

**To Explore and Analyze the Role of IOT, Artificial
Intelligence and Machine Learning in Solving the
Commuting Problems of Smart Cities**

स्मार्ट शहरों में आवागमन करने की समस्याओं को हल करने में आईओटी,
आर्टिफिशियल इंटेलिजेंस और मशीन लर्निंग की भूमिका का अन्वेषण और
विश्लेषण करना

**A
Thesis**

**Submitted for the Award of the Ph.D. degree of
PACIFIC ACADEMY OF HIGHER EDUCATION
AND RESEARCH UNIVERSITY**

By

AVINASH DANGWANI

अविनाश डंगवानी

Under the supervision of

Dr. ASHOK KUMAR JETAWAT

Professor,
Pacific Academy of Higher Education,
& Research University, Udaipur.

Dr. CHANDAN SINGH RAWAT

HOD, Department of Electronics
& Telecommunication VESIT,
Chembur Mumbai.



**FACULTY OF COMPUTER ENGINEERING
PACIFIC ACADEMY OF HIGHER EDUCATION
AND RESEARCH UNIVERSITY, UDAIPUR**

2024

DECLARATION

I, **AVINASH DANGWANI S/O SHRI SUNDER DANGWANI** resident Thane District Maharashtra, hereby declare that the research work incorporated in the present thesis entitled **“To Explore and Analyze the Role of IOT, Artificial Intelligence and Machine Learning in Solving the Commuting Problems of Smart Cities”** (स्मार्ट शहरों में आवागमन करने की समस्याओं को हल करने में आईओटी, आर्टिफिशियल इंटेलिजेंस और मशीन लर्निंग की भूमिका का अन्वेषण और विश्लेषण करना) is my original work. This work (in part or in full) has not been submitted to any University for the award or a Degree or a Diploma. I have properly acknowledged the material collected from secondary sources wherever required.

I solely own the responsibility for the originality of the entire content.

Signature of the Candidate

Date:

FACULTY OF ENGINEERING
PACIFIC ACADEMY OF HIGHER EDUCATION AND
RESEARCH UNIVERSITY, UDAIPUR

Dr. ASHOK KUMAR JETAWAT
Professor

CERTIFICATE

It gives me immense pleasure in certifying that the thesis **“To Explore and Analyze the Role of IOT, Artificial Intelligence and Machine Learning in Solving the Commuting Problems of Smart Cities”** (स्मार्ट शहरों में आवागमन करने की समस्याओं को हल करने में आईओटी, आर्टिफिशियल इंटेलिजेंस और मशीन लर्निंग की भूमिका का अन्वेषण और विश्लेषण करना) and submitted by **AVINASH DANGWANI** is based on the research work carried out under my guidance. He / she has completed the following requirements as per Ph.D. regulations of the University;

- (i) Course work as per the University rules.
- (ii) Residential requirements of the University.
- (iii) Regularly presented Half Yearly Progress Report as prescribed by the University.
- (iv) Published / accepted minimum of two research paper in a refereed research journal.

I recommend the submission of thesis as prescribed/notified by the University.

Date:

Name and Designation of Supervisor

Dr. ASHOK KUMAR JETAWAT
Professor,
Pacific Academy of Higher

CERTIFICATE

It gives me an immense pleasure in certifying that the thesis **“To Explore and Analyze the Role of IOT, Artificial Intelligence and Machine Learning in Solving the Commuting Problems of Smart Cities”** (स्मार्ट शहरों में आवागमन करने की समस्याओं को हल करने में आईओटी, आर्टिफिशियल इंटेलिजेंस और मशीन लर्निंग की भूमिका का अन्वेषण और विश्लेषण करना) and submitted by **AVINASH DANGWANI** is based on the research work carried out under my guidance. He / she has completed the following requirements as per Ph.D. regulations of the University;

- (i) Course work as per the University rules.
- (ii) Residential requirements of the University.
- (iii) Regularly presented Half Yearly Progress Report as prescribed by the University.
- (iv) Published / accepted minimum of two research paper in a refereed research journal.

I recommend the submission of thesis as prescribed/notified by the University.

Date:

Name and Designation of Co-Supervisor

Dr. CHANDANSINGH RAWAT

HOD Department of Electronics
& Telecommunication, VESIT,
Chembur Mumbai.

COPYRIGHT

I, **AVINASH DANGWANI**, hereby declare that the Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan, shall have the rights to preserve, use and disseminate this dissertation entitled **“To Explore and Analyze the Role of IOT, Artificial Intelligence and Machine Learning in Solving the Commuting Problems of Smart Cities”** (स्मार्ट शहरों में आवागमन करने की समस्याओं को हल करने में आईओटी, आर्टिफिशियल इंटेलिजेंस और मशीन लर्निंग की भूमिका का अन्वेषण और विश्लेषण करना) in print or in electronic format for the academic research / purpose.

Date:

Signature of Candidate

Place:

ACKNOWLEDGEMENT

I do not find adequate words to express my deepest regard and gratitude to my eminent and esteemed supervisor **Dr. Ashok Kumar Jetawat**, Professor, Department of Computer Engineering, Pacific Academy of higher education and research centre Udaipur Rajasthan, for giving me inspiration, guidance, valuable suggestions, opinions and corrections for betterment of my research work. I will always be grateful to him. He was always available for help at any point of time. His knowledge and experience about smart cities had inspired me to take up this research project. I am fortunate to have an advisor and mentor who surpassed all expectations.

I am profoundly grateful to my co advisor **Dr. Chandan Singh Rawat**, Head of Department Electronics and Telecommunication Vivekanand Education society institute of Technology Chembur Mumbai for his unwavering support, which has played a pivotal role in shaping both my research endeavour's and my overall experience as a doctoral student. Working under his mentorship has been a privilege and a source of inspiration.

As I reflect on this milestone, I am reminded of the profound significance of the support network that surrounds us. To **Prof. Jayshree Jain madam**, I extend my deepest gratitude for her unwavering support and understanding throughout this journey. In this journey, I have witnessed the impact of her support, whether it was through guiding me at different stages of my research work or offering words of encouragement when the path seemed daunting. Her contribution, though perhaps less visible, has been just as vital in shaping my academic growth.

I express my special and sincere thanks to our honourable **Dr. Hemant Kothari**, Dean Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan, and **Shri Ramesh Agarwal** who guided me at different stages of my research work. Without their precious support it would not be possible to conduct this research.

I would like to thank shri **P.S. TALESARA**, Director Pyrotech companies, for allowing me to understand their company operations and giving me invaluable suggesting on AI.

I am thankful to General manager **BEST Mumbai Mr. Shetty** and **Mr. Sunil Jadhav** Deputy Head Planning Department Wadala Mumbai for providing me required help and support.

I am grateful to express my thanks to **Dr. Kushal Tuckley (Ex. Prof IIT Mumbai)**, currently holding the post of Chairman and Director R&D in “**AGV Systems**” Pvt.Ltd Ambernath for giving me, opinion and invaluable suggestions to improve my research work in the smart cities.

I would like to thank **Ms. Meenakshi Tyagi**, Librarian Vivekanand Education Socety’s Institute of Technology Chembur, Mumbai and **Ms. Kusum** and **Dr. Surya** Pacific University Udaipur for helping me in creating Plagiarism reports.

I pay all my heartfelt gratitude to my friends **Dr. Prasad Kulkarni** and **Mr. Lal Gwalani** Sound Solutions Sion Mumbai for their unwavering support and all those, who have helped me directly or indirectly making this research work a success.

Special note of thanks to my father **Shri Sunder Dangwani**, my late mother **Smt. Leena Dangwani** whose blessings, love and support have always helped in my research work. My lovely wife **Smt. Mahima Dangwani**, my Son’s **Mr. Mohit Dangwani** and **Mr. Arvin Dangwani** who stood by me always. I thank them for their encouragement, love and support.

I acknowledge my gratitude with sense of reverence to the “**Almighty God**” and those who have contributed directly or indirectly and spared time for the completion of my research work and making it successful.

Last but not the least, my distinctive thanks to **M/s Shorya Thesis Printing & Binding, Udaipur**, in shaping the matter, creative design work and bringing out this document meticulously, neatly and timely.

Thank you all for everything.

OM SAI RAM!
Avinash Dangwani

PREFACE

The increase in the smart city initiatives in the past decades has made the world community appreciate the depth of the complicated problems of the urban settings and they appear to be in desperate need of the new ideas how to deal with them. It is undeniable that the world is getting more and more connected and the pace of urbanization is quickly increasing. This situation makes it much more promising that a variety of smart technologies and systems can be put into practice to effectively increase the efficiency and sustainability of urban systems and significantly enhance the quality of life for city inhabitants.

The given thesis represents the outcome of the thorough investigation and further studies which shows how machine learning algorithms may be used to solve major commuting issues in smart cities. By combining the cross-system perspective of computer science, urban planning, and data analytics, the study explores the complex dance between technological creativity and municipal management.

Such a trip is not about reaching the sizeable figure; rather, it is about the travel itself. It navigates the universe of machine learning methods, starting from the classical algorithms to the cutting-edge ones submitted to the differentiable nature of the challenges of smart cities, to determine which one offers the most suitable and effective solution to all these predicaments. Careful experimentation and comparative analysis of different machine learning approaches are the major pillars of this research. It refrains from settling on the side of the various algorithms and consensus is reached on their effectiveness in real world based on the findings.

I firmly hope that the claim presented herein contributes significantly to the ongoing discussion of the design, implementation, and another international level matter of smart cities governance. The proposed research is divided into six chapters, which gives complete roadmap of the research scope and implementation.

Chapter – 1 Introduction:

It emphasizes on the basic terminology of research topic, it creates awareness about requirement of smart transportation in smart city, merits and demerits of this terminology. This chapter highlights the critical transportation issues such as traffic

congestion, seat availability, bus tracking system, lack of bus interval information, and the need for enhanced passenger communication tools such as public addressing system. It also discuss role of Internet of Things, Artificial Intelligence and Machine learning Algorithms in effectively addressing commuting issues in smart cities.

Chapter – 2 Literature Review:

It focuses on the groundwork done on smart transportation till date. Its manifest will be to denote gaps in research already done. It highlights the innovative work done or proposed by various Authors in the relevant fields.

Chapter – 3 Research Methodology:

This chapter signify the role of different research methodologies adopted to study a research problem, along with the underlying logic behind them. This chapter also include, Hypothesis to be tested, Scope of study, various IOT data collection methods, research design and different tools required for analysis of machine learning algorithms.

Chapter – 4 Feature Extraction and Data Processing using IoT:

This chapter focuses on features selection and data processing. Feature extraction aims to reduce the number of features in a dataset by selecting required features from the existing ones (and then discarding the redundant features). These new reduced set of features were being summarize most of the information contained in the original set of features. Out of twenty one features, only seven features were filtered using eight feature selection Machine learning algorithms.

Chapter – 5 Utilization of AI and ML Prediction Algorithms:

This chapter discuss how we can make use of Artificial Intelligence and Machine Learning algorithms for predicting Traffic congestion. Machine learns from experiences; Algorithms develop multiple models and each model is analogous to an experience. The main objective of Machine Learning algorithm is to improve prediction accuracy for Traffic congestion. This doctoral thesis embarks on a detailed exploration of nine prominent machine learning algorithms, evaluating their performance across a spectrum of critical performance metrics. The primary objective of this research is to conduct a rigorous comparative analysis of machine learning algorithms, shedding light on their strengths, weaknesses, and applicability in diverse scenarios. The evaluation

framework encompasses an array of performance parameters, including accuracy, incorrectly classified instances, kappa statistics, and various aspects of the confusion matrix such as true positives (TP), true negatives (TN), false positives (FP), false negatives (FN), precision, recall, and F-measure. The nine machine learning algorithms listed below are selected for this study represent a comprehensive spectrum of techniques commonly employed in real-world applications:

1. Bayes Net
2. Naïve Bayes
3. Logistic
4. SMO
5. IBk
6. KStar
7. MultiClass Classifier
8. Random Forest
9. RandomTree

This research contributes to the field of machine learning by offering a systematic comparative and in-depth analysis of nine prominent algorithms based on a set of performance parameters. Each algorithm is thoroughly analyzed and compared based on their performance using Udaipur traffic data set obtained from TOMTOM server. To assess the performance and ability of Machine learning models, K fold cross validation techniques and percentage split methods are used. K-Fold cross-validation involves splitting the dataset into K subsets (or folds) of approximately equal size. The model is trained K times, each time using K-1 folds as training data and the remaining fold as validation data. 10 fold, 25 fold and 30% split cases are analyzed for thesis.

Finally this doctoral thesis inquiries intellectual landscape through a focused hypothesis-driven study aimed at investigating usefulness of different technologies and different machine learning algorithms in solving commuting problems in smart cities.

Chapter – 6 Conclusion and Future Scope:

Last chapter put emphasis on the culmination of research on smart commuting and future improvisation that can be done, as learning is continuous process for any field of research and development. It also leave some open questions to be investigated in future for researchers.

