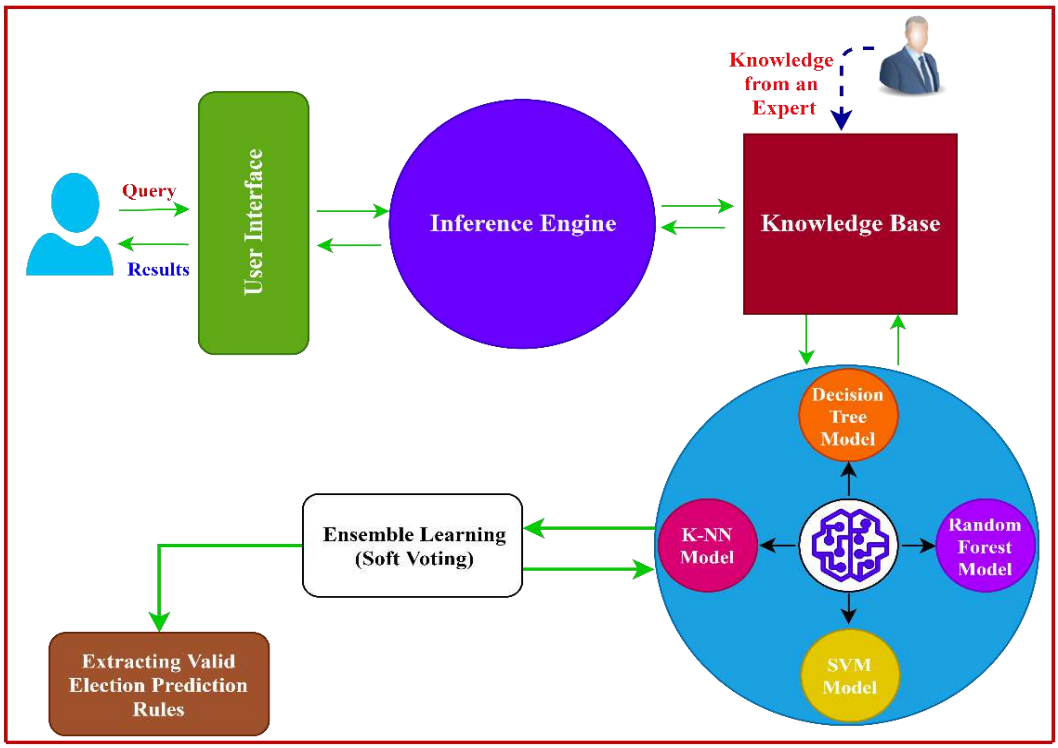

Figure 6.9: Election Prediction Evaluation Tool Components

The knowledge base applies the proposed models on the Jammu and Kashmir election data attributes to extract the expert system rules. The attributes are the Central Government Influence, Religion Followers, Party wave, Party Abbreviations, Sensitive Areas, Vote Bank,

Hereditary, Incumbent Party and Caste Factor. The inference engine uses the extracted rules and the users input to draw conclusions from the knowledge base and presents them to the user via the user interface. The user interface allows for "communication" screens where the user enters input data and the expert system returns the chance of winning or losing the election as calculated by the inference engine.

6.10 Jammu and Kashmir Election Prediction Model (JKEPM)

Researcher build an election prediction model for Jammu and Kashmir that can be used for predicting the election outcomes at constituency wise easily. In Chapter IV and Chapter V, the developed models used various attributes in building the election prediction model. The results demonstrate that the combination of Party wave, Central Government Influence, Religion followers, Party abbreviations, Sensitive areas, vote bank, caste factor, Incumbent party and Hereditary provides the best results. These results seem sufficiently high of the ensemble model which consists of four different machine learning algorithms viz (Decision Tree, Random Forest, K-Nearest Neighbors and Support Vector Machine) could be used to create a screening test for the evaluation of election outcomes that can be understood and used by naïve as well as professional users. The rules are extracted to create a chart as a parametric screen test which help political or non-political professionals in forecasting the election outcomes at constituency wise.

The JKEPM development plan consists of two main phases. The first phase includes loading the attributes and applying the proposed models to the attributes of the Jammu and Kashmir election dataset and then the diagnostic rules are extracted and stored. Figure 6.10.1 shows the start screen of JKEPM where the user enters data on different parameters and the chances of election won or lost is calculated and displayed.

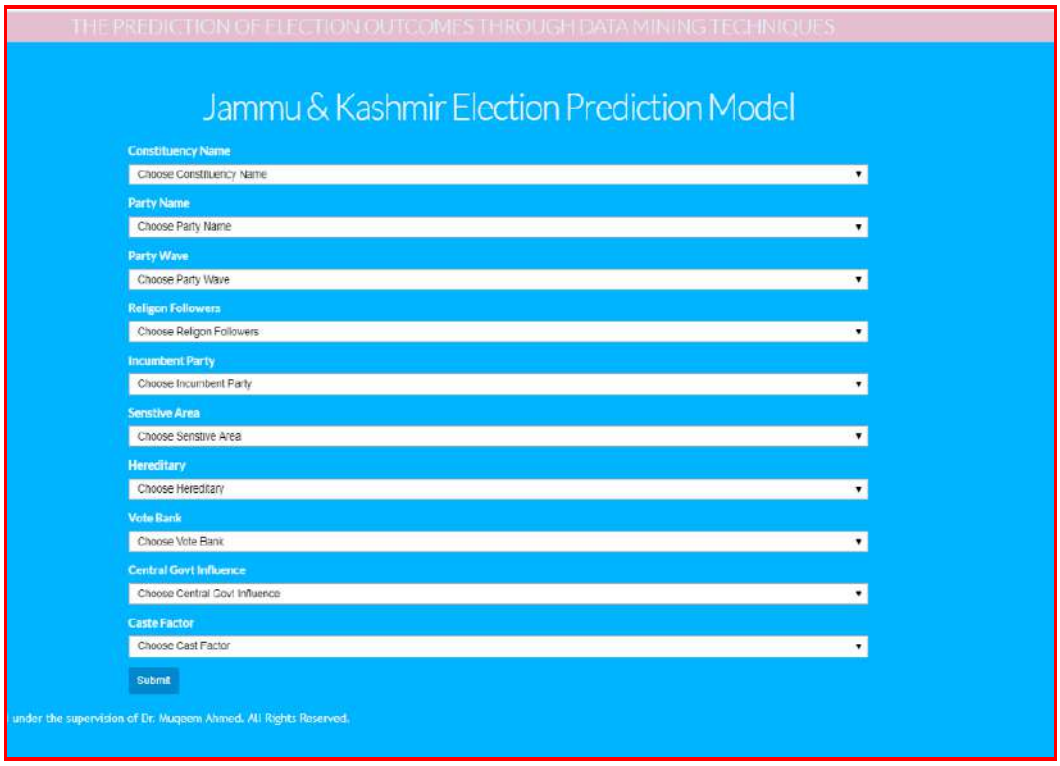

Figure 6.10.1: The Election Prediction Model Interface

In the second phase, the user clicks on different attributes shown on the model screen and enters the stored values; these attributes are used by the stored diagnostic rules to calculate the user's degree election forecasting which is displayed to the user. The JKEPM is implemented using Python Jupyter Notebook. The researcher set threshold value >=.5 which means 50%. Above it is Political parties or Independent candidates won the election and below it they lost the election. Figure 6.10.2 is an output result of the data entered by the user into different corresponding attribute values of the start screen; here JKEPM calculates the high degree of Election outcomes for a party at constituency wise for the entered data. The Figure 6.10.3 below is the second example of JKEPM model, in this case, a low chance of winning the election is calculated based on the entered data at the start screen. These examples demonstrate that JKEPM can act as a public level screening test.