LIST OF CONTENTS

Chapter – 1 INTRODUCTION

1.1	Smart City	1
1.2	Transportation Problems in Smart Cities	5
1.2.1	Traffic Congestion	5
1.2.2	Bus Location	7
1.2.3	Seat Availability	8
1.2.4	Bus Interval Time	10
1.2.5	Public Addressing System	11
1.3	Internet of Things	13
1.4	Role of Machine Learning	16
1.4.1	Supervised	18
1.4.2	Unsupervised	18
1.4.3	Semi Supervised	18
1.5	Machine Learning Algorithms	19
1.5.1	Bayes Net	19
1.5.2	Naive Bayes	19
1.5.3	Logistic	20
1.5.4	SMO	20
1.5.5	IBK	20
1.5.6	K Star	21
1.5.7	Multi Class Classifier	21
1.5.8	Random Forest	22
1.5.9	Random Tree	23
1.6	Applications of Machine Learning	23
1.6.1	Smart Traffic Management and Transportation	23
1.6.2	Combating Pollution	24
1.6.3	Public Safety	24
1.6.4	Smart Grids and Machine Learning	24
1.7	Role of Artificial Intelligence	25

Chapter – 2 LITERATURE REVIEW

2.1	Introduction	32
2.2	Enhancing Public Transport Experience	32
2.3	Commuting and Traffic Congestion Issues	41
2.4	Technologies and Transportation Systems	54
2.5	IoT Based Traffic Prediction Models	61
2.6	Artificial Intelligence and Traffic Management	71

Chapter – 3 RESEARCH METHODOLOGY

3.1	Introduction	74
3.2	Significance of Research	74
3.3	Problem Statement	75
3.4	Objectives	75
3.5	Hypothesis	76
3.6	Scope of Study	77
3.7	Research Design	77
3.7.1	Information Collection Procedure	78
3.7.2	Comparative Analysis of Technologies and Models	80
3.7.3	Machine Learning Predictive Model Development	81
3.7.4	Performance Evaluation	81
3.7.5	Machine Learning Approach Selection	81
3.8	Data Collection Sources	81
3.8.1	Chicago Dataset: “Chicago_Traffic_1000	82
3.8.2	Udaipur Dataset: “Udaipur Traffic”	84
3.9	Tools and Techniques	87
3.9.1	Weka Tool	87
3.9.1.1	Components and Techniques	89
3.9.2	Python	93
3.10	Summary	94

Chapter – 4 FEATURE EXTRACTION

4.1 Overview	96
4.2 Feature Selection Methods	99
4.2.1 Info Gain Attribute Eval	101
4.2.2 Correlation Attribute Eval	103
4.2.3 Classifier Attribute Eval	105
4.2.4 Cfs Subset Eval	106
4.2.5 Gain Ratio Attribute Eval	107
4.2.6 OneR Attribute Eval	109
4.2.7 ReliefF Attribute Eval	111
4.2.8 Symmetrical Uncert Attribute Eval	113
4.3 Feature Extraction Using Multiple Regression	115
4.4 Combined Feature Selection Matrix	117
4.5 Rank and Percentile	121
4.6 Summary	122

Chapter – 5 UTILIZATION OF AI AND ML PREDICTION ALGORITHMS

5.1 Traffic Control Systems for Smart Cities	123
5.1.1 IoT-Based Traffic Prediction Models	124
5.1.2 Machine Learning-Based Traffic Prediction	125
5.2 Machine Learning Predictive Model for Smart Transportation	126
5.3 Analysis of Machine Learning Modules for Smart Transportation	128
5.3.1 Performance Measure	128
5.3.2. Error Measures	132
5.3.3 Cross-Validation Configuration Setting (10-folds) Results	134
A. Performance Measures	135
B. Error Measure Results	146
C. Execution Time Results	149
5.3.4 Cross-Validation Configuration Setting (25-folds) Results	150
A. Performance Measures	151
B. Error Measure Results	162
C. Execution Time Results	164

5.3.5	Cross-Validation Configuration Setting (30% Split) Results	166
A.	Performance Measures	167
B.	Error Measure Results	177
C.	Execution Time Results	179
5.3.6	Consolidated Result	181
5.3.7	Dominance Chart	191
5.3.8	Weighted Sum Model Analysis Using Python	193
5.3.9	Rank and Percentile Method	197
5.4	Hypothesis Testing Results	204
5.5	Summary	209
Chapter – 6 CONCLUSION AND FUTURE SCOPE		
6.1	Summary of Findings	210
6.1.1	Configuration Setting: Cross Validation – 10 folds	212
6.1.2	Configuration Setting: Cross Validation – 25 folds	212
6.1.3	Configuration Setting: Cross Validation – 30% Split	213
6.2	Hypotheses Based Findings	214
6.2.1	Hypotheses 1	214
6.2.2	Hypotheses 2	215
6.3	Challenges in Smart Transportation	217
6.4	Future Directions in Smart Transportation	218
	REFERENCES	220
	DISSEMINATION OF RESEARCH WORK	232
	ANNEXURE	
	Hypothesis – 1 Survey Questions	234
	Paper – 1: Exploring the Role of Machine Learning Algorithms for Smart Commuting in Smart Cities	
	Paper – 2: Use of AI in Cloud based Certificate Authentication for Travel Concession.	
	Paper – 3: Data Analytics Sales Prediction Model	
	CERTIFICATES	

LIST OF FIGURES

1.1	Smart City Integration	2
1.2	Smart City Components & IoT	4
1.3	Architecture of IoT	13
1.4	Steps of Applying Machine Learning Techniques	17
1.5	Old Transportation System	25
1.6	Sub-systems in Intelligent Transportation System	26
3.1	Chicago Traffic Tracker	83
3.2	Chicago Data Portal	84
3.3	tomtom Route Monitoring	85
3.4	tomtom Traffic Stats	86
3.5	tomtom Junction Analytics	86
3.6	Preprocessing Step Tool	89
3.7	Attribute Selection	90
3.8	All Attribute (Applied on Dataset Chicago_Traffic_1000)	90
3.9	Classification Algorithms Tool	91
3.10	Clustering Algorithm	92
3.11	Visualization	92
4.1	Attribute Score for Info Gain Attribute Eval	102
4.2	Attribute Score for Correlation Attribute Eval	104
4.3	Attribute Score for Gain Ratio Attribute Eval	108
4.4	Attribute Score for OneR Attribute Eval	110
4.5	Attribute Score for ReliefF Attribute Eval	112
4.6	Attribute Score for Symmetrical Uncert Attribute Eval	114
4.7	Attribute and Frequency	119
5.1	Vehicle Location Tracking Using IoT and Machine Learning	124
5.2	Smart Transportation System	127
5.3	Confusion Matrix	130
5.4	ROC Curve	131
5.5	Area Under ROC Curve	132
5.6	4-fold Cross-Validation Example	134
5.7	Performance Measure Accuracy (Cross-Validation: 10 Folds)	135

5.8	Incorrectly Classified Instances (Cross-Validation: 10 Folds)	136
5.9	Kappa Statistic Values (Cross-Validation: 10 Folds)	136
5.10	TP Rate (Cross Validation: 10-Folds – Low Traffic)	138
5.11	FP Rate (Cross Validation: 10-Folds – Low Traffic)	138
5.12	Precision (Cross Validation: 10-Folds – Low Traffic)	139
5.13	Recall (Cross Validation: 10-Folds – Low Traffic)	140
5.14	F-Measure (Cross Validation: 10-Folds – Low Traffic)	140
5.15	ROC Area (Cross Validation: 10-Folds – Low Traffic)	141
5.16	TP Rate (Cross Validation: 10-Folds – Heavy Traffic)	142
5.17	FP Rate (Cross Validation: 10-Folds – Heavy Traffic)	143
5.18	Precision (Cross Validation: 10-Folds – Heavy Traffic)	144
5.19	Recall (Cross Validation: 10-Folds – Heavy Traffic)	144
5.20	F-Measure (Cross Validation: 10-Folds – Heavy Traffic)	145
5.21	ROC Area (Cross Validation: 10-Folds – Heavy Traffic)	145
5.22	Mean Absolute Error (Cross-Validation: 10 Folds)	146
5.23	Root Mean Squared Error (Cross-Validation: 10 Folds)	147
5.24	Relative Absolute Error (Cross Validation: 10-Folds)	147
5.25	Root Relative Squared Error (Cross Validation: 10-Folds)	148
5.26	Average Execution Time (Cross Validation: 10-Folds)	149
5.27	Performance Measure Accuracy (Cross-Validation: 25 Folds)	151
5.28	Incorrectly Classified Instances (Cross-Validation: 25 Folds)	152
5.29	Kappa Statistic (Cross-Validation: 25 Folds)	152
5.30	TP Rate (Cross Validation: 25-Folds – Low Traffic)	154
5.31	FP Rate (Cross Validation: 25-Folds – Low Traffic)	154
5.32	Precision (Cross Validation: 25-Folds – Low Traffic)	155
5.33	Recall (Cross Validation: 25-Folds – Low Traffic)	155
5.34	F-Measure (Cross Validation: 25-Folds – Low Traffic)	156
5.35	ROC Area (Cross Validation: 25-Folds – Low Traffic)	157
5.36	TP Rate (Cross Validation: 25-Folds – Heavy Traffic)	158
5.37	FP Rate (Cross Validation: 25-Folds – Heavy Traffic)	159
5.38	Precision (Cross Validation: 25-Folds – Heavy Traffic)	159
5.39	Recall (Cross Validation: 25-Folds – Heavy Traffic)	160

5.40	F-Measure (Cross Validation: 25-Folds – Heavy Traffic)	160
5.41	ROC Area (Cross Validation: 25-Folds – Heavy Traffic)	161
5.42	Mean Absolute Error (Cross-Validation: 25 Folds)	162
5.43	Root Mean Squared Error (Cross-Validation: 25 Folds)	163
5.44	Relative Absolute Error (Cross Validation: 25-Folds)	163
5.45	Root Relative Squared Error (Cross Validation: 25-Folds)	164
5.46	Average Execution Time (Cross Validation: 25-Folds)	165
5.47	Data Set Split	166
5.48	Performance Measure Accuracy (Cross-Validation: 30% Split)	167
5.49	Incorrectly Classified Instances (Cross-Validation: 30% Split)	168
5.50	Kappa Statistic (Cross-Validation: 30% Split)	168
5.51	TP Rate (Cross-Validation: 30% Split)	169
5.52	FP Rate (Cross-Validation: 30% Split)	170
5.53	Precision (Cross-Validation: 30% Split)	170
5.54	Recall (Cross-Validation: 30% Split)	171
5.55	F-Measure (Cross-Validation: 30% Split)	171
5.56	ROC Area (Cross-Validation: 30% Split)	172
5.57	TP Rate (Cross-Validation: 30% Split)	173
5.58	FP Rate (Cross-Validation: 30% Split)	174
5.59	Precision (Cross-Validation: 30% Split)	174
5.60	Recall (Cross-Validation: 30% Split)	175
5.61	F-Measure (Cross-Validation: 30% Split)	175
5.62	ROC Area (Cross-Validation: 30% Split)	176
5.63	Mean Absolute Error (Cross-Validation: 30% Split)	177
5.64	Root Mean Squared Error (Cross-Validation: 30% Split)	178
5.65	Relative Absolute Error (Cross-Validation: 30% Split)	178
5.66	Root Relative Square Error (Cross-Validation: 30% Split)	179
5.67	Average Execution Time (Cross-Validation: 30% Split)	180
5.68	Accuracy (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	181
5.69	Incorrectly Classified Instances (10-Fold, 25-Fold and 30% Split)	181
5.70	Kappa Statistics (Cross-Validation: 10-Fold, 25-Fold and 30%	182
5.71	TP Rate (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	182

5.72	FP Rate (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	183
5.73	Precision (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	183
5.74	Recall (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	184
5.75	F Measure (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	184
5.76	ROC Area (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	185
5.77	TP Rate (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	185
5.78	FP Rate (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	186
5.79	Precision (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	186
5.80	Recall (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	187
5.81	F Measure (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	187
5.82	ROC Area (Cross-Validation: 10-Fold, 25-Fold and 30% Split)	188
5.83	Mean Absolute Error (10-Fold, 25-Fold and 30% Split)	188
5.84	Root Mean Squared Error (10-Fold, 25-Fold and 30% Split)	189
5.85	Relative Absolute Error(10-Fold,25-Fold & 30% Split)	189
5.86	Root Relative Squared Error (10-Fold, 25-Fold and 30% Split)	190
5.87	Execution Time (10-Fold, 25-Fold and 30% Split)	191
5.88	Running MCDM Method in Python Environment	196
5.89	Right Tailed Chi-Square curve	206
5.90	T-Test Python Program Output	208
5.91	Left Tailed T-test curve	208
6.1	Cross Validation 10 – Fold Findings	212
6.2	Cross Validation 25 – Fold Findings	213
6.3	Cross Validation 30% – Split Findings	213

LIST OF TABLES

4.1	Attribute Names and ID	100
4.2	Attribute Score for Info Gain Attribute Eval	102
4.3	Attribute Score for Correlation Attribute Eval	104
4.4	Attribute Score for Classifier Attribute Eval	106
4.5	Attribute Score for Gain Ratio Attribute Eval	108
4.6	Attribute Score for OneR Attribute Eval	110
4.7	Attribute Score for ReliefF Attribute Eval	112
4.8	Attribute Score for Symmetrical Uncert Attribute Eval	114
4.9	Variables Entered / Removed	115
4.10	Model Summary	115
4.11	ANOVA Summary	116
4.12	Coefficients Summary	116
4.13	Attribute Names and ID	117
4.14	Combined Feature Selection Matrix	118
4.15	Attribute and Count	119
4.16	Overall Attribute Performance in Percentage (%)	120
4.17	Rank & Percentile Approach Results	121
5.1	Classifiers and Accuracy Measures (Cross Validation: 10-Folds)	135
5.2	Classifiers and Performance Measures Class Label: Low Traffic	137
5.3	Classifiers Performance Measure Class Label: Heavy Traffic	142
5.4	Classifiers and Error Measures (Cross Validation: 10-Folds)	146
5.5	Classifiers and Average Execution Time (10-Folds)	149
5.6	Classifiers and Accuracy Measures (Cross-Validation: 25-Folds)	151
5.7	Classifiers and Performance Measures Class Label: Low Traffic	153
5.8	Classifiers Performance Measure Class Label: Heavy Traffic:	158
5.9	Classifiers and Error Measures (Cross Validation: 25-Folds)	162
5.10	Classifiers and Average Execution Time (25-Folds)	165
5.11	Classifiers and Accuracy Measures (30% Split)	167
5.12	Classifiers Performance Measures Class Label: Low Traffic:	169
5.13	Classifiers Performance Measure Class Label: Heavy Traffic:	173
5.14	Classifiers and Error Measures (Cross Validation: 30% Split)	177