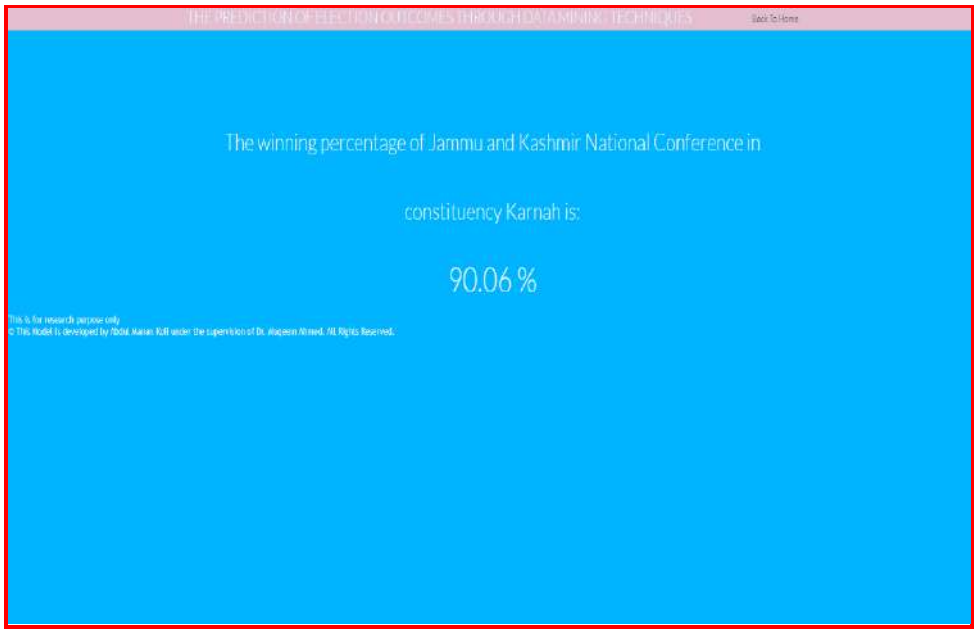

Figure 6.10.2: Election Prediction Evaluation Example

The simplicity of the user interface allows Political or non-political professional to identify the winning or losing chances by political parties or independent candidates constituency wise. The JKEPM is implemented on mobile as well as desktop applications.

Figure 6.10.3: Low-Chances of winning the Election Example

6.11 Summary

In this chapter, researcher introduced hyperparameter optimization techniques and their various types. Researcher optimized the proposed models to help Political or non- political users in accurately prediction of election outcomes. In this chapter, researcher also perform a comparison between the hyper-parameterized models and the ensemble models. Researcher built the proposed model on Jupyter Notebook web application and conduct various experiments and compute the TPR, TNR, Accuracy, Precision, Misclassification Rate and AUROC curve scores for each developed model and finally combine all the four models into one model by using ensemble technique viz soft voting. Then researcher applied statistical test i.e. t-paired test to find either model is significant or non-significant. Experimental results showed that Ensemble model is significant as it outperforms other proposed models.

This chapter discusses the development of the JKEPM (Jammu and Kashmir Election Prediction Model). The JKEPM acts as a public-level screening model that identifies the degree of losing or winning the election based upon most vital parameters for Jammu and Kashmir, thus providing general public as well as trained poll forecaster with an inexpensive, reliable screening of potential election outcomes. The optimal set of election outcomes rules are generated by these models and are evaluated and validated by the poll forecasters domain experts. Importantly, the JKEPM implementation results, as well as the extracted diagnostic rules, are indicative but not definitive as they are based on only the Jammu and Kashmir election dataset, because different areas have different socio-cultural habitat so the attributes selection may vary from area to area.

CHAPTER 7

7. Conclusion and Future Scope

This chapter outlines the research conclusions, discusses the research limitations, and illustrates the future research aspects. The Predictions of Election outcomes using data mining techniques for Jammu and Kashmir that too constituency-wise based upon parametric approach is a challenging task as different constituency have different culture, customs and habitats. Also, the people of Jammu division have different political perspective then the people of Kashmir division. Although newer forecasting methods based on Twitter and Facebook data or exit polls have now become the standard of Predictions but these modalities are costly and complex operational that restrain their use in rural areas of Jammu and Kashmir. This is due to the frequents internet shutdown, less connectivity, hence only a small fraction of population take part in poll surveys and express their views about elections on social media. The poll surveys expenditure, repeated exit polls and inaccurate predictions especially from Twitter and Facebook data have transformed election predictions an anxiety worldwide. Hence, researcher seek to build an appropriate model particularly for J&K that would facilitate the early prediction of constituency-wise election outcomes.

In this work, the researcher developed an election prediction models based upon parametric approach especially for the Jammu and Kashmir, that will not help only the poll forecaster to forecast the election outcomes, but also gives warning to political parties or independent candidates for working hard on the specified parameters (used for this model building) before contesting the elections. This model is useable for professional as well as general public for early and accurate predictions of election outcomes before actual results announced.

In this research work, researcher build the election prediction model based on the parametric approach using the Jupyter notebook web applications. Researcher applied Data mining techniques mainly classifications techniques like Decision tree, Random Forest, K-Nearest Neighbors, Support Vector Machine on the collected election data set and finally ensemble them into one Model using Soft voting, in order to find either the Political Parties or Independent candidates will have won the elections or not.

Researcher compute sensitivity, recognition rate, precision, and accuracy, AUROC score and misclassification score for each individual technique-based model and sample testing to check the accuracy and reliability of the developed election prediction model. Experimental results showed that the ensembled election prediction model outperforms other classification-based election prediction models on the hyperparameter settings with the sensitivity of 86%, the specificity of 92%, the accuracy of 89%, the precision of 91%, miss-classification rate of 10.1%, and AUROC score of 90%, as shown in chapter 6 table number 6.6. Finally, researcher run the statistical t-paired test to get the significance level of the model and got less than (0.05) p value for all technique-based models like (Decision tree, K-NN, Random Forest, Support Vector Machine). Applying this statistical t-paired test on our model, results in null hypothesis rejection, because of the statistical difference between ensemble model and other proposed election prediction models.

7.2 Research Limitations

There are some limitations to this research:

I. In this work, the researcher used few data mining techniques like Decision Tree, K-Nearest Neighbour, Random Forest, Support Vector Machine and finally ensemble technique in election result prediction.

II. The number of attributes can be enhanced with the addition of Development factor, vote swing, candidate's qualification, and GDP.

III. Testing the data mining techniques on the Jammu and Kashmir election dataset which contains a limited number of records.

IV. Model is limited to the past elections dataset with different types of election forecasting techniques and not social media data especially Twitter or Facebook data.

7.3 Future Work

For future enhancement, researcher would prefer to improve the proposed work in the following directions: Investigating the performance of the other robust techniques such as Machine Learning, Deep Learning, Genetic Algorithm, and Hybrid Ensemble techniques to develop more accurate election prediction models. The work can also be enhanced by incorporating the following steps in the near future.

i. Predicting the election outcomes at candidates' levels instead of constituency levels.

ii. Exploring the significance of adding other attributes like (Development factor, vote swing and candidate's qualification) on the performance of different data mining techniques in the forecasting of election outcomes.

iii. Using other large real datasets with numerous features, different state election datasets and a huge number of records.

iv. Testing the realistic usability and acceptability of the JKEPM among other poll forecaster and exit poll providers.

References

[1] C. Bjørnskov and M. Rode, "Regime Types and Regime Change: A New Dataset," *SSRN Electron. J.*, 2018.

[2] J. and K. Chief Electoral Officer, "Parliamentary Constituencies," *ceojammukashmir.nic.in (accessed on January 2018)*.

[3] Chowdhary, Rekha. Jammu and Kashmir: Politics of identity and separatism. Routledge, 2015.

[4] A. S. Bojang, "The Election System in India," *J. Polit. Sci. Public Aff.*, vol. 7, no. 1, p. 2, 2019.

[5] K. K. Hirji, "Discovering data mining," *ACM SIGKDD Explor. Newsl.*, vol. 1, no. 1, p. 44, Jun. 1999.

[6] M. Gohar, S. H. Ahmed, M. Khan, N. Guizani, A. Ahmad, and A. U. Rahman, "Big Data and Its Implementation Toward Future Smart Cities A Big Data Analytics Architecture for the Internet of Small Things," *IEEE Commun. Mag.*, vol. 56, no. February, pp. 128–133, 2018.

[7] Z. Lv, H. Song, P. Basanta-Val, A. Steed, and M. Jo, "Next-Generation Big Data Analytics: State of the Art, Challenges, and Future Research Topics," *IEEE Trans. Ind. Informatics*, vol. 13, no. 4, pp. 1891–1899, 2017.

[8] A. Mavragani and K. P. Tsagarakis, "Predicting referendum results in the Big Data Era," *J. Big Data*, vol. 6, no. 1, pp. 24–27, 2019.

[9] Jagdev, Gagandeep, and Amandeep Kaur. "Analyzing and scripting indian election strategies using big data via Apache Hadoop framework." *In 2016 5th International Conference on Wireless Networks and Embedded Systems (WECON)*, pp. 1-9. IEEE, 2016.

[10] "How Much Data Do We Create Every Day? The Mind-Blowing Stats Everyone Should Read." [Online]. Available: https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/#c1791f260ba9.

[Accessed: 03-Dec-2019].

[11] M. Sarnovsky, P. Bednar, and M. Smatana, “Big Data Processing and Analytics Platform Architecture for Process Industry Factories,” *Big Data Cogn. Comput.*, vol. 2, no. 1, p. 3, 2018.

[12] P. R. Rathi, P. A. S. Bhala, and S. G. Rathi, “Big Data Analytics for Social Network - The Base Study,” vol. 36, no. 9, pp. 467–470, 2016.

[13] Z. Chen and Z. Zhang, “What network topology can tell in election prediction,” *Discret. Math. Algorithms Appl.*, vol. 10, no. 02, p. 1850027, 2018.

[14] G. Jagdev and A. Kaur, “Excavating Big Data associated to Indian Elections Scenario via Apache Hadoop,” *Int. J. Adv. Res. Comput. Sci.*, vol. 7, no. 6, pp. 117–123, 2016.

[15] R. J. González, “Hacking the citizenry?: Personality profiling, ‘big data’ and the election of Donald Trump,” *Anthropol. Today*, vol. 33, no. 3, pp. 9–12, 2017.

[16] Jamie Bartlett; Josh Smith; Rose Acton, “The future of political campaigning,” *Demos*, vol. 3, no. 1, pp. 14–21, 2018.

[17] R. S. Erikson and C. Wlezien, “Markets vs. polls as election predictors: An historical assessment,” *Elect. Stud.*, vol. 31, no. 3, pp. 532–539, 2012.

[18] J. G. Geer, *From tea leaves to opinion polls : a theory of democratic leadership*. Columbia University Press, 1996.

[19] Stegmaier, Mary, and Helmut Norpoth. Election forecasting. Oxford University Press, 2013.

[20] D. Rothschild, “Forecasting Elections: Comparing prediction markets, polls, and their biases,” *Public Opin. Q.*, vol. 73, no. 5, pp. 895–916, 2009.

[21] A. Leigh and J. wolfers, “Competing Approaches to Forecasting Elections: Economic Models, Opinion Polling and Prediction Markets*,” *Econ. Rec.*, vol. 82, no. 258, pp. 325–340, Sep. 2006.

[22] A. J. Healy, M. Persson, and E. Snowberg, “Digging into the pocketbook: Evidence on economic voting from income registry data matched to a voter survey,” *Am. Polit. Sci. Rev.*, vol. 111, no. 4, pp. 771–785, Nov. 2017.

[23] C. Wlezien, “The myopic voter? The economy and US presidential elections,” *Elect. Stud.*, vol. 39, pp. 195–204, Sep. 2015.

[24] G. D. Whitten, “Methodological innovations in the study of election,” *Elect. Stud.*, vol. 39, pp. 179–180, Sep. 2015.

[25] F. Nooralahzadeh, V. Arunachalam, and C. G. Chiru, “2012 presidential elections on twitter - An analysis of how the us and french election were reflected in tweets,” in *Proceedings - 19th International Conference on Control Systems and Computer Science, CSCS 2013*, 2013, pp. 240–246.

[26] L. Guo, J. A. Rohde, and H. D. Wu, “Who is responsible for Twitter’s echo chamber problem? Evidence from 2016 U.S. election networks,” *Inf. Commun. Soc.*, vol. 0, no. 0, pp. 1–18, 2018.

[27] J. Davis, “Presidential campaigns and social networks: How Clinton and Trump used Facebook and Twitter during the 2016 election,” 2017.

[28] M. ANCU, “From Soundbite to Textbite: Election 2008 Comments on Twitter,” pp. 37–47, Jun. 2014.

[29] Gayo Avello, Daniel, Panagiotis T. Metaxas, and Eni Mustafaraj. "Limits of electoral predictions using twitter." *Proceedings of the fifth international AAAI conference on weblogs and social media. Association for the Advancement of Artificial Intelligence*, 2011.

[30] P. T. Metaxas, E. Mustafaraj, and D. Gayo-Avello, “How (Not) to predict elections,” *Proc. - 2011 IEEE Int. Conf. Privacy, Secur. Risk Trust IEEE Int. Conf. Soc. Comput. PASSAT/SocialCom 2011*, pp. 165–171, 2011.

[31] M. Zolghadr, S. A. A. T. A. Niaki, and S. A. A. T. A. Niaki, “Modeling and forecasting US presidential election using learning algorithms,” *J. Ind. Eng. Int.*, vol. 14, no. 3, pp. 491–500, 2017.

[32] H. Singh, G. Singh, and N. Bhatia, “Fuzzy cognitive maps based election results prediction system,” *Int. J. Comput. Technol.*, vol. 7, no. 1, pp. 483–492, 2013.

[33] P. Hummel and D. Rothschild, “Fundamental models for forecasting elections at the state level,” *Elect. Stud.*, vol. 35, pp. 123–139, 2014.

[34] W. Arulampalam, S. Dasgupta, A. Dhillon, and B. Dutta, “Electoral goals and center-state transfers: A theoretical model and empirical evidence from India,” *J. Dev. Econ.*, vol. 88, no. 1, pp. 103–119, 2009.

[35] Singh, Harmanjit, Gurdev Singh, and Nitin Bhatia. "Election results prediction system based on fuzzy logic." *International Journal of computer applications* 53.9 2012.