5.15	Classifiers and Average Execution Time (30% Split)	180
5.16	Machine Learning Algorithms Dominance Chart	192
5.17	Weighted Sum Score (Cross Validation-10 Folds)	196
5.18	Classification Models and Ranks (Cross Validation-10 Folds)	200
5.19	Weighted Sum Score (Cross Validation-25 Folds)	201
5.20	Classification Models and Ranks (Cross Validation-25 Folds)	201
5.21	Weighted Sum Score (Cross Validation-30% Split)	202
5.22	Classification Models and Ranks (Cross Validation-30% Split)	203
5.23	Type of Technology and Level of Enhancement	204
5.24	Chi-Square Test Results	205
5.25	Calculation of Expected Frequency	205
5.26	Observed and Expected Frequency calculations.	205
5.27	T-test calculation	207

ABBREVIATIONS

AALS	Accident Alert Sound System
AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Program Interfaces
AQI	Air Quality Index
ARIMA	Autoregressive Integrated Moving Average
ATM	Adaptive Traffic Management
AUC	Area Under ROC Curve
AV	Autonomous Vehicles
CNN	Convolution Neural Networks
CoAP	Constrained Application Protocol
CRN	Cognitive Radio Networks
DL	Deep Learning
ETM	Electronic Ticketing Machines
FN	False Negative
FP	False Positive
FPGA	Field Programmable Gate Array
FPR	False Positive Rate
GRU	Gated Recurrent Unit
HTTP	Hypertext transfer protocol
IBK	Instance-Based k-Nearest Neighbors
IBSO	Improved Binary Swarm Optimization
ICT	Information Communication and Technology
IoT	Internet of Things
IEEE	Institute of Electrical and Electronics Engineers
ITS	Intelligent Transportation System
KNN	k-Nearest Neighbours
LPWAN	Low-power wide area network
LSTM	Long Short-Term Memory
MaaS	Mobility-as-a-Service
ML	Machine Learning

MLP	Multilayer Perceptron
MQTT	Message Queuing Telemetry Transport
OSM	Open Street Map
PAS	Public Addressing System
RF	Random Forest
RNN	Recurrent Neural Networks
ROC	Receiver Operating Characteristics
SARIMA	Seasonal Auto Regressive Integrated Moving Average
SCs	Smart Cities
SDN	Software-Defined Network
SEE-TREND	Secure Early Traffic-Related Event Detection
SFC	Static Feedback Control
SMO	Sequential Minimal Optimization
SOAP	Simple Object Access Protocol
SSL	Secure Socket Layer
SVM	Support Vector Machines
TMS	Transport Management System
TN	True Negative
TNSTC	Tamil Nadu State Transport Corporation
TP	True Positive
TPR	True Positive Rate
UTAUT	Unified Theory of Acceptance and Use of Technology
VKT	Vehicle Kilometers Traveled
VMT	Vehicle miles travelled
WSDL	Web Services Description Language

SYMBOLS

μ	Mean
σ	Standard Deviation
χ	Chi – square
π	Pi
α	Significant Level
f	Frequency
Ef	Expected Frequency
β	Standardized Coefficient

Chapter – 1

Introduction

- 1.1 Smart City
- 1.2 Transportation Problems in Smart Cities
 - 1.2.1 Traffic Congestion
 - 1.2.2 Bus Location
 - 1.2.3 Seat Availability
 - 1.2.4 Bus Interval Time
 - 1.2.5 Public Addressing System
- 1.3 Internet of Things
- 1.4 Role of Machine Learning
 - 1.4.1 Supervised
 - 1.4.2 Unsupervised
 - 1.4.3 Semi supervised
- 1.5 Machine Learning Algorithms
 - 1.5.1. Bayes Net
 - 1.5.2. Naive Bayes
 - 1.5.3. Logistic
 - 1.5.4. SMO
 - 1.5.5. IBK
 - 1.5.6. K Star
 - 1.5.7. Multi Class Classifier
 - 1.5.8. Random Forest
 - 1.5.9. Random Tree
- 1.6 Applications of Machine Learning
 - 1.6.1 Smart traffic management and transportation
 - 1.6.2. Combating Pollution
 - 1.6.3. Public Safety
 - 1.6.4. Smart grids and machine learning
- 1.7 Role of Artificial Intelligence

The Internet of Things is a technology-based system that uses numerous devices and gadgets in place of human contact. This enables the emergence of smart cities everywhere in the world. The development of smart city systems for sustainable living, improved comfort, and enhanced productivity for people has been accelerated by the internet of things, which hosts several technologies and permits interactions between them. The Internet of Things for Smart Cities has a significant impact on a wide range of businesses, and it relies on a variety of underlying technologies to function. In this research, the Internet of Things in Smart Cities is fully studied. The key components of the IoT¹-based Smart City environment are given first, followed by the architectures, networking, and Artificial Algorithms that enable these domains in IoT-based Smart City systems.

1.1 Smart City

The world is changing quickly, cities are expanding, and the urban population is growing. Housing affordability pushes the public to live far from their places of employment, which increases the demand for transportation and exacerbates commuting issues. The world's future lies in smart cities, which will exhibit human-like agility and sharp reasoning. Building smart cities will contribute more to fostering economic development. Any city's ability to grow successfully and sustainably depends on mobility and transportation. We must increase mobility, but we must do so more intelligently.

A cleverly organized network of interconnected gadgets and objects—also alluded to as a "Advanced Digital City"—that exchange information by means of wireless innovations and the cloud make up a noteworthy portion of this ICT² system. Real-time information collection, examination, and administration capabilities given by cloud-based IoT apps help businesses, governments, and people in taking more brilliant choices that upgrade their quality of life. Smartphones, portable gadgets, associated automobiles and homes are a couple of the ways that citizens associate with the ecosystems of intelligent cities. Costs can be decreased and sustainability can be expanded by integrating gadgets and information with a city's physical foundation and administration. With the utilization of the IoT, communities can distribute energy more

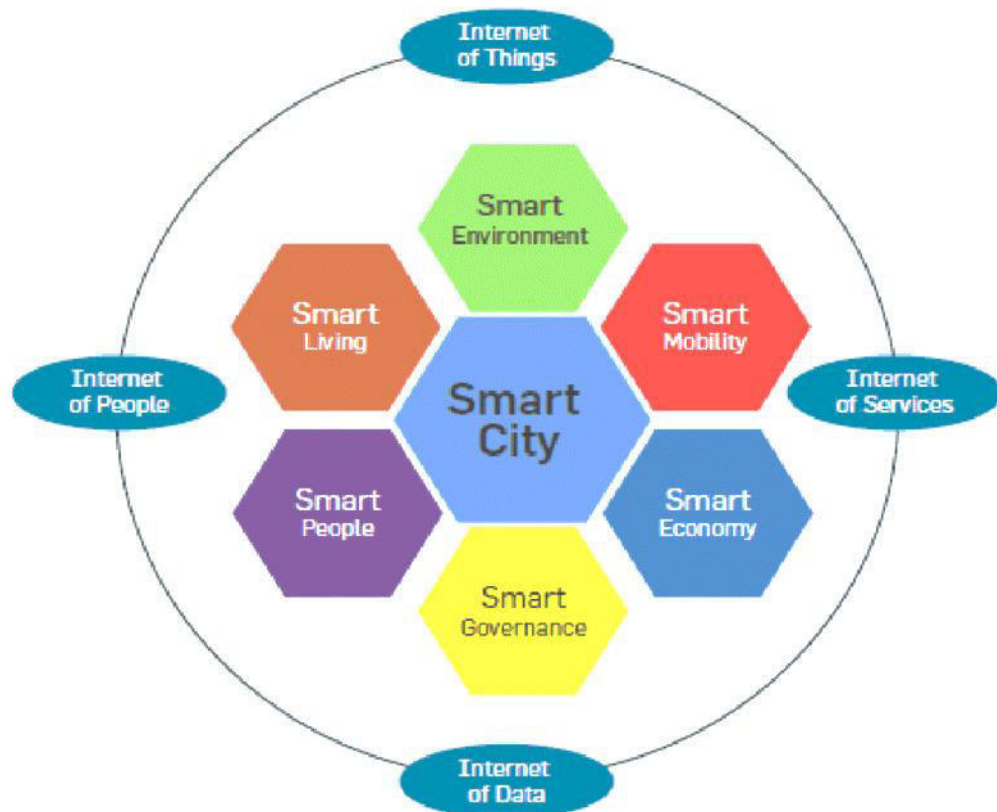
¹ Internet of Things

² Information Communication and Technology

effectively, collect waste more effectively, relieve traffic blockage, and upgrade Air quality. AQI³ is a measure of how polluted the air is, or is likely to be. Problems associated with high and low AQI levels include: Health effects: Respiratory diseases: High AQI levels can cause respiratory diseases such as asthma, bronchitis, and more. may worsen conditions such as respiratory illness. Cardiovascular effects , Environmental Impact, Climate Change and Economic cost:

The lack of suitable public transportation is the fundamental problem with urban mobility. Users feel uncomfortable while the system responds to a temporary rise in demand due to increase of crowd. By utilizing smart commuting, we can increase accessibility for those who use public transportation.

Figure 1.1: Smart City Integration



The figure above shows the integrated environment required to develop smart city, Six important components are listed in the figure 1.1. The integration includes the following aspects:

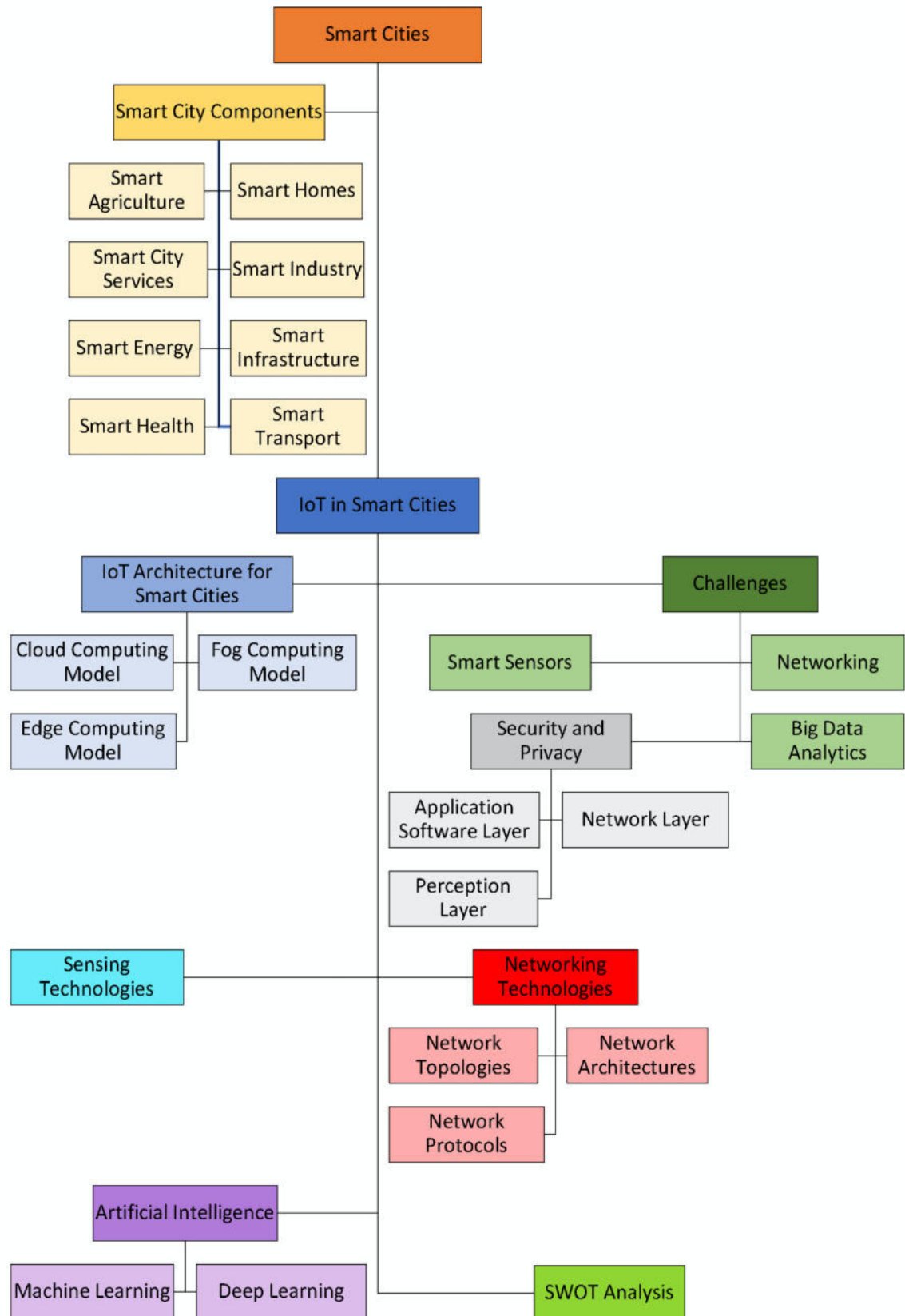
³ Air Quality Index

- Smart Environment
- Smart Mobility
- Smart Economy
- Smart Governance
- Smart People
- Smart Living
- Internet of Things
- Internet of Services
- Internet of Data
- Internet of People

Smart mobility is an important part of the broader concept of smart cities and represents a paradigm shift in urban planning and transportation. As cities grapple with challenges such as population growth, environmental sustainability, and the need for efficient transportation, the integration of smart mobility solutions is becoming increasingly important. The Bus information system is the important extension of smart mobility, which can feature a bus location information subsystem that gathers bus data and transfers it to a traffic information centre where users can get real-time bus-related information on their mobile phones. Information such as bus operating routes, bus locations, seat availability, arrival times, and bus interval periods boosts public transportation in smart cities and increases consumer happiness.