[36] Alam, Mohd Manjur, Md MezbahUddin, and Shamsun Nahar Shoma. "Election Result Prediction System using Hidden Markov Model [HMM]." *International Journal of Computer Applications* 975 2015: 8887.

[37] Gill, G. S. "Election result forecasting using two layer perceptron network." *Journal of Theoretical and Applied Information Technology* 4.11 (2005): 144-146.

[38] Sawant, Omkar, et al. "Election Analysis and Prediction Using Big Data Analytics." *International Journal on Recent and Innovation Trends in Computing and Communication* 5.2: 107-111.

[39] “J&K communication blockade eased, 280 e-terminals set up - The Economic Times.” [Online]. Available: https://economictimes.indiatimes.com/news/politics-and-nation/jk-communication-blockade-eased-280-e-terminals-set-up/articleshow/72192733.cms. [Accessed: 23-Nov-2019].

[40] “Journalists protest 100 days of internet gag in Kashmir.” [Online]. Available: https://www.indiatoday.in/india/story/journalists-protest-100-days-of-internet-gag-in-kashmir-1618315-2019-11-13. [Accessed: 23-Nov-2019].

[41] “After 100 days of the shutdown in Kashmir, not much has changed.” [Online]. Available: https://www.telegraphindia.com/opinion/after-100-days-of-the-shutdown-in-kashmir-not-much-has-changed/cid/1718980. [Accessed: 23-Nov-2019].

[42] R. S. Erikson and C. Wlezien, “The economy and the presidential vote: What leading indicators reveal well in advance,” *Int. J. Forecast.*, vol. 24, no. 2, pp. 218–226, 2008.

[43] J. M. Pavía, B. Larraz, and J. M. Montero, “Election forecasts using spatiotemporal models,” *J. Am. Stat. Assoc.*, vol. 103, no. 483, pp. 1050–1059, 2008.

[44] H. Norpoth and T. Gschwend, “The chancellor model: Forecasting German elections,” *Int. J. Forecast.*, vol. 26, no. 1, pp. 42–53, 2010.

[45] M. S. Lewis-Beck and R. Nadeau, “Economic voting theory: Testing new dimensions,” *Elect. Stud.*, vol. 30, no. 2, pp. 288–294, 2011.

[46] E. Toros, “Forecasting elections in Turkey,” *Int. J. Forecast.*, vol. 27, no. 4, pp. 1248–1258, 2011.

[47] Singh, Harmanjit, Gurdev Singh, and Nitin Bhatia. "Election results prediction system based on fuzzy logic." *International Journal of computer applications* 53.9 2012.

[48] Kodinariya, T. M., and Ravi Seta. "Visual data mining in indian election system." *International Journal on Computer Science and Engineering* 4.7 (1323) 2012.

[49] S. Dahlberg, “Does context matter - The impact of electoral systems, political parties and individual characteristics on voters’ perceptions of party positions,” *Elect. Stud.*, vol. 32, no. 4, pp. 670–683, 2013.

[50] Y. Wang, Y. Feng, J. Luo, and X. Zhang, “Voting with feet: Who are leaving Hillary Clinton and Donald Trump?,” *Proc. - 2016 IEEE Int. Symp. Multimedia, ISM 2016*, pp. 71–76, 2017.

[51] M. Zolghadr, S. A. A. Niaki, and S. T. A. Niaki, “Modeling and forecasting US presidential election using learning algorithms,” *J. Ind. Eng. Int.*, vol. 14, no. 3, pp. 491–500, 2017.

[52] M. D. Conover, B. Gonçalves, J. Ratkiewicz, A. Flammini, and F. Menczer, “Predicting the political alignment of twitter users,” *Proc. - 2011 IEEE Int. Conf. Privacy, Secur. Risk Trust IEEE Int. Conf. Soc. Comput. PASSAT/SocialCom 2011*, pp. 192–199, 2011.

[53] Shi, Lei, Neeraj Agarwal, Ankur Agrawal, Rahul Garg, and Jacob Spoelstra. "Predicting US primary elections with Twitter." *URL: http://snap. stanford. edu/social2012/papers/shi. pdf 2012.*

[54] T. Mahmood, T. Iqbal, F. Amin, W. Lohanna, and A. Mustafa, “Mining Twitter big data to predict 2013 Pakistan election winner,” *2013 16th Int. Multi Top. Conf. INMIC 2013*, pp. 49–54, 2013.

[55] S. E. Polykalas, G. N. Prezerakos, and A. Konidaris, “An algorithm based on Google Trends’ data for future prediction. Case study: German elections,” *IEEE Int. Symp. Signal Process. Inf. Technol.*, pp. 000069–000073, 2013.

[56] F. Pimenta, D. Obradović, and A. Dengel, “A comparative study of social media prediction potential in the 2012 U.S. Republican presidential preelections,” *Proc. - 2013 IEEE 3rd Int. Conf. Cloud Green Comput. CGC 2013 2013 IEEE 3rd Int. Conf. Soc. Comput. Its Appl. SCA 2013*, pp. 226–232, 2013.

[57] M. Song, M. C. Kim, and Y. K. Jeong, “Analyzing the political landscape of 2012 korean presidential election in twitter,” *IEEE Intell. Syst.*, vol. 29, no. 2, pp. 18–26, 2014.

[58] A. Ceron, L. Curini, and S. M. Iacus, “Using Sentiment Analysis to Monitor Electoral

Campaigns: Method Matters—Evidence From the United States and Italy," *Soc. Sci. Comput. Rev.*, vol. 33, no. 1, pp. 3–20, 2015.

[59] M. Anjaria and R. M. R. Guddeti, "A novel sentiment analysis of social networks using supervised learning," *Soc. Netw. Anal. Min.*, vol. 4, no. 1, pp. 1–15, 2014.

[60] N. A. Gayatri Wani, "A Survey on Impact of Social Media on Election System," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 6, pp. 7363–7366, 2014.

[61] Y. Nam, Y. O. Lee, and H. W. Park, "Measuring web ecology by Facebook, Twitter, blogs and online news: 2012 general election in South Korea," *Qual. Quant.*, vol. 49, no. 2, pp. 675–689, 2014.

[62] V. Kagan, A. Stevens, and V. S. Subrahmanian, "Using twitter sentiment to forecast the 2013 Pakistani election and the 2014 Indian election," *IEEE Intell. Syst.*, vol. 30, no. 1, pp. 2–5, 2015.

[63] Ullah, Rahman, Abdur Rashid Khan, and Muhammad Irfan. "Forecasting political weather for pakistan local government election-using opinion mining." *Journal of Multidisciplinary Engineering Science and Technology* 2.12, 2015.

[64] B. A. Conway, K. Kenski, and D. Wang, "The Rise of Twitter in the Political Campaign: Searching for Intermedia Agenda-Setting Effects in the Presidential Primary," *J. Comput. Commun.*, vol. 20, no. 4, pp. 363–380, 2015.

[65] A. Tsakalidis, S. Papadopoulos, A. I. Cristea, and Y. Kompatsiaris, "Predicting Elections for Multiple Countries Using Twitter and Polls," *IEEE Intell. Syst.*, vol. 30, no. 2, pp. 10–17, 2015.

[66] K. Singhal, B. Agrawal, and N. Mittal, "Modeling indian general elections: Sentiment analysis of political twitter data," *Adv. Intell. Syst. Comput.*, vol. 339, pp. 469–477, 2015.

[67] V. D. Jadhav and S. N. Deshmukh, "Web Site : www.ijettcs.org Email : editor@ijettcs.org with Twitter Using Bayesian Classifier AND TRAINING," vol. 5, no. 2, pp. 2015–2017, 2016.

[68] K. Ismail, “Data Mining for Social Media Analysis:Using Twitter to Predict the 2016 US Presidential Election,” *Int. J. Sci. Eng. Res.*, vol. 7, no. 10, pp. 1972–1980, 2016.

[69] D. J. S. Oliveira, P. H. de S. Bermejo, and P. A. dos Santos, “Can social media reveal the preferences of voters? A comparison between sentiment analysis and traditional opinion polls,” *J. Inf. Technol. Polit.*, vol. 14, no. 1, pp. 34–45, 2017.

[70] M. H. Wang and C. L. Lei, “Boosting election prediction accuracy by crowd wisdom on social forums,” *2016 13th IEEE Annu. Consum. Commun. Netw. Conf. CCNC 2016*, pp. 348–353, 2016.

[71] P. Sharma and T. S. Moh, “Prediction of Indian election using sentiment analysis on Hindi Twitter,” *Proc. - 2016 IEEE Int. Conf. Big Data, Big Data 2016*, pp. 1966–1971, 2016.

[72] M. Vergeer, “Adopting, Networking, and Communicating on Twitter: A Cross-National Comparative Analysis,” *Soc. Sci. Comput. Rev.*, vol. 35, no. 6, pp. 698–712, 2017.

[73] J. Mellon and C. Prosser, “Twitter and Facebook are not representative of the general population: Political attitudes and demographics of british social media users,” *Res. Polit.*, vol. 4, no. 3, pp. 1–9, 2017.

[74] H. Ranjan and M. Reza, “Predictive Analytics on Gujrat Assembly Election 2017 Using Twitter Sentiment Analysis based on Word2Vec Model,” no. December, pp. 2–4, 2017.

[75] V. K. Jain and S. Kumar, “Towards prediction of election outcomes using social media,” *Int. J. Intell. Syst. Appl.*, vol. 9, no. 12, pp. 20–28, Dec. 2017.

[76] N. S, “Indian Election trend prediction using improved competitive vector regression model,” *Int. J. Recent Trends Eng. Res.*, vol. 3, no. 5, pp. 388–392, 2017.

[77] S. Goyal, “Review Paper on Sentiment Analysis of Twitter Data Using Text Mining and Hybrid Classification Approach,” *Int. J. Eng. Dev. Res.*, vol. 5, no. 2, pp. 2321–9939, 2017.

[78] M. Safiullah, P. Pathak, S. Singh, and A. Anshul, “Social media as an upcoming tool for political marketing effectiveness,” *Asia Pacific Manag. Rev.*, vol. 22, no. 1, pp. 10–15, 2017.

[79] Dubey, Gaurav, Shilpi Chawla, and Kirandeep Kaur. "Social media opinion analysis for indian political diplomats." 2017 7th *International Conference on Cloud Computing, Data Science & Engineering-Confluence.* IEEE, 2017.

[80] A. Hernandez-Suarez *et al.*, “Predicting political mood tendencies based on Twitter data,” *Proc. - 2017 5th Int. Work. Biometrics Forensics, IWBF 2017*, pp. 1–6, 2017.

[81] P. Singh and R. S. Sawhney, “Influence of Twitter on prediction of election results,” *Adv. Intell. Syst. Comput.*, vol. 564, pp. 665–673, 2018.

[82] K. P. Neetu Narwal, “A Case Study On Delhi MCD Election Prediction Using Social Media Analtics,” *ijarcs*, vol. 9, pp. 504–510, 2018.

[83] A. Hasan, S. Moin, A. Karim, and S. Shamshirband, “Machine Learning-Based Sentiment Analysis for Twitter Accounts,” *Math. Comput. Appl.*, vol. 23, no. 1, p. 11, 2018.

[84] I. El Alaoui, Y. Gahi, R. Messoussi, Y. Chaabi, A. Todoskoff, and A. Kobi, “A novel adaptable approach for sentiment analysis on big social data,” *J. Big Data*, vol. 5, no. 1, 2018.

[85] Mazumder, Pritom, Navid Anjum Chowdhury, Moh Anwar-Ul-Azim Bhuiya, Shabbir Haque Akash, and Rashedur M. Rahman. "A Fuzzy Logic Approach to Predict the Popularity of a Presidential Candidate." *In Modern Approaches for Intelligent Information and Database Systems*, pp. 63-74. Springer, Cham, 2018.

[86] S. M. Thampi, A. Gelbukh, and J. Mukhopadhyay, “Advances in signal processing and intelligent recognition systems,” *Adv. Intell. Syst. Comput.*, vol. 264, 2014.

[87] Gilmartin, Raymond, and Sean Roelofs. "Active Polling: Improving Presidential Election Predictions by Geographically Targeting Polls with active Learning." *Methods* 2.2, 2019.

[88] Gill, G. S. "Election result forecasting using two layer perceptron network." *Journal of Theoretical and Applied Information Technology* 4.11 (2005): 144-146.

[89] J. E. Campbell and M. S. Lewis-Beck, “US presidential election forecasting: An introduction,” *Int. J. Forecast.*, vol. 24, no. 2, pp. 189–192, 2008.

[90] G. R. Murray, C. Riley, and A. Scime, “Pre-election polling: Identifying likely voters using iterative expert data mining,” *Public Opin. Q.*, vol. 73, no. 1, pp. 159–171, 2009.

[91] S. Munzert, “Forecasting elections at the constituency level: A correction–combination procedure,” *Int. J. Forecast.*, vol. 33, no. 2, pp. 467–481, 2017.

[92] R. Dassonneville, M. S. Lewis-Beck, and P. Mongrain, “Forecasting Dutch elections: An initial model from the march 2017 legislative contests,” *Res. Polit.*, vol. 4, no. 3, 2017.

[93] G. M. Draper and R. F. Riesenfeld, “Who votes for what? A visual query language for opinion data,” *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 6, pp. 1197–1204, 2008.

[94] S. E. Rigdon, S. H. Jacobson, W. K. Tam Cho, E. C. Sewell, and C. J. Rigdon, “A Bayesian prediction model for the U.S. presidential election,” *Am. Polit. Res.*, vol. 37, no. 4, pp. 700–724, 2009.

[95] C. H. De Vreese, “Second-rate election campaigning? An analysis of campaign styles in European parliamentary elections,” *J. Polit. Mark.*, vol. 8, no. 1, pp. 7–19, 2009.

[96] J. . and M. W. Kesten C. Greene and J. Scott Armstrong and Randall J. Jones, “Predicting Elections from Politicians%99 Faces,” vol. 22, no. 4, 2010.

[97] M. S. Lewis-Beck and M. Stegmaier, “Citizen forecasting: Can UK voters see the future?,” *Elect. Stud.*, vol. 30, no. 2, pp. 264–268, 2011.

[98] R. Ford, W. Jennings, and W. Somerville, “Public opinion, responsiveness and constraint: Britain’s three immigration policy regimes,” *J. Ethn. Migr. Stud.*, vol. 41, no. 9, pp. 1391–1411,

2015.

[99] A. Khatua, A. Khatua, K. Ghosh, and N. Chaki, “Can #Twitter-Trends predict election results? Evidence from 2014 Indian general election,” *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, vol. 2015-March, pp. 1676–1685, 2015.

[100] C. Rallings, M. Thrasher, and G. Borisyuk, “Forecasting the 2015 general election using aggregate local election data,” *Elect. Stud.*, vol. 41, pp. 279–282, 2016.