There is abundant literature available and research is done in smart cities which cover a wide topics related to smart city components and provide insight into the technical, social, and governance aspects that contribute to smart city development and sustainability. Comprehensive understanding of the various factors are required for shaping the future of urban environments. The smart city concept incorporates different components that contribute to the, by and large, goal of making a more productive, economical, and habitable city environment. The following diagram shows the various components of smart cities.

Figure 1.2: Smart City Components & IoT



Source: Syed and Kumar (2021)

Research into the integration of smart city components and the IoT has unlocked a wide range of possibilities and redefined the way we conceptualize, plan, and live in cities. Looking back at the interconnected web of technology, infrastructure, and data that underpins smart cities, it is clear that our urban environments are on the brink of transformation. The combination of smart components and his IoT has the potential to address pressing urban challenges and promote sustainability, efficiency, and improved quality of life for citizens.

1.2 Transportation Problems in Smart Cities

The main challenges associated with smart city transportation are traffic congestion, bus location, seat availability, arrival times, and bus interval time.

1.2.1 Traffic Congestion

Traffic congestion is an important aspect of building smart and sustainable cities. If the problem of traffic congestion is not properly addressed, it can affect various aspects of urban life, economy, environment, and public welfare, resulting in several negative effects. The possible consequences of not addressing traffic congestion are:

Increased Travel Time: Congestion slows traffic flow and increases travel time. Commuters experience delays, lost productivity, and reduced overall transportation efficiency.

Economic Impact: Economic costs associated with traffic congestion, reduce productivity and increased fuel consumption and increases the ecological footprint of transportation, contributing to climate change and environmental degradation.

Environmental Impact: Air pollution and greenhouse gas emissions are the main causes of traffic congestion in cities leading to serious health effects due to long-term exposure to traffic-related pollutants. Stop-and-go traffic contributes to increased emissions and poor air quality. Businesses experience increased transportation costs and the economy as a whole becomes less efficient.

Public Transportation Efficiency: Congestion affects the reliability and efficiency of public transportation. Bus delays may occur, making public transport less attractive and less competitive with private cars.

Impact on Emergency Services: Congestion can affect emergency services response times. Delays in reaching an emergency can have serious consequences and impact public safety.

Stress and Health Problems: Commuters suffer from stress due to delays caused by traffic jams. Increased stress causes psychological problems and reduces the overall quality of life of city dwellers.

Accessibility Limitations: Traffic congestion impedes accessibility and makes it difficult for people to reach their destinations on time. Accessibility limitations impact businesses, schools, and healthcare facilities, reducing the quality of life for the entire city.

Urban Planning Challenge: Traffic congestion makes urban planning and infrastructure development difficult. Cities face the challenge of optimizing land use, creating sustainable transportation solutions, and managing population growth.

Increase in Traffic Accidents: Traffic congestion increases the risk of traffic accidents. Increased accident rates result in injuries and fatalities, and further strain on emergency services.

Quality of Life: Traffic congestion affects the general well-being and daily life of citizens. Some of the ways in which traffic congestion can affect quality of life are time wastage, health impacts, Economic cost, Environmental effect, social life etc.

Inefficiency in Moving Goods: Traffic congestion hinders the movement of goods and delivery services. Logistics and supply chains are facing delays, impacting businesses and consumer services.

Understanding these impacts highlights the importance of implementing smart and sustainable transport solutions to address urban traffic congestion. Strategies may include investing in public transport, implementing smart traffic management systems, promoting different modes of transport, and promoting urban planning that prioritizes accessibility and sustainability. Reducing traffic congestion is essential to creating liveable, sustainable and economically dynamic urban spaces. Deploying intelligent transportation solutions, investing in public transport, promoting different modes of transportation, and adopting efficient city planning strategies can help reduce the negative effects of traffic congestion.

1.2.2 Bus Location

The lack of accurate bus location information in smart cities can lead to a number of issues that impact both the efficiency of public transportation and the overall commuting experience. Here are some issues related to the lack of bus location services in smart cities:

Unsafe Commuting Experience: Without real-time bus location services, commuters are uncertain about their current location and bus arrival time. This uncertainty can lead to longer wait times at bus stops, making it difficult for commuters to plan their trips effectively.

Inefficient Route Planning: Lack of real-time location data prevents the ability to dynamically adjust bus routes based on current traffic conditions. Buses follow a static schedule, which can cause inefficiencies when traffic jams, road closures, or unforeseen events occur.

Increased Congestion: Without real-time tracking, buses can be delayed, leading to increased congestion and congestion. When routes become congested, travel times become slower, less reliable, and public transportation options become less attractive.

Low Operational Efficiency: Without accurate location data, transit agencies cannot effectively monitor and manage their bus fleets. This can lead to operational inefficiencies, difficulty resolving maintenance issues in a timely manner, and difficulty optimizing service levels.

User Accessibility Limitations: Commuters do not have access to real-time information about bus locations, making it difficult to plan and coordinate their trips. This restriction may have a disproportionate impact on people using public transport and may impact the accessibility of the transport system as a whole.

Reduced Passenger Satisfaction: Without real-time information, passengers become frustrated with unpredictable bus arrivals and potential service interruptions. Lower passenger satisfaction leads to lower public trust and may lead to less choice of public transport as a preferred mode of transport.

Missed Connections: Without real-time tracking, commuters can miss connections to other modes of transportation. Connection failures can increase travel time, stress, and compromise a seamless intermodal transportation experience.

Inefficient Traffic Management: Without bus location data, cities cannot effectively manage traffic and optimize bus signal time. This results in suboptimal traffic flow, congestion, and overall inefficiency of the transportation network.

Emergency Response Difficulties: During bus emergencies and accidents, the lack of real-time location data complicates emergency response efforts. Delays in emergency response may jeopardize public safety and exacerbate the severity of the incident.

Underutilization of Smart City Technology: Lack of bus tracking services means missed opportunities to make the most of smart city technology to improve transportation. Smart city initiatives may not reach their full potential to optimize urban mobility, reduce environmental impact, and improve overall quality of life.

Ensuring accurate bus tracking services in smart cities is critical to building an efficient, reliable, and easy-to-use public transportation system. By addressing these issues and implementing real-time tracking solutions, smart cities can improve the overall commuting experience, promote sustainable transportation, and contribute to the success of broader smart city initiatives.

1.2.3 Seat Availability

Lack of information about seat availability in smart cities can lead to various challenges and inconveniences for public transport users. Here are some issues related to the lack of availability services in smart cities.

Unsafe Commuting Experience: Commuters do not know about seat availability on public transportation. Passengers may lack confidence in finding a seat, which may cause discomfort and inconvenience during the journey.

Crowded Vehicles: Lack of availability information leads to congestion on buses and Passengers may be required to stand only, reducing overall comfort and potentially leading to safety concerns.

Inefficient Use of Space: Without real-time data on seat occupancy, a transportation company cannot optimize its use of space. Buses can have uneven passenger distribution, which can result in wasted space and operational inefficiencies.

Decreased Passenger Satisfaction: Passengers can become frustrated if they cannot find a seat. Lower satisfaction with public transport services can lead to lower ridership and negative perceptions of the overall public transport experience.

Inconvenience for Passengers with Special Needs: Passengers with special needs such as Elderly people or pregnant ladies and people with disabilities may have difficulty finding an available seat. Lack of accessibility can cause inconvenience and difficulty to passengers who rely on the seats.

Suboptimal Resource Allocation: A transportation company cannot allocate resources efficiently without real-time data about seat occupancy. Suboptimal resource allocation can lead to unnecessary operating costs and problems meeting passenger demand.

Uncomfortable Commuting Environment: Crowded vehicles can create an unpleasant commuting environment. Passengers may suffer from stress, decreased sense of well-being, and an overall negative perception of public transport.

Inefficient Public Transportation Planning: Public transportation planners do not have accurate data about seat demand. This can result in suboptimal route planning, lack of capacity to meet peak demand, and operational issues.

Customer Dissatisfaction: Passengers who cannot find an available seat may express dissatisfaction with public transportation. Negative feedback and decreased customer satisfaction can affect the reputation and attractiveness of public transport.

Challenges for Long Distance Travellers: Long-distance travellers may have difficulty finding a seat. Feeling unwell while traveling long distances may prevent people from choosing public transportation for long distance journeys.

Providing real-time availability information is important to improve the overall quality of public transport services in smart cities. Addressing these issues by implementing seat availability services will result in a more comfortable, efficient, and user-friendly transportation experience and encourage the adoption of sustainable transportation options.

1.2.4 Bus Interval Time

Bus interval time, or the time between consecutive bus arrivals, is an important aspect of public transportation in smart cities. Issues with bus spacing can affect the efficiency and attractiveness of bus services. Some issues related to bus interval times in smart are

Irregular Bus Intervals: Inconsistency in bus arrival times. Passengers may experience unexpected waiting times, causing inconvenience and frustration.

Overcrowded Buses: Insufficient spacing can lead to overcrowded buses. Passenger discomfort, safety concerns, and negative impact on the overall quality of the transportation experience.

Reduced Reliability: Unpredictable bus intervals reduce the reliability of public transportation. Commuters may be reluctant to rely on buses for their daily commute and may choose alternative modes of transportation.

Making Public Transportation Less Attractive: Inefficient bus spacing makes public transportation less attractive. Cities may face challenges in promoting sustainable transport, leading to increased reliance on private cars.

Increased Travel Time: The longer the bus interval, the longer the travel time. Long travel times can deter people from using public transportation, especially for time-sensitive activities.

Operational Inefficiency: Poor management of bus spacing can result in operational inefficiency. Transportation operators may have difficulty optimizing their resources, increase operating costs, and have difficulty meeting passenger demand.

Difficulty in Timed Transfers: Irregular bus intervals make it difficult for passengers to make timed transfers between different bus routes. Commuters may miss their connections, causing overall travel disruption.

Unequal Service Distribution: Bus spacing can be unevenly distributed across different routes or districts. Some regions may experience longer distances, creating disparities in service quality and accessibility.

Implications for Urban Planning: Irregular bus spacing can complicate urban planning efforts. Cities may face challenges in optimizing land use, designing efficient transportation corridors, and creating sustainable urban environments.

Negative Environmental Impact: Inefficient bus spacing increases consumption of fuel consumption and produces carbon emissions. The environmental footprint of public transport is increasing, impacting air quality and sustainability goals.

Frequency Modal Shift: People cannot switch to public transportation because bus intervals are unreliable. Cities may struggle to reduce traffic congestion and promote more sustainable transportation options. Irregular bus intervals can force people to take alternate transport arrangement which results in revenue loss.

Activity Planning Difficulty: Unpredictable bus intervals make it difficult for passengers to plan their activities. Commuters may have difficulty scheduling appointments, work, and other daily activities.

To create an efficient, reliable, and attractive public transport system in smart cities, it is important to solve problems related to bus spacing. The implementation of smart technology, data analytics and effective scheduling strategies are key elements of the solution to optimize bus spacing and improve the overall quality of public transport services.

1.2.5 Public Addressing System

The lack of a PAS⁴ in smart cities presents a variety of challenges and drawbacks that can impact the efficiency, safety, and overall user experience of transportation and public communication systems. Some of the issues related to the lack of provision of public information systems in smart cities are:

Lack of Real-Time Information: Without PAS, there will be lack of communication of real-time information to the public about traffic schedules, delays, emergencies, and other important updates. This can cause confusion for commuters and lead to inefficiencies in the transportation system.

⁴ Public Addressing System

Reduced Emergency Preparedness: Public address systems are essential for emergency notification and evacuation procedures. Without these systems, it would be difficult to convey emergency information quickly and effectively, putting public safety at risk during crises such as natural disasters or security incidents.

Accessibility Limitations: PAS is essential for providing information to passengers with visual or hearing impairments and ensuring access to public transport. The lack of such systems can make it difficult for people with disabilities to use public transport and get around.

Inefficient Traffic Management: Without the ability to communicate real-time traffic updates, route changes, and alternative options to commuters, the overall efficiency of traffic management in smart cities can decrease. This may increase congestion and delays.

Poor Passenger Experience: Lack of PAS impacts the overall passenger experience as commuters are unable to obtain important information about their journey. Passengers may experience frustration, inconvenience and dissatisfaction with public transport.

Limited Public Participation: The public information system helps engage the public by providing information about City events, cultural activities, and community announcements. The lack of these systems can lead to a lack of community awareness and participation.

The Difficulty of Coordinating Multimodal Transportation: Smart cities often integrate different transportation modes, and PAS can help coordinate these transportation modes seamlessly. Without such a system, coordination between buses, trains, trams, and other modes of transportation becomes even more difficult.