[101] Q. You, L. Cao, Y. Cong, X. Zhang, and J. Luo, “A Multifaceted Approach to Social Multimedia-Based Prediction of Elections,” *IEEE Trans. Multimed.*, vol. 17, no. 12, pp. 2271–2280, 2015.

[102] Z. Xie, G. Liu, J. Wu, L. Wang, and C. Liu, “Wisdom of fusion: Prediction of 2016 Taiwan election with heterogeneous big data,” *2016 13th Int. Conf. Serv. Syst. Serv. Manag. ICSSSM 2016*, 2016.

[103] Z. Xu and W. Liu, “Apriori-based prediction of the multi seat elections,” *2017 2nd IEEE Int. Conf. Comput. Intell. Appl. ICCIA 2017*, vol. 2017-Janua, pp. 214–218, 2017.

[104] D. Sathiaraj, W. M. Cassidy, and E. Rohli, “Improving Predictive Accuracy in Elections,” *Big Data*, vol. 5, no. 4, pp. 325–336, 2017.

[105] A. C. B. Garcia, W. Silva, and L. Correia, “The PredNews forecasting model,” *Proc. 19th Annu. Int. Conf. Digit. Gov. Res. Gov. Data Age - dgo ’18*, pp. 1–6, 2018.

[106] B. Sharar and M. Abd-El-Barr, “Citizens’ perspective on the impact of social media on politics in Kuwait,” in *2018 International Conference on Computing Sciences and Engineering, ICCSE 2018 - Proceedings*, 2018, pp. 1–6.

[107] P. Pranay, “Role of Big Data In US Presidential Election,” Mount Royal University, 2018.

[108] S. Srivastava, “Weka: A Tool for Data preprocessing, Classification, Ensemble, Clustering and

Association Rule Mining," *Int. J. Comput. Appl.*, vol. 88, no. 10, pp. 26–29, 2014.

[109] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software," *ACM SIGKDD Explor. Newsl.*, vol. 11, no. 1, p. 10, Nov. 2009.

[110] A. Jović, K. Brkić, and N. Bogunović, "An overview of free software tools for general data mining," in *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2014 - Proceedings*, 2014, pp. 1112–1117.

[111] P. Ristoski, C. Bizer, and H. Paulheim, "Mining the Web of Linked Data with RapidMiner," *J. Web Semant.*, vol. 35, pp. 142–151, Dec. 2015.

[112] P. Tripathi, S. K. Vishwakarma, and A. Lala, "Sentiment Analysis of English Tweets Using Rapid Miner," in *Proceedings - 2015 International Conference on Computational Intelligence and Communication Networks, CICN 2015*, 2016, pp. 668–672.

[113] A. Naik and L. Samant, "Correlation Review of Classification Algorithm Using Data Mining Tool: WEKA, Rapidminer, Tanagra, Orange and Knime," in *Procedia Computer Science*, 2016, vol. 85, pp. 662–668.

[114] J. Demsar and B. Zupan, "Orange: data mining fruitful and fun," *Informatica.*, vol. 37, no. 1, p. 55, 2013.

[115] S. Sivanandam, S. Sumathi, and S. Deepa, *Introduction to fuzzy logic using MATLAB*. Springer Berlin Heidelberg 2007, 2007.

[116] Marzjarani, Morteza. "Exploratory Data Analysis With MATLAB: by Wendy L. Martinez, Angel R. Martinez, and Jeffery L. Solka. Boca Raton, FL: CRC Press, 2017, xxv+ 590 pp., 100.00(Hardback), 46.36 (eBook), ISBN: 13: 978-1-498-7606-6." (2019): 565-566.

[117] R. M. Rahman and F. Afroz, "Comparison of Various Classification Techniques Using

Different Data Mining Tools for Diabetes Diagnosis," *J. Softw. Eng. Appl.*, no. 6, pp. 85–97, 2013.

[118] J. Y. Yuxing Yan, *Hands-On Data Science with Anaconda: Utilize the right mix of tools to ... - Yuxing Yan, James Yan - Google Books*. Packt Publishing Ltd., 2018.

[119] W. Mohammad, "Python Anaconda Tutorial | Getting Started With Anaconda | Edureka," 2019. [Online]. Available: https://www.edureka.co/blog/python-anaconda-tutorial/. [Accessed: 09-Feb-2020].

[120] P. Vogel, T. Klooster, V. Andrikopoulos, and M. Lungu, "A Low-Effort Analytics Platform for Visualizing Evolving Flask-Based Python Web Services," in *Proceedings - 2017 IEEE Working Conference on Software Visualization, VISSOFT 2017*, 2017, vol. 2017-October, pp. 109–113.

[121] F. Armash Aslam, H. Nabeel Mohammed Jummal Musab Mohd Munir Murade Aaraf Gulamgaus, and P. S. Lokhande Assistant Professor, "Efficient Way Of Web Development Using Python And Flask," *Int. J. Adv. Res. Comput. Sci.*, vol. 6, no. 2, 2015.

[122] B. H. Lee, E. K. Dewi, and M. F. Wajdi, "Data security in cloud computing using AES under HEROKU cloud," in *2018 27th Wireless and Optical Communication Conference, WOCC 2018*, 2018, pp. 1–5.

[123] I. Kavakiotis, O. Tsave, A. Salifoglou, N. Maglaveras, I. Vlahavas, and I. Chouvarda, "Machine Learning and Data Mining Methods in Diabetes Research," *Comput. Struct. Biotechnol. J.*, vol. 15, pp. 104–116, 2017.

[124] C. Geiß *et al.*, "Estimation of seismic building structural types using multi-sensor remote sensing and machine learning techniques," *ISPRS J. Photogramm. Remote Sens.*, vol. 104, pp. 175–188, Jun. 2015.

[125] Abu-Nimeh, Saeed, et al. "A comparison of machine learning techniques for phishing detection." *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*. 2007.

[126] G. M. Souza, T. A. Catuchi, S. C. Bertolli, and R. P. Soratto, "Soybean under water deficit: physiological and Yield Responses," *A Compr. Surv. Int. Soybean Res. - Genet. Physiol. Agron. Nitrogen Relationships*, p. 624, 2013.

[127] Zhu, Xiaojin Jerry. *Semi-supervised learning literature survey*. University of Wisconsin-Madison Department of Computer Sciences, 2005.

[128] T. C. Smith and E. Frank, "Introducing machine learning concepts with WEKA," in *Methods in Molecular Biology*, vol. 1418, Humana Press Inc., 2016, pp. 353–378.

[129] R. S. Sutton and A. G. Barto, "Introduction to Reinforcement Learning 1st," *Handb. Neural Comput.*, p. 342, 1998.

[130] Kuhn, Max, and Kjell Johnson. *Applied predictive modeling*. Vol. 26. New York: Springer, 2013.

[131] A. S. Ghareb, A. A. Bakar, and A. R. Hamdan, "Hybrid feature selection based on enhanced genetic algorithm for text categorization," *Expert Syst. Appl.*, vol. 49, pp. 31–47, 2016.

[132] I. H. Lee, G. H. Lushington, and M. Visvanathan, "A filter-based feature selection approach for identifying potential biomarkers for lung cancer," *J. Clin. Bioinforma.*, vol. 1, no. 1, pp. 1–8, 2011.

[133] K. Pavya and D. B. Srinivasan, "Feature Selection Techniques in Data Mining: A Study," *Int. J. Sci. Dev. Res.*, vol. 2, no. 6, pp. 594–598, 2017.

[134] A. E. Isabelle Guyon, "An Introduction to Variable and Feature Selection Isabelle," *J. of Machine Learn. Res. 3*, no. 2, pp. 1157–1182, 2003.

[135] Z. Yin and J. Zhang, “Operator functional state classification using least-square support vector machine based recursive feature elimination technique,” *Comput. Methods Programs Biomed.*, vol. 113, no. 1, pp. 101–115, 2014.

[136] K. Belkhayat Abou Omar, G. Bruner, and Y. Wang Roger Wattenhofer, “XGBoost and LGBM for Porto Seguro’s Kaggle challenge: A comparison Semester Project,” 2018.

[137] K. Yan and D. Zhang, “Feature selection and analysis on correlated gas sensor data with recursive feature elimination,” *Sensors Actuators, B Chem.*, vol. 212, pp. 353–363, 2015.

[138] L. Meier, S. Van De Geer, and H. Zou, “The group lasso for logistic regression - Meier - 2008 - Journal of the Royal Statistical Society: Series B (Statistical Methodology) - Wiley Online Library,” *J. Am. Stat. Assoc.*, vol. 101, no. 476, pp. 1418–1429, 2006.

[139] Fonti, Valeria, and Eduard Belitser. "Feature selection using lasso." *VU Amsterdam Research Paper in Business Analytics* 30 (2017): 1-25

[140] Pang-Ning Tan Michael Steinbach Vipin Kumar, *Introduction to Data Mining*, vol. 19. 2006.

[141] Y. Baştanlar and M. Özuysal, “Introduction to machine learning,” *Methods Mol. Biol.*, vol. 1107, pp. 105–128, 2014.

[142] Y. Xia, C. Liu, Y. Y. Li, and N. Liu, “A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring,” *Expert Syst. Appl.*, vol. 78, pp. 225–241, Jul. 2017.

[143] S. Drazin, “Decision Tree Analysis using Weka,” *Mach. Learn. II, Univ. Miami*, pp. 1–3, 2010.

[144] W. Alawad, M. Zohdy, and D. Debnath, “Tuning Hyperparameters of Decision Tree Classifiers Using Computationally Efficient Schemes,” *Proc. - 2018 1st IEEE Int. Conf. Artif. Intell. Knowl. Eng. AIKE 2018*, pp. 168–169, 2018.

[145] E. Jyoti, E. Amandeep, S. Walia, M. Tech, and C. Science, “‘ A Review on Recommendation System and Web Usage Data Mining using K-Nearest Neighbor (KNN) method ,’” *Int. Res.*

J. Eng. Technol., vol. 4, no. 4, pp. 2931–2934, 2017.

[146] D. A. Adeniyi, Z. Wei, and Y. Yongquan, “Automated web usage data mining and recommendation system using K-Nearest Neighbor (KNN) classification method,” *Appl. Comput. Informatics*, vol. 12, no. 1, pp. 90–108, 2016.

[147] Q. Kuang and L. Zhao, “A practical GPU based kNN algorithm,” *Int. Symp. Comput. Sci. Comput. Technol.*, vol. 7, no. 3, pp. 151–155, 2009.

[148] S. Yu and S. Kak, “A survey of prediction using social media,” *arXiv Prepr. arXiv1203.1647*, pp. 1–20, 2012.

[149] Y. Zhang, G. Cao, B. Wang, and X. Li, “A novel ensemble method for k-nearest neighbor,” *Pattern Recognit.*, vol. 85, pp. 13–25, Jan. 2019.

[150] S. B. Kotsiantis, “Supervised Machine Learning: A Review of Classification Techniques,” *Informatica*, vol. 31, pp. 249–268, 2007.

[151] Jiao, Licheng, Liefeng Bo, and Ling Wang. "Fast sparse approximation for least squares support vector machine." *IEEE Transactions on Neural Networks* 18.3 (2007): 685-697.

[152] Wu, Xindong, et al. "Top 10 algorithms in data mining." *Knowledge and information systems* 14.1 (2008): 1-37.

[153] J. Nayak, B. Naik, and H. S. Behera, “A Comprehensive Survey on Support Vector Machine in Data Mining Tasks: Applications & Challenges,” *Int. J. Database Theory Appl.*, vol. 8, no. 1, pp. 169–186, 2015.

[154] E. Juliussen, “Telematics: Status and future perspectives,” *Intell. Transp. Soc. Am. - 12th World Congr. Intell. Transp. Syst. 2005*, vol. 8, no. 2008, pp. 4618–4630, 2009.

[155] N. Lay and A. Barbu, “Supervised aggregation of classifiers using Artificial Prediction Markets,” *ICML 2010 - Proceedings, 27th Int. Conf. Mach. Learn.*, pp. 591–598, 2010.

[156] M. Pal, “Random forest classifier for remote sensing classification,” *Int. J. Remote Sens.*, vol. 26, no. 1, pp. 217–222, 2005.

[157] T. R. Patil and M. . Sherekar, “Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification,” *Int. J. Comput. Sci. Appl.*, vol. 6, no. 2, pp. 256–261, 2013.

[158] N. D. Marom, L. Rokach, and A. Shmilovici, “Using the confusion matrix for improving ensemble classifiers,” in *2010 IEEE 26th Convention of Electrical and Electronics Engineers in Israel, IEEEI 2010*, 2010, pp. 555–559.

[159] Y. Xing, J. Wang, Z. Zhao, and Y. Gao, “Combination data mining methods with new medical data to predicting outcome of Coronary Heart Disease,” *Int. Conf. Converg. Inf. Technol. Comb.*, vol. 41, no. 3, pp. 1–6, 2007.

[160] N. V Chawla, “Data Mining and Knowledge Discovery Handbook,” *Data Min. Knowl. Discov. Handb.*, 2010.

[161] N. D. Rubinstein, I. Mayrose, and T. Pupko, “A machine-learning approach for predicting B-cell epitopes,” *Mol. Immunol.*, vol. 46, no. 5, pp. 840–847, 2009.

[162] R. B. Domingues, G. B. Peres, F. B. Moura-Leite, and C. S. Soares, “Quantitative and qualitative evaluation of IgG synthesis in 8,947 cerebrospinal fluid samples,” *J. Bras. Patol. e Med. Lab.*, vol. 56, Jan. 2020.

[163] C. Bergmeir and J. M. Benítez, “On the use of cross-validation for time series predictor evaluation,” *Inf. Sci. (Ny).*, vol. 191, pp. 192–213, May 2012.

[164] M. Aoshima and K. Yata, “A distance-based, misclassification rate adjusted classifier for multiclass, high-dimensional data,” *Ann. Inst. Stat. Math.*, vol. 66, no. 5, pp. 983–1010, 2014.

[165] Liu, Xu-Ying, and Zhi-Hua Zhou. "Ensemble methods for class imbalance learning." *Imbalanced Learning: Foundations, Algorithms and Applications* (2013): 61-82.

[166] H. G. Ayad and M. S. Kamel, “On voting-based consensus of cluster ensembles,” *Pattern Recognit.*, vol. 43, no. 5, pp. 1943–1953, 2010.

[167] M. Ali-Fauzi, “Automatic complaint classification system using classifier ensembles,” *Telfor J.*, vol. 10, no. 2, pp. 123–128, 2018.

[168] Zhou, Qimin, and Hao Wu. "NLP at IEST 2018: BiLSTM-attention and LSTM-attention via soft voting in emotion classification." *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. 2018.