Reduced Operational Efficiency: PAS contributes to the efficient operation of transportation services by providing real-time updates to drivers, maintenance personnel, and other stakeholders. Without these systems, operational efficiency may decrease and operating costs may increase.

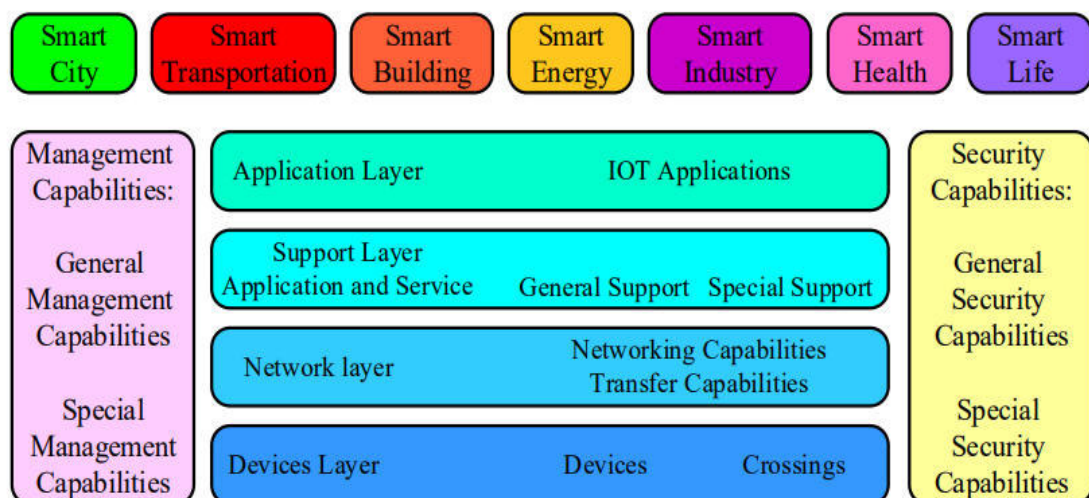
Limited Adaptability to Smart Infrastructure: In smart city environments, integration of PAS with other smart systems such as smart traffic lights and sensors is essential for optimal traffic management.

In summary, the lack of a public address system in smart cities can lead to a number of problems, including limited real-time communication, security breaches, poor accessibility, and overall inefficiency of transportation and public services.

1.3 Internet of Things

The worldwide worthiness of IOT administrations and huge information analytics have reinforced smart city ventures. These administrations have essentially moved forward the quality of human life by progressing the infrastructure and transportation framework, lessening traffic congestion, and giving waste disposal. This article presents a comprehensive audit of the integrated ICT network types, attainable potential, and critical prerequisites for the IOT worldview for smart cities. The Internet of Things is a network of physical objects, such cars, structures, and even the vital electrical appliances we use every day that are connected to one another over the internet so they may gather and share data among themselves. These "Things" can prioritize tasks, self-organize, and interact with other things without the help of humans. Each individual has more than six devices linked to the Internet. The idea behind the Internet of Things is to make the web even more pervasive and immersive. Additionally, by facilitating simple access to and interaction with a wide range of devices, for example, household appliances, monitoring, security cameras, sensors, displays, actuators, and automobiles. The Architecture of IOT is shown.

Figure 1.3: Architecture of IoT



Source: Moazzami and Majid(2021)

The IoT is characterized by a layered architecture that organizes and structures the flow of information and data. This architecture enables efficient communication and coordination between the various components of an IoT system. The hierarchical structure of IoT information provides a systematic approach to designing, implementing, and managing IoT systems. A typical hierarchical structure for IoT information is shown above, It has four layers:

Application Layer: The application layer in the context of the IoT refers to the top layer of the IoT architecture that is responsible for communication and interaction between IoT devices and applications. This layer plays a critical role in facilitating data exchange, data processing, and application-specific functionality. Important aspects of the application layer in IoT are: Data processing and analysis: The application layer processes and analyses data generated by IoT devices. The goal is to gain valuable insights from raw data in order to make informed decisions. Application-specific protocols: IoT applications often require specific protocols for communication. The application layer defines the rules and conventions for exchanging data between devices and applications to ensure interoperability. APIs⁵ and Web Services: APIs and web services are used at the application layer to enable communication between the various components of the IoT ecosystem. This enables third-party developers to create applications that interact with IoT devices and data. In summary, the application layer of IoT is responsible for managing the communication, processing, and functionality that defines how IoT devices interact with applications and services.

Support Layer: In the context of IoT architecture, the support layer includes various components and functions that provide support services essential to the smooth operation of the IoT ecosystem. This layer typically includes the overall infrastructure, management, and elements that contribute to optimizing IoT deployments.

Network Layer: The network layer of the IoT architecture plays a critical role in facilitating communication and data exchange between a myriad of interconnected devices. These are the infrastructure and protocols that allow devices to connect, share information, and operate consistently within the IoT ecosystem. The key components and aspects of the network layer in an IoT architecture are: Connection Protocols: The network layer involves the selection and implementation of connection protocols that

⁵ Application Program Interfaces

allow devices to communicate with each other and the broader network. Common IoT protocols include MQTT⁶, CoAP⁷, and HTTP⁸/HTTPS. Wireless Communication Technology: IoT devices often rely on wireless communication technology for flexibility and ease of deployment. These may include Wi-Fi, Bluetooth, Zigbee, Z-Wave, cellular networks (3G, 4G, 5G), and LPWAN⁹.

Device Layer: The lowest layer of an IoT architecture is often referred to as the device layer or perception layer. This layer is the foundation of the IoT ecosystem and consists of physical devices or "things" that collect data, sense the environment, and interact with the real world. Devices in this layer are responsible for collecting information and passing it to higher layers for processing and analysis. The key components and aspects of the lowest layer of the IoT architecture are: Sensors and Actuators: Sensors are devices that detect and measure physical properties such as temperature, humidity, pressure, light, and movement. Actuators, on the other hand, allow devices to perform actions based on the data they receive, such as turning on/off, adjusting settings, or triggering other physical processes. IoT devices are often embedded systems with embedded computing capabilities. These systems can process data locally, perform simple calculations, and control connected sensors and actuators. Microcontrollers and microprocessors provide the computing power needed to operate IoT devices. They control device functionality, handle data processing, and manage communication with other devices or networks. The lowest level device is equipped with a communication module that allows it to send and receive data. The data collected by devices at this layer is transferred to the network layer for further processing, analysis, and decision-making.

Internet of Things architecture is critical to the effective deployment and functioning of IoT systems. It provides a structured framework for connecting and managing a variety of devices and sensors, enabling seamless communication, data sharing, and intelligent decision-making.

⁶ Message Queuing Telemetry Transport

⁷ Constrained Application Protocol

⁸ Hypertext transfer protocol

⁹ low-power wide area network

1.4 Role of Machine Learning

It is common knowledge that many individuals would relocate from rural to urban areas as a result of industrialization. The transition from rural to urban areas will bring about a number of challenging issues, including increased traffic, pollution, stress on the fundamental infrastructure, and challenges with waste management. Finding effective, ethical, and sustainable solutions to improve lives will become increasingly important as a result. This is where the still-evolving idea of a "smart city" comes into play. SCs¹⁰ are designed to effectively manage energy use, preserve the environment, raise the economic and living standards of their residents, and improve their capacity to effectively employ and adopt contemporary information and communication technologies. SCs have cutting-edge sensors that manage city resources and contribute significantly to the gathering of crucial data that can subsequently be used in a variety of applications. The different types of sensors include electronic ones like parking and speedometer sensors, chemical ones like oxygen and carbon dioxide sensors and catalytic bead sensors, biosensors for identifying biomedical components, and smart grid sensors for effective power generation, transmission, and distribution from the point of generation to the users. Most of the data flowing from the sensors is wasted since there are no set standards or processes for extracting potentially useful information. To analyse massive data as effectively as feasible, consider long-term goals, and arrive at the best or nearly best conclusions, machine learning (ML¹¹) can be utilized.

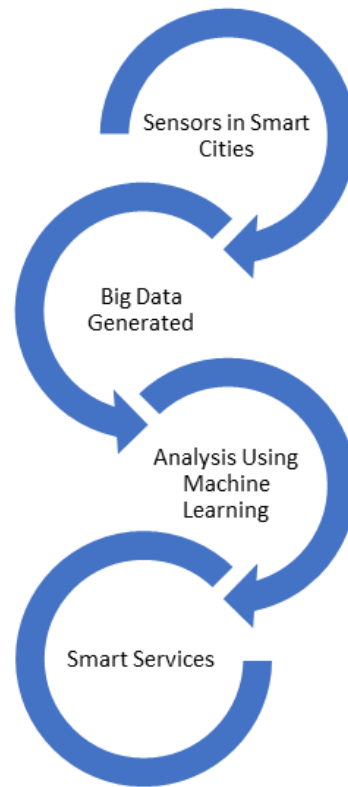
A subfield of AI¹², ML enables computers to automatically learn by exploring the data made available and using it to gain experience and learn new things without being explicitly programmed. Arthur Samuel first used the phrase "Machine Learning" in 1959. Since the development of the first artificial neural network for computers, the perceptron, and the use of the "nearest neighbor" algorithm to solve the nearest neighbour problem, ML has advanced from being a simple tool for pattern recognition to one that can carry out complex tasks and incorporate deeper domains of AI.

¹⁰ Smart Cities

¹¹ Machine Learning

¹² Artificial Intelligence

Figure 1.4: Steps of Applying Machine Learning Techniques



A system that can learn from past experiences and develop on its own is required due to the abundance of data generated by sensors and smart devices from SCs that cannot be inspected individually. As a smart city application's context and operational environment change, it needs a generic, dynamic, and ongoing learning mechanism. Therefore, it is essential for increased efficiency to investigate the possibilities of ML and big data in the creation of individualized services in SCs. The ML technique is rapidly changing due to progressions in calculations, information collection procedures, computer networks, modern sensors, and IO gadgets, and interest in self-customization to client conduct.

The primary goal of machine learning algorithms is to accurately understand data that has never been "seen" before and make predictions that go beyond the training samples, similar to data from the real world. By increasing the amount of training data and strengthening their ability to learn, the algorithms' accuracy and precision can be further improved. Based on the type of learning "signal" or "feedback" that a learning system is able to receive, machine learning algorithms can be roughly divided into three groups.

1.4.1 Supervised

Giving training data that has previously been "known" or "labelled" with the proper response and consists of N input-output pairs (X, Y) is how supervised learning functions. The ANN¹³ then generates an output Z for each unknown X , which is then compared against Y using an error (cost or distance) function. Finally, an iterative process is used to minimize this mistake. Image Classification: Training with image/label datasets are examples of supervised learning methods. A new image is then presented later with the hope that the computer will pick up on the new object. Regression: Giving the system marked historical data so it can forecast the future result of an identical circumstance.

1.4.2 Unsupervised

Using unsupervised learning methods, it self-organizes and finds hidden patterns in unlabelled input data to create neural networks. It can analyse data without sending an error signal so that the potential fix can be assessed. Unsupervised learning can occasionally be useful since it allows the algorithm to search the past for patterns that weren't previously taken into account. Unsupervised learning is necessary because manually inspecting huge datasets like those for speech recognition is highly expensive. Clustering is a very basic but well-known example of unsupervised learning.

1.4.3 Semi Supervised

This category is a hybrid of the previous two. The algorithm is trained on a dataset that contains both labelled and unlabelled data. It works by taking enormous amounts of input data and labelling only a subset of it as training data. Reinforcement learning, a related strategy, provides feedback to guide the computer programme in interacting with a dynamic environment. In this approach, a model is deployed using a small set of labelled samples and a larger set of unlabelled samples. The goal is to use labelled data to make predictions about unlabelled data and use the additional information to improve model performance.

¹³ Artificial Neural Network

1.5 Machine Learning Algorithms

To avoid imprecise or erroneous predictions, the data collected/generated must go through pre-processing, merging, modifying, and cleaning (removing null values). The computational intensity and speed of a specific technique are two significant characteristics to consider while employing ML techniques. The best algorithm is chosen based on the user application and should be fast enough to track changes in the input data and provide the desired output in a reasonable amount of time. ML algorithms create a mathematical model using sample data, known as "training data," on which to make predictions or choices. The training phase of supervised ML classifier development involves training a specific classifier from a set of labelled data. As the size of the training data increases, so does the performance of the classifiers. Some of the most popular ML algorithms are detailed further below.

1.5.1 Bayes Net

A Bayesian network, also known as a Bayes net or belief network, is a graphical model that represents probabilistic relationships between a set of variables. Bayes net are not algorithms themselves, but they serve as a framework for thinking under uncertainty. Several algorithms are concerned with learning the structure and parameters of Bayes net from data. The two main types of algorithms related to Bayes net are:

Structure Learning Algorithms: These algorithms determine the graphical structure of Bayes net and specify dependencies and conditional relationships between variables.

Parameter Learning Algorithm: Once the structure of the Bayes net is known, the parameter learning algorithm estimates the parameters (conditional probability distributions) associated with each node based on its parents in the network.

1.5.2 Naive Bayes

Based on Bayes' theorem, a naive Bayes classifier is a probabilistic classifier that works by assuming that no pair of features are dependent. Naive Bayes is a simple but powerful machine learning algorithm based on Bayes' theorem and the assumption of independence between features. Despite its simplicity, Naive Bayes is often effective and computationally efficient, so it is often used in a variety of classification tasks. It is particularly suitable for text classification and spam filtering.

1.5.3 Logistic

Logistic regression is a machine learning algorithm commonly used for binary classification tasks, where the goal is to predict whether an instance belongs to one of two classes. Despite its name, logistic regression is more of a classification algorithm than a regression algorithm. Logistic regression is a fundamental machine learning algorithm that is widely used in various applications such as medical diagnostics, spam detection, and credit scoring due to its simplicity, interpretability, and effectiveness. Although it is designed for binary classification, it can be extended to handle multiple classes through techniques such as one-vs-rest regression and softmax regression.

1.5.4 SMO

SMO¹⁴ is a machine learning algorithm designed to train SVMs¹⁵ in supervised learning. SVM is used for classification and regression tasks, and SMO is a specific algorithm used to efficiently solve the optimization problems associated with training these models. Although SMO is an important algorithm for SVM training, there are alternative approaches and optimizations to solve SVM problems, such as the widely used libsvm library that implements more general optimization techniques. Still, understanding SMO provides insight into the support vector machine training process.

1.5.5 IBK

IBk¹⁶ is a machine learning algorithm used for classification and regression tasks. It is the part of the family of k-NN¹⁷ algorithms, where the prediction of a new instance is based on the majority class for classification or mean for regression of the k-nearest neighbors in a function space. The main features of the IBk algorithm is Instance-based learning. This means that no explicit model is created during training. Instead, save the training instance and use it to make predictions for new instances. In K-NN Predictions for new instances are determined by examining the class labels for classification or values for regression of the k-nearest neighbors in the training data set. Small values of k gives the model that is more flexible and sensitive to noise, and large values of k gives the model that is smoother and less sensitive. Regression uses the average of the

¹⁴ Sequential Minimal Optimization

¹⁵ Support Vector Machines

¹⁶ Instance-Based k-Nearest Neighbors

¹⁷ k-Nearest Neighbors

k nearest neighbor target values as the prediction. IBk can be computationally expensive, especially for large datasets, as it must calculate the distance for each prediction. It is often more efficient when the dataset is small. IBk performance can be sensitive to feature scaling. Therefore, it is often recommended to normalize or standardize features to obtain a similar scale. IBk is a simple but effective algorithm, especially in situations where the decision boundary is complex and not easily captured by parametric models. It is widely used in various fields such as pattern recognition, classification, and regression.

1.5.6 K Star

K Star was developed in 2009. K Star was originally implemented as part of DiPro toolset for generating counter examples in probabilistic model checking. K Star A directed search algorithm also called as K*. It Finds the k shortest paths between the given pair of vertices in the given directed weighted graph. K Star works on the fly. This means that the graph does not have to be made explicitly available and stored in main memory. K Star can be also be controlled using a heuristic function.

1.5.7 Multi Class Classifier

A multiclass classifier is a type of machine learning algorithm that can assign instances to one of three or more classes. Unlike binary classifiers, which distinguish between two classes (such as positive or negative), multiclass classifiers handle scenarios where there are multiple possible classes. Some of the common Multi Class algorithms are Support vector machine, Random Forest, K Nearest Neighbours, Neural Networks and Decision Trees. The choice of algorithm often depends on factors such as the size and type of the dataset, computational efficiency, and the desired interpretability of the model.

1.5.8 Random Forest

A decision tree-based supervised machine learning approach called RF¹⁸ depends on values from a random vector that is sampled separately and with the same distribution across all of the trees in a forest. By averaging the results, this ensemble method lowers over-fitting and bias-related error, leading to superior outcomes. Random Forest is a powerful and versatile machine learning algorithm that belongs to the ensemble learning category. Ensemble learning combines the predictions of multiple models to create a more robust and accurate model. Random forests are particularly effective for both classification and regression tasks. The main features and characteristics of the Random Forest algorithm are: Ensemble of Decision Trees: Random Forest is an ensemble of Decision Trees. A decision tree is a discrete model that makes predictions based on a series of hierarchical decisions. Random forests create multiple decision trees and combine their predictions during the training phase. During the training process, Random Forest randomly selects a subset of the training data (with permutations) to train each decision tree. This process is called bootstrapping. Additionally, at each decision point in the tree, a random subset of features is also considered. Random Forest uses a technique called bagging, where each decision tree is trained independently on a different subset of the data. The final prediction is determined by aggregating the predictions of all trees. By training multiple decision trees on different subsets of data and features, random forests become more robust and less prone to overfitting compared to a single decision tree. Overfitting occurs when a model learns the training data well enough but is unable to generalize to new, unseen data. Random Forest provides a measure of feature importance. Analysing the contribution of each feature across multiple trees can help determine which features have the greatest impact on predictions. The training of individual decision trees in a random forest can be performed in parallel, resulting in a scalable algorithm that can efficiently process large amounts of data. Random forests tend to be less sensitive to outliers in a dataset. Because each tree is trained on a subset of the data, the impact of outliers is reduced. Random Forest has been implemented in various machine learning libraries such as Scikit-Learn in Python, making it highly accessible and widely used.

¹⁸ Random Forest

1.5.9 Random Tree

Random Tree is a term often associated with two different machine learning algorithms, Random Forest and Highly Randomized Trees (Extra Trees). Both algorithms fall into the category of ensemble learning and are used for classification and regression tasks. Both Random Forest and Extra Trees are powerful algorithms that leverage the concept of ensemble learning to improve predictive performance. They are widely used in various applications such as classification, regression, and feature importance analysis. The choice between random forests and extra trees may depend on the specific properties of your data and the desired trade-offs between computational efficiency and model accuracy.

1.6 Applications of Machine Learning

Machine learning algorithms are applied in a variety of fields to provide solutions to complex problems and make data-driven predictions. Here are some notable applications of machine learning algorithms:

1.6.1 Smart Traffic Management and Transportation

Traffic bottlenecks disrupt people's lives on a daily basis. It results in lengthier trips due to traffic congestion, more pollution, and huge economic losses due to delays and other transportation issues. Thus, one of the primary problems in modern cities is Smart Mobility, which focuses on providing sustainable transportation systems and logistics to allow for seamless urban traffic and commuting through the use of mostly information and communication technology. They also include ways that employ personal information to make useful recommendations for small-scale personal management tasks such as looking for free parking. To decide control actions, certain traditional control approaches such as SFC¹⁹ and traditional AI techniques based on historical data (recorded similar occurrences) such as case-based reasoning and rule-based systems were developed.

¹⁹ Static Feedback Control

However, these technologies have shortcomings such as difficulty dealing with traffic network dynamics and the lack of a learning mechanism to cope with unknown events and automatically update their model. The advancement of ML and DL²⁰ cleared the way for a generic and adaptable method of developing intelligent and adaptive traffic control systems.

1.6.2 Combating Pollution

Today, worry over pollution is on the rise. It is a significant threat to the environment as well as a primary risk factor for a number of illnesses like lung and skin diseases. Large, intricate, and varied time-series data have been subjected to a number of Machine Learning approaches during the past ten years to produce estimates for air pollution. The conventional method for predicting air quality utilized mathematical and statistical methodologies, where data was coded using differential equations and a physical model. These methods were time-consuming and only partially accurate because they couldn't foresee the extremes, such as the pollution maximum and minimum cut-offs.

1.6.3 Public Safety

Massive amounts of data are being collected in SCs by sophisticated sensor networks, which has increased the difficulty of protecting that data from intrusions and made the implementation of effective cybersecurity measures necessary. SCs are extremely susceptible to online risks including data theft and illegal device access. By employing techniques like facial and speech recognition, recognizing criminal patterns around the city to create better public policies, and detecting fraud, ML can also be utilized to facilitate the job of law enforcement authorities.

1.6.4 Smart Grids and Machine Learning

Energy management and conservation are regular 24-hour concerns for consumers and energy service providers. With the aid of digital communications technology, a smart grid is an energy network that enables the two-way flow of data and electricity, allowing us to detect and respond to changes in demand as well as a number of other difficulties. It guarantees effective electricity transmission, prompt power restoration during power

²⁰ Deep Learning

outages, increased customer-owner power generation system integration, including the integration of renewable energy systems, and enhanced security against electricity theft. Applying ML techniques also strengthens the power system's endurance and adaptability and improves its readiness to handle emergencies like natural and man-made disasters.

1.7 Role of Artificial Intelligence

The field of transportation has benefited greatly and substantially from AI. The solutions provide vehicle and driver safety through the use of autonomous vehicles, traffic control, optimal routing, and logistics. The data produced by the devices deployed in vehicles employing AI technology is used to build ITS²¹. Four transportation-related subsystems—the Intelligent Traffic Management System, the Intelligent Public Transport System, the Intelligent Safety Management System, and the Intelligent Manufacturing & Logistics System—are the focus of the current study.

Figure 1.5: Old Transportation System,



Source: Iyer (2021)

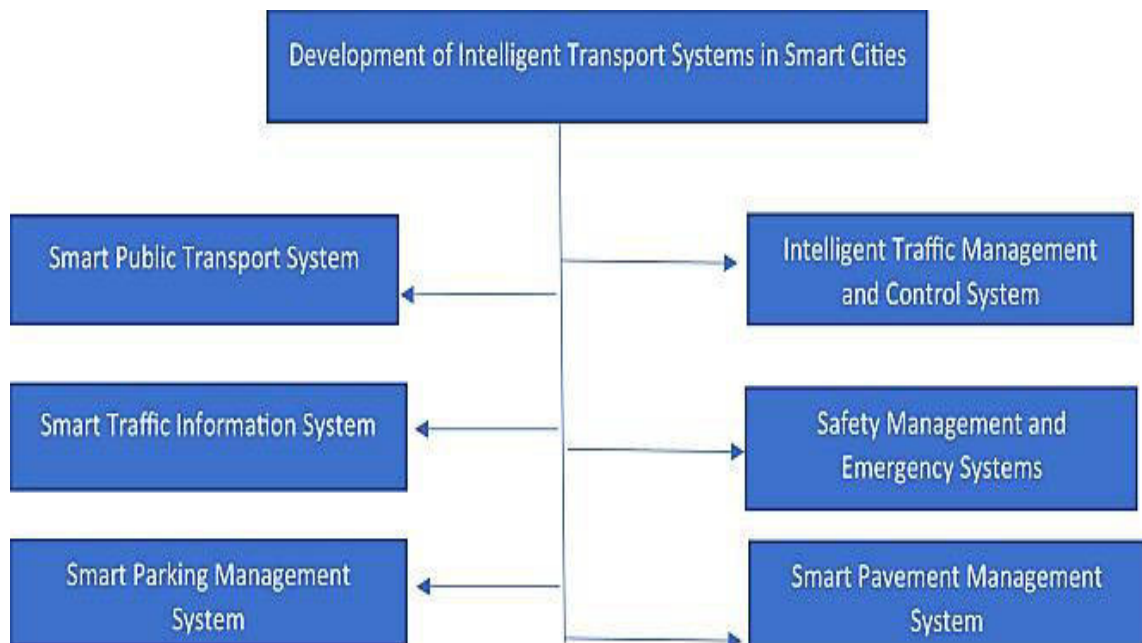
²¹ Intelligent Transportation System

Old traditional transportation systems refer to the transportation methods and infrastructure that were prevalent before the advent of modern, technologically advanced transportation. These systems varied by region and historical period and were characterized by their dependence on traditional techniques and means of propulsion.

The Above figure shows old TMS²² Where Every thing was controlled manually right from fleet management, Route determination, Route Scheduling and Inbound outbound logistics. Manual methods are time consuming, in efficient and prone to errors. Before advanced technology became widespread, such systems relied on manual processes and simple tools, such as Manual record keeping, Schedule maintenance, Fuel management, Driver management, Fixed routes and Time table, Paper schedules, Manual Adjustments.

The transition from traditional fleet management and route planning to modern ones has significantly improved efficiency, responsiveness, and overall transportation services.

Figure 1.6: Sub-systems in Intelligent Transportation System



Source: Agarwal, Gurjar and Birla (2015)

²² Transport Management System

The above figure shows the development steps of Intelligent Transport Systems in Smart Cities. There are six important components:

Smart Public Transport System: Smart public transportation systems use advanced technology and data-driven solutions to improve the efficiency, reliability, and user experience of public transportation services. The aim is to make urban mobility more sustainable, accessible and comfortable for passengers, while optimizing the operation of public transport networks. The main components and features of a smart public transportation system are:

- Real-time tracking and monitoring
- GPS location tracking of public transport (buses, trains).
- Predictive Analytics and Arrival Time Estimation
- Contactless Ticketing and Payments
- Automatic Fare Collection

Deploying smart public transportation systems requires collaboration between transportation authorities, technology providers, and communities. Integrating new technologies, data analytics, and user-centred design principles will play a key role in transforming traditional public transport into more efficient and passenger-friendly systems.

Intelligent Traffic Management and Control System: It has been famous that machine learning algorithms are regularly utilized to figure out traffic jams and arrange routes. A city-by-city survey of the selection of AI to address transportation concerns shows that the larger part of developed countries have rapidly grasped these innovations. Given that it incorporates venture and long-term considering the portion of the best administration, this selection requires the support of the specific organizations and the administration. Due to two factors, certain businesses and governments are still on the fence about using AI: either they are worried about the risks involved, or the general level of technology adoption in these nations is low.

AI helps with transportation problems by recommending alternate routes and monitoring traffic signals in real-time amid heavy traffic. Through effective traffic management, environmental pollution can be reduced, and sustainable cities can be

built. AI offers arrangements for foreseeing weather and traffic designs, managing streets, and alarming on-duty police officers. Sometime recently beginning their trip, these frameworks back drivers, commuters, and people on foot. It is vital to have innovative support to form a viable open transportation framework that helps in arranging and decision-making.

The frequency of accidents on the roads has decreased thanks to AI, which also warns drivers about road safety and anticipates accidents based on the state of the roads. When the transportation sector is productive, an economy may function successfully. It is important to accomplish this by developing secure transportation systems with the use of AI technologies. Using AI during the automobile production process For better results, sensors, cameras, and other technology have been used in this sector. Some of the automaker's built-in AI technologies are already standard equipment for both commercial and passenger vehicles.

Smart Traffic Information: It uses technology and data to optimize traffic flow, improve road safety, and provide real-time information to both drivers and traffic authorities. These systems use a variety of technologies and data sources to improve overall transportation efficiency. The main components and features of an smart Traffic information are:

- Interactive displays
- Announcements on public addressing system.
- Provide passengers with real-time information, service updates, and emergency notifications.

Smart traffic information systems play a key role in improving mobility across cities, reducing congestion, and increasing the safety and efficiency of transportation networks. Integration of technology, data analytics, and communication systems is key to the success of these smart traffic management solutions.

Safety Management and Emergency Systems: Safety management and emergency systems are an integral part of intelligent transportation solutions and contribute to the security and resilience of the entire transportation network. The main aspects and features of security management and emergency systems in intelligent transportation are:

- Real-time monitoring and monitoring
- Incident Detection and Response
- Emergency Communication Systems
- Integrated Emergency Services like police, fire, medical
- Dynamic Signs and Warnings
- Intelligent Traffic Signal prioritization for Emergency Vehicle like Ambulance and VIP Vehicles.
- Surveillance Drone Technology equipped with cameras and sensor to provide real-time surveillance and assessment, especially in hard to reach areas.
- Implement robust cybersecurity measures to protect transportation systems from cyber threats that can compromise security and disrupt operations.
- Encourage the community to actively participate in safety reporting through mobile apps, social media, or other channels.
- Provide a platform for citizens to report incidents and dangers.
- Developing evacuation plans and conduct simulations to ensure effective response and evacuation procedures during emergencies and natural disasters.

Integrating safety management and emergency systems with intelligent transportation technologies makes urban transportation networks more resilient and responsive, contributing to safer and more secure mobility for everyone.

Smart Parking Management System: Smart parking management systems use technology to improve the efficiency and user experience of parking facilities. These systems use a variety of technologies and data-driven solutions to optimize parking space utilization, reduce congestion, and improve the overall parking experience. The key components and features of smart parking management are:

- Real-time Parking Availability Using sensors and cameras.
- Parking Guide Mobile App to provides real time parking information.
- Smart Parking Meters to accept electronic payments and mobile payments.
- License Plate Recognition for vehicle identification and payment.
- Reservation System to Allow users to reserve parking spaces in advance.
- Dynamic Pricing Models based on real time demand or peak times.

Smart Parking Management System makes the parking process more convenient and efficient, contributing to more efficient use of urban space, reducing traffic congestion, and increasing customer satisfaction. The integration of technology and data analytics plays a key role in transforming traditional parking lots into smart, user-friendly spaces.

Smart Pavement Management System: Smart pavement management systems are technology-driven solutions aimed at monitoring, assessing and managing the condition of pavements and infrastructure. These systems use a variety of technologies and data analytics to enable efficient maintenance, improve road safety, and extend the lifespan of transportation infrastructure. The key components and features of a smart pavement management system are:

- **Pavement Condition Monitoring:** Sensors and imaging technology, such as laser scanners and cameras, are used to assess road surface conditions. This includes detecting cracks, potholes, ruts, and other signs of damage.
- **Data Collection and Analysis:** Collection of data on road conditions, traffic loads, weather conditions and other relevant factors.
- **IoT Sensors and Smart Infrastructure:** Integrate IoT sensors into pavement or road infrastructure to monitor conditions in real time. These sensors can provide continuous data on temperature, humidity, and other factors that affect pavement health.
- **Machine Learning and Predictive Analytics:** Implement machine learning and predictive analytics algorithms to predict future conditions of pavements based on historical data. This enables proactive maintenance planning and resource allocation.
- **Automated Inspection and Evaluation:** Introduce automated systems for pavement inspection to reduce the need for manual inspection. Utilizing drones equipped with cameras and sensors enables efficient and comprehensive evaluation.
- **Dynamic Maintenance Scheduling:** Develop algorithms to dynamically plan maintenance activities based on real-time conditions, budget constraints, and priorities. This ensures optimal use of resources and minimal disruption to traffic flow.

- **Life Cycle Cost Analysis:** To find out economic viability of various maintenance strategies life cycle cost analysis is performed. This also takes into account the cost of renovations, rebuilding, and ongoing maintenance.
- **Remote Monitoring and Control:** Allows remote monitoring of road conditions and control of maintenance processes. This allows authorities to make real-time decisions and respond quickly to emerging issues.
- **Environmental Aspects:** Consideration of environmental factors such as climate, soil conditions and pollution when assessing road conditions.
- **Public Participation and Reporting:** Public participation via mobile app or his website to report road problems. Providing real-time updates on maintenance activities and progress helps build transparency and trust in the community.

Intelligent pavement management systems play a key role in ensuring the durability, safety and sustainability of road infrastructure. Using technology and a data-driven approach, transportation authorities can increase pavement resiliency, reduce maintenance costs, and improve the overall performance of road networks.

Chapter - 2

Review of Literature

- 2.1 Introduction
- 2.2 Enhancing Public Transport Experience
- 2.3 Commuting and Traffic Congestion Issues
- 2.4 Technologies and Transportation Systems
- 2.5 IoT Based Traffic Prediction Models
- 2.6 Artificial Intelligence and Traffic Management

2.1 Introduction

A literature review is a critical and comprehensive analysis of existing literature and scholarly works on a specific topic. It serves as a foundation for research by providing an overview of relevant theories, methodologies, and findings related to the research question or area of interest. Through a literature review, researchers identify gaps in current knowledge, assess the credibility and quality of existing studies, and establish the context for their own research. It involves a systematic search, evaluation, and synthesis of published materials such as academic articles, books, reports, and other relevant sources. A well-executed literature review not only demonstrates a researcher's familiarity with the existing body of knowledge but also contributes to the development of new insights and understanding within a particular field or discipline. Literature review helps in finding gaps in research already done, identifies conflicts in previous studies and finds out open questions left for other researchers.

Fourth Industrial revolution is confluence of different technologies like IoT, Data science, Artificial Intelligence and Machine Learning, which can completely change the outlook of public transport information system. Artificial Intelligence technology has human like brain and self-correcting ability. Machine Learning is subset of Artificial Intelligence that imitates the human learning process. There is need to explore use of Artificial Intelligence and Machine Learning in solving public transport information system problems with help of IoT. The research needs to be done on different Artificial Intelligence and Machine Learning algorithms with the help of IoT and GPS in the following areas.

1. Enhancing public transport experience.
2. Commuting and traffic congestion issues associated with smart cities
3. Technologies and Transportation Systems in Smart Cities
4. IoT based traffic prediction models for smart cities
5. Artificial Intelligence and Traffic Management

2.2 Enhancing Public Transport Experience

Improving the public transport experience is critical to promoting public transport use, reducing traffic congestion and making cities more sustainable. Enhancing the public transport experience requires a multi-pronged approach involving infrastructure development, technology, policy and community engagement. When implemented

effectively, these strategies can make public transport a more attractive and sustainable option for commuters. Some of the ways to improve your public transportation experience are by using IoT, Artificial Intelligence and Machine Learning algorithms to display real time bus arrival time information, bus interval information and seat availability.

Abigail, Jorge, Franklin, Angel and Santiago (2019) presented data framework for the observing and control of urban open transportation stations. they proposed clients will be informed through web administrations, smartphone application, counting data at the transport halt around courses, area of stops and times of open transport. Author recommended radio recurrence gadgets utilization for remote communication between the car and the transport station. portable unit was utilized to send data to a database show on a WEB server, where it was prepared and sent to the station for its socialization with the client. In his paper an sound framework that gives data on the course through a speaker found within the station and within the versatile unit was actualized, encouraging the utilize of this benefit to clients with visual inabilities and ignorant populace. The framework was tried in three stops and two portable mobile units. The quick reaction of the framework with an approximate time of up to 1second from the transport enrollment at the halt to the updating of the information on the site.

Abduljabbar (2019) gave a comprehensive overview of the applications of artificial intelligence in transportation systems. The study explores various AI technologies and their potential impact on enhancing transportation efficiency, safety, and sustainability. By examining AI-driven solutions, the authors contribute valuable insights into the evolving landscape of intelligent transport systems, shedding light on the integration of cutting-edge technologies to address contemporary transport challenges.

Cai (2009) hunts through Dynamic programming for adaptive traffic signal control. The study explores advanced algorithms to optimize traffic signal control dynamically. By employing innovative computational techniques, the research aims to enhance traffic signal efficiency, ultimately improving traffic flow and reducing congestion in urban areas.

Collotta (2018) study explores the intersection of green technologies and intelligent transportation, emphasizing sustainability and energy efficiency. By incorporating smart solutions, the research aims to create environmentally friendly transportation systems, contributing to both energy conservation and ecological preservation.

Dresner (2008) research introduces a multiagent approach to Autonomous Intersection Management, as published in the Journal of Artificial Intelligence Research. The study explores advanced techniques for managing traffic flow at intersections using autonomous agents. By leveraging multiagent systems, the research presents innovative solutions for optimizing intersection management, enhancing traffic efficiency, and reducing congestion, thereby contributing to the development of intelligent traffic control systems.

Dou (2015) has designed three independent interconnected systems using server, Android application and public transport bus distributed network on-board device. The communication between three components was provided using protocols such as HTTP²³, WSDL²⁴, SOAP²⁵ and SSL²⁶. Author studied the various parameters for analysis like bus travelling time, dwelling time at bus stops, Passenger travel habits, and observed the execution of schedule time table. Author used historical parameters like route distance, travelling time, bus speed, dwelling time at each stop. Root mean square error method was used by author to measure predicted accuracy of travel time in comparison with bus timetable by using historical based data model and linear regression model.

Ersoy (2015) studied intelligent transportation systems and their applications in the road transportation industry in Turkey. The research investigates the integration of intelligent transportation solutions in the Turkish road transportation sector. By analyzing applications and challenges, the study provides valuable insights into the potential benefits and obstacles faced in implementing intelligent transportation systems, essential for optimizing road transport operations in the country.

²³ Hyper Text Transfer Protocol

²⁴ Web Services Description Language

²⁵ Simple Object Access Protocol

²⁶ Secure Socket Layer

Ferrara (2018) focused on the fundamentals of traffic dynamics within the context of freeway traffic modeling and control. The study delves into the intricate dynamics of traffic flow, providing a detailed analysis of fundamental principles. By examining traffic dynamics, the research aids in enhancing our understanding of freeway behavior, paving the way for improved traffic modeling techniques and control strategies. The findings contribute to the broader field of traffic engineering, informing the development of advanced traffic management systems and solutions.

Ghasem-Aghaee (2019) explored the synergy between simulation, intelligence, and agent-based systems. The study delves into the integration of these technologies, offering insights into their collaborative potential. By examining current and future developments in artificial intelligence, the research contributes to the exploration of innovative approaches for creating intelligent systems. The study's findings provide a foundation for advancing AI-driven solutions in simulation and agent-based modeling, shaping the future of intelligent transportation systems.

According to **Iyer (2021)** Artificial intelligence can perform cognitive functions like seeing, thinking, learning and problem-solving which people can performing easily. From past twenty years internet is generating massive information in all over the world which has created solid foundation for the AI. There has been a colossal advantage to governments and businesses by handling this information using advanced algorithms in the recent past. The strong growth of Automation, IoT, Robotics, computer vision, Natural language processing and Machine Learning algorithms, have enabled the growth of AI. According to author AI in the current form has the ability to solve problems in real time transport, vehicle scheduling, time management, managing design, operation, and administration of logistical systems and freight transport.

Jacobsen (2015) presented an conference framework discussion of the Digital System Design. The study focuses on developing a framework for aggregating demand response programs in Europe. By creating efficient aggregation methods, the research contributes to the effective management of demand response initiatives, essential for energy conservation and sustainable energy practices.

In the 2nd International Conference on IoT, Big Data, and Security **Jacobsen (2017)** presented a secure SMS based vehicle tracking system. The study focuses on creating an efficient and secure vehicle tracking system using SMS communication. The research aims to enhance vehicle tracking capabilities, contributing to transportation security and management.

Khallouk (2018) researched traffic flow at un-signalized junctions with crossing pedestrians. The research delves into the complex dynamics of traffic interactions involving pedestrians, shedding light on the challenges faced in un-signalized junctions. By analyzing pedestrian influence on traffic flow, the study provides valuable insights for traffic engineers.

Kharchenko (2019) tried to predict trolley bus arrival time with the help of historical average, Kalman filtering technique, and Google Maps API. He focused on choosing an efficient model and framework which can be used to for real-time data acquisition and which can be easily implemented to improve public transport services by using the GPS data and IoT data. He designed information service infrastructure model for public transport. He also investigated model's performance and their effectiveness with real time data. He has also explained that combination of distance from Google Maps API with Kalman filtering and average travel speed gave the best arrival time predictions for low speed urban transport.

Kuberkar and Singhal (2020) found that many countries including India are developing smart cities to reduce increasing burden on public transport systems. he found various issues of overcrowding, delayed services and travel dissatisfaction of commuters due to lack of infrastructure. Research paper also studied the need of AI powered Chatbot by the citizens of the smart city for delivering automated anytime, anywhere, public transport information services. Author also favored an extended UTAUT²⁷ model to measure the adoption intention.

²⁷ Unified Theory of Acceptance and Use of Technology

Lee (2007) have shown in the National Cooperative Highway Research Program Report 572, published by the Transportation Research Board of the National Academies, offers a comprehensive overview of traffic management strategies and techniques. The report covers a wide array of topics, including traffic control, modeling, and safety measures. By synthesizing expertise from various contributors, the report serves as a valuable reference for transportation professionals, policymakers, and researchers, providing essential guidance for improving traffic management practices and ensuring safer, more efficient road networks.

Liu (2014) introduced Green Optical Character Recognition, an energy-efficient optimal clustering routing protocol, published in the Computer Journal. The study focuses on optimizing routing protocols in energy efficient wireless sensor networks. By employing optimal clustering techniques, the research aims to conserve energy in sensor networks, extending their operational lifetime and enabling sustainable data communication practices essential for various applications, including intelligent transportation systems.

Misbahuddin (2015) presented his research paper at the 12th International Conference on High-Capacity Optical Networks and Enabling/Emerging Technologies in Islamabad, Pakistan. He explored dynamic road traffic management for smart cities using IoT. The study focuses on leveraging IoT technology to create an adaptive traffic handling system. Author research was focused on to increase flow of traffic and reduce traffic congestion by integrating real-time data and smart city technologies, which contributes to the development of efficient and responsive urban transportation systems.

Mar and Sheng (2018) had suggested, incorporating data collection and sharing of services using IoT which has a potential to transform the communities into smart Cities. Author has suggested many advantages of smart city. According to him smart city concept can improve traffic, lower pollution levels, provide real time weather prediction and provide overall better quality of life to all it's citizens. Author proposed cloud computing should be used to collect and store city data using mobile environmental sensing platform. Raspberry Pi 3 based sensing and geolocation module was designed by author to sense temperature, air quality and humidity. PhpMyAdmin local database was used to store environmental parameters, and uploaded to Thing Speak cloud database in real-time. Author had taken help of various Graphical visualization and

monitoring tools for further information and data analysis. Author has designed prototype model to investigate whether developed system is working properly or not using the conventional static sensing and new mobile sensing. It was found that there is a difference of 6.93% between the static and the mobile sensing method, which was considered to be small value.

Mukti and Prambudia (2018) have written about Jakarta digital transportation system. he highlighted the various obstacles in governing such systems. The research examines the complexities of managing digital transportation services within the framework of a smart city. By analyzing governance challenges, smart city planners and policymakers are given valuable insights.

Prakash and Anudeep (2018) explained reduction in passenger waiting time by constantly updating real time passenger information system in their IoT²⁸ research paper. The proposed framework centers upon the current area of transport. It predicts arrival time, empty seats, and keeps the traveler updated. A model execution is done utilizing Node MCU with GPS module as vehicle node and bus transport data is sent to the cloud utilizing MQTT²⁹ protocol. To reduce traffic congestion, an convenient and user friendly Android Based Mobile app has been developed to create awareness about real-time information to passengers which compel people to use public transport, thereby reducing traffic that the cities face every day.

Pau (2018) focused on fuzzy-based systems, which helps pedestrian crossing traffic light junctions. The study introduces a fuzzy logic-based system for managing pedestrian crossings efficiently. By employing fuzzy logic algorithms, the research aims to create adaptive and responsive pedestrian crossing systems, enhancing safety and traffic flow at intersections, thereby contributing to the development of intelligent urban transportation networks.

Rahman and Ajala (2018) thrown light on mobility challenges which Nigerian cities are facing like any other county in the world due to exponential population growth pressure and fast urbanization. Smart people and Smart policies are the main factors in the determination of smartness of a city. Author cautioned about overcrowding and

²⁸ Internet of Things

²⁹ Message Queuing Telemetry Transport

infrastructure decadence with uncontrollable increase in the population without substantial development in transportation area . Author brought attention about United Nation statistics. In 2016 United nations assessed that 54.5% of the world's population lived in cities and by 2050 the number will increase to 67%, this perpetually will increment demand for the development of products and administrations.

Sundar (2017) found uncertain arrival times and overcrowding are the major obstacles in using buses as public transport system in India. Author found cameras, infrared devices at doors are not suitable for Indian metros to predict crowd volume. Hence Author proposed a IoT based solution. Author used Android application to simulate the handheld ETM³⁰ used by TNSTC³¹ conductors. On-board passenger online tickets were used to predict crowd volume. Each bus online tickets record was maintained at the server side. The ETMs communicate with the server via an API³². All ETM Machines were fitted with the GPS³³ receiver to track buses in real time to provide real time information to all passengers traveling in the bus, it will also help people to estimate approximate the bus arrival time. The information such as number of people traveling and location of the bus can be displayed in the android application used with Google Maps.

Sunha (2017) revealed the utilization of four variables which reflected within the definition of smart city(portability, open security, wellbeing and efficiency) can be used for citizen time saving. In Top 20 list Singapore is ranked as Number 1 and Bhubaneshwar as Number 19 as smartest city by considering consolidated performance index, while Mobility index kept Singapore as Number 1 and Bhubaneshwar as Number 20. Juniper has found that smart mobility projects have the potential to 'give back' 59.4 hours per year per citizen; these are broken down as: 19.4 hours for Intelligent traffic systems; 31 hours for Open data(City open data can be harnessed to enable public transport information services to develop innovation around that data); 1.2 hours for Cashless payment; 7.8 hours Safer roads.

³⁰ Electronic Ticketing Machines

³¹ Tamil Nadu State Transport Corporation

³² Application Programming Interface

³³ Global Positioning System

Sun and Song (2017) explored secure and trustworthy transportation systems. The research delves into the challenges and solutions in ensuring security and reliability in transportation cyber-physical systems. By addressing cybersecurity issues, the book contributes to the development of secure transportation infrastructures, critical for the future of connected and autonomous vehicles.

Sharma (2018) introduced energy efficient surveillance using Long Range Wide Area Network. This technology uses drones for smart transportation systems. The study explores the integration of drone technology and IoT networks for energy-efficient surveillance in transportation. By employing a low-power, long-range wireless communication technology, the research focuses on creating energy-efficient surveillance solutions, enhancing security and monitoring capabilities in intelligent transportation systems.

Troutbeck (2016) focused on modeling signalized and un-signalized junctions. The chapter provides in-depth insights into the methodologies and techniques employed in modeling both types of intersections. By offering a comprehensive understanding of junction modeling, the research aids transportation professionals and researchers in developing accurate traffic models, essential for urban planning, traffic management, and infrastructure design.

Vakula (2017) proposed that city should have an smart public transport system, Author noticed travelers are facing allot of issues within the smart cities. Travelers are compel to wait for long time since they do not have public transport bus information. He suggested the introduction of technology into the public transport system for the travel comfort of passengers. He also pointed out that if existing bus systems will be empowered with latest technology like providing the arrival departure times of the buses via display boards and Android applications, Global positioning system for bus tracking then passengers will be more keen to travel by public transport systems. The Author proposed Raspberry Pi 3 based GPS system which can send sensor data continuously to a server. High speed Wi-Fi should be provided to all passengers at the Bus terminus to access smart information.

Wen (2008) study introduced solution for road congestion problems. He advised to use a adaptive and automatic traffic light control system to overcome traffic congestion problems. The research presents an intelligent traffic control approach. By integrating expert system methodologies, the study focuses on real-time decision-making to optimize traffic light sequences, aiming to alleviate congestion and enhance urban mobility.

World Health Organization (2018) provided a detailed analysis of road safety worldwide in their 2018 report. The report presents key statistics, trends, and challenges related to road accidents and safety measures. By offering a global perspective on road safety issues, the document serves as a crucial resource for policymakers and organizations working towards improving road safety standards and implementing effective interventions to reduce road accidents and their associated fatalities.

This literature review considered the myriad ways in which advances in technology, urban planning, and customer-focused strategies are helping to improve the public transportation experience. The overall research reviewed highlights the importance of continued research, innovation, and collaboration among stakeholders in building seamless and user-friendly public transportation networks. By harnessing the power of new technology and adopting a human-centered mindset, we can transform the daily commute into a positive and empowering experience for everyone. As we move forward, policymakers, urban planners, technology developers, and the general public must work together to create a future for public transportation that is sustainable, accessible, and improves the overall quality of urban life.

2.3 Commuting and Traffic Congestion Issues

Smart cities aim to harness technology and data to improve citizens' overall quality of life. However, besides many benefits, smart cities can also face challenges related to commuting and traffic congestion. To address these challenges and maximize the benefits of smart cities, it's essential to consider a holistic approach that includes infrastructure development, data privacy safeguards, equitable access, public awareness campaigns, and the active involvement of city residents in the planning and decision-making processes. Smart city initiatives need to be designed with the aim of reducing traffic congestion and improving the overall commuting experience while minimizing

negative side effects. Transport solution theories should have detailed sustainable plan for smart cities. Following are some issues related to travel and traffic in smart cities studied by various scholars in the past.

Albino, Vito, Berardi, and Dangelico (2015) provided comprehensive insights into smart cities, including their definitions, dimensions, performance metrics, and initiatives. The authors explore the multifaceted nature of smart cities, considering technological, economic, social, and environmental dimensions. By analyzing various smart city initiatives, the study contributes to a deeper understanding of the concept, guiding future smart city development strategies and policies.

Ammara, Rasheed, Mansoor, Al-Fuqaha and Qadir (2022) suggested that modern cities are intricate and dynamic systems, characterized by interdependencies and interactions among diverse stakeholders, components, and subsystems. The advent of digital ICT³⁴ has paved the way for the concept of smart cities, aiming to enhance the quality of life for urban dwellers and improve city management. However, the mere deployment of ICT solutions does not guarantee automatic or universal improvements in citizens' well-being. To truly understand the dynamics and outcomes of smart cities, it is essential to analyze them as complex adaptive systems, where multiple interconnected subsystems influence each other. Such an approach enables us to evaluate policy interventions, assess their effectiveness, and anticipate unintended consequences. In this paper, researcher explored the perspective of systems thinking and complex systems in understanding smart cities.

Al-Dweik, Arafat, Radu Muresan, Mayhew, and Lieberman (2017) presented an IoT-based smart transportation Systems. The authors focus on developing a versatile roadside unit capable of integrating various IoT sensors and technologies. The Enhanced Road Side Unit enhances the capabilities of ITS, enabling real-time data collection, analysis, and adaptive traffic management. Their multifunctional approach showcases the flexibility and scalability of IoT solutions in creating intelligent transportation infrastructures.

³⁴ Information and Communications Technology

Bai, Lin, Ma, Wang, and Duan (2020) introduced Pre-Position Congestion Tensor, a cutting-edge traffic congestion prediction model specifically tailored for smart cities. The model utilizes a Relative Position Congestion Tensor, a sophisticated mathematical construct, to forecast traffic congestion accurately. By incorporating complex spatial relationships between vehicles and roadways, Pre-PCT enhances the precision of congestion predictions. This advancement is crucial for urban planners and traffic management authorities, offering them a robust tool to anticipate congestion patterns and optimize traffic flow effectively.

Carmona (2010) delves into the multidimensional aspects of urban design. The text explores the complexities of shaping public spaces within cities, emphasizing the importance of thoughtful urban planning and design principles. By examining the social, cultural, and spatial dimensions of public places, Carmona's work contributes valuable insights to the field of urban design, providing guidance for creating inclusive and vibrant urban environments.

Chakraborty (2019) focused on the development of an Intelligent Traffic Control System specifically designed for smart cities. By harnessing cutting edge technologies, such as artificial intelligence and IoT devices, the system aims to enhance the efficiency of traffic management in urban environments. The study emphasized the significance of integrating intelligent systems into smart city initiatives, showcasing the potential of these systems to enhance traffic flow, minimize congestion, and improve urban mobility.

Dimitrakopoulos, George, and Bravos (2016) explored contemporary technologies in vehicular communication systems. The authors delve into the advancements in communication technologies for vehicles, emphasizing their significance in intelligent transportation systems. By discussing state-of-the-art vehicular communication techniques, the book informs researchers, engineers, and practitioners about the latest developments in the field, fostering innovation in intelligent transportation technologies.

Doolan and Muntean (2016) represented a groundbreaking approach to address environmental concerns related to vehicle emissions. By leveraging Vehicle Ad-Hoc Networks, EcoTrec optimizes traffic flow, subsequently reducing vehicle emissions.

This research not only highlights the potential of Vehicle Ad-Hoc Networks-based solutions but also underscores the importance of intelligent transportation systems in promoting environmental sustainability. EcoTrec serves as a notable example of how technology can be harnessed to reduce the environmental impact of urban transportation.

Deak and Walravens (2019) explored the concept of integrated smart mobility solutions and their implications for the future of urban transportation. The authors examine how emerging technologies and innovative approaches can address the challenges of city mobility and pave the way for more accurate and sustainable transportation systems. In their study, Deak and Walravens analyze the key components and characteristics of integrated smart mobility solutions. They discuss the role of various technological advancements, such as connected vehicles, autonomous driving MaaS³⁵ platforms, and data analytics, in transforming the urban transportation landscape. The authors argue that integrating different modes of transportation and leveraging advanced technologies can lead to more seamless, user-centric, and sustainable mobility experiences. They explore the potential benefits of integrated smart mobility solutions, such as reducing congestion, enhancing accessibility, improving safety, and minimizing environmental impacts. The article also addresses the challenges and barriers that need to be overcome for successful implementation of smart mobility solutions. These include issues related to data privacy and security, infrastructure requirements, policy and regulatory frameworks, and public acceptance. By