[169] Shukla, Sanyam, and R. N. Yadav. "Unweighted class specific soft voting based ensemble of extreme learning machine and its variant." *International Journal of Computer Science and Information Security* 13.3 (2015): 59.

[170] J. Menke and T. R. Martinez, “Using permutations instead of student’s distribution for p-values in paired-difference algorithm comparisons,” *IEEE Int. Conf. Neural Networks - Conf. Proc.*, vol. 2, pp. 1331–1335, 2004.

[171] Azevedo, Ana Isabel Rojão Lourenço, and Manuel Filipe Santos. "KDD, SEMMA and CRISP-DM: a parallel overview." *IADS-DM* (2008).

[172] Z. Bošnjak, O. Grljević, and S. Bošnjak, “CRISP-DM as a framework for discovering knowledge in small and medium sized enterprises’ data,” *Proc. - 2009 5th Int. Symp. Appl. Comput. Intell. Informatics, SACI 2009*, no. 114, pp. 509–514, 2009.

[173] R. Wirth, “CRISP-DM : Towards a Standard Process Model for Data Mining,” *Proc. Fourth Int. Conf. Pract. Appl. Knowl. Discov. Data Min.*, no. 24959, pp. 29–39, 2000.

[174] R. Chowdhary, “Electoral Politics in the Context of Separatism and Political Divergence: An Analysis of 2009 Parliamentary elections in Jammu & Kashmir,” *South Asia Multidiscip. Acad. J.*, no. 3, pp. 0–26, 2009.

[175] P. Rebentrost, M. Mohseni, and S. Lloyd, “Quantum support vector machine for big data classification,” *Phys. Rev. Lett.*, vol. 113, no. 3, pp. 1–5, 2014.

[176] L. Zhang, W. Zhou, and L. Jiao, “Wavelet Support Vector Machine,” *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 34, no. 1, pp. 34–39, 2004.

[177] Dani Yogatama Gideon Mann, “Efficient Transfer Learning Method for Automatic Hyperparameter Tuning Dani,” *jmlr.org*, vol. 7, no. 2, pp. 48–50, 2014.

[178] Koch, Patrick, et al. "Automated hyperparameter tuning for effective machine learning." *Proceedings of the SAS Global Forum 2017 Conference*. Cary, NC: SAS Institute Inc., 2017.

[179] R. Bardenet, M. Brendel, B. Kégl, and M. Sebag, “Collaborative hyperparameter tuning,” *30th Int. Conf. Mach. Learn. ICML 2013*, vol. 28, no. PART 2, pp. 858–866, 2013.

[180] J. Bergstra, B. Komer, C. Eliasmith, D. Yamins, and D. D. Cox, “Hyperopt: A Python library for model selection and hyperparameter optimization,” *Comput. Sci. Discov.*, vol. 8, no. 1, 2015.

[181] M. Claesen, J. Simm, D. Popovic, and B. L. R. De Moor, “Hyperparameter tuning in Python using Optunity,” *Proc. Int. Work. Tech. Comput. Mach. Learn. Math. Eng.*, no. September, pp. 6–7, 2014.

[182] Feurer, Matthias, and Frank Hutter. "Hyperparameter optimization." *Automated Machine Learning*. Springer, Cham, 2019. 3-33.

[183] “4. Hyperparameter Tuning - Evaluating Machine Learning Models [Book].” [Online]. Available:https://www.oreilly.com/library/view/evaluating-machine learning /978149204 8756 /ch04.html. [Accessed: 08-Sep-2020].

[184] R. Kohavi and G. H. John, “Automatic Parameter Selection by Minimizing Estimated Error,”

Mach. Learn. Proc. 1995, no. November 2016, pp. 304–312, 1995.

[185] Claesen, Marc, and Bart De Moor. "Hyperparameter search in machine learning." *arXiv preprint arXiv:1502.02127* (2015).

[186] Montgomery, Douglas C. *Design and analysis of experiments*. John wiley & sons, 2017.

[187] Y. Bengio, "Gradient-based optimization of hyperparameters," *Neural Comput.*, vol. 12, no. 8, pp. 1889–1900, 2000.

[188] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.

[189] F. Hutter, H. Hoos, and K. Leyton-Brown, "An efficient approach for assessing hyperparameter importance," *31st Int. Conf. Mach. Learn. ICML 2014*, vol. 2, pp. 1130–1144, 2014.

[190] A. Snoek, Larochelle, "Practical Bayesian Optimization of Machine Learning Algorithms," *Relig. Arts*, vol. 17, no. 1–2, pp. 57–73, 2013.

[191] Z. Wang, M. Zoghiy, F. Hutterz, D. Matheson, and N. De Freitas, "Bayesian optimization in high dimensions via random embeddings," *IJCAI Int. Jt. Conf. Artif. Intell.*, pp. 1778–1784, 2013.

[192] K. Eggensperger *et al.*, "Towards an Empirical Foundation for Assessing Bayesian Optimization of Hyperparameters," *BayesOpt Work.*, pp. 1–5, 2013.

[193] R. Garcia Leiva, A. Fernandez Anta, V. Mancuso, and P. Casari, "A Novel Hyperparameter-Free Approach to Decision Tree Construction That Avoids Overfitting by Design," *IEEE Access*, vol. 7, pp. 99978–99987, 2019.

[194] B. Lakshminarayanan, D. M. Roy, and Y. W. Teh, "Top-down particle filtering for Bayesian decision trees," *30th Int. Conf. Mach. Learn. ICML 2013*, vol. 28, no. PART 2, pp. 1317–1325, 2013.

[195] K. Q. Weinberger, J. Blitzer, and L. K. Saul, “Distance metric learning for large margin nearest neighbor classification,” *Adv. Neural Inf. Process. Syst.*, pp. 1473–1480, 2005.

[196] T. Denœux, “A k-Nearest Neighbor Classification Rule Based on Dempster-Shafer Theory,” *IEEE Trans. Syst. Man. Cybern.*, vol. 25, no. 5, pp. 804–813, 1995.

[197] C. Leslie, E. Eskin, and W. S. Noble, “The spectrum kernel: a string kernel for SVM protein classification.,” *Pac. Symp. Biocomput.*, pp. 564–575, 2002.

[198] Soman, K. P., R. Loganathan, and V. Ajay. *Machine learning with SVM and other kernel methods*. PHI Learning Pvt. Ltd., 2009.

[199] J. Kamruzzaman, R. A. Sarker, and I. Ahmad, “SVM based models for predicting foreign currency exchange rates,” *Proc. - IEEE Int. Conf. Data Mining, ICDM*, pp. 557–560, 2003.

[200] C. P. Diehl and G. Cauwenberghs, “SVM Incremental Learning, Adaptation and Optimization,” *Proc. Int. Jt. Conf. Neural Networks*, vol. 4, no. x, pp. 2685–2690, 2003.

[201] C. C. Hsu, C. H. Wu, S. C. Chen, and K. L. Peng, “Dynamically optimizing parameters in support vector regression: An application of electricity load forecasting,” *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, vol. 2, no. 3, pp. 1–8, 2006.

[202] Y. Yao, Q. Hu, H. Yu, and J. W. Grzymala-Busse, “Rough Sets, fuzzy sets, data mining, and granular computing: 15th International Conference, RSFDGrC 2015 Tianjin, China, November 20–23, 2015 proceedings,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9437, pp. 464–474, 2015.

[203] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, “Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms,” *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, vol. Part F1288, pp. 847–855, 2013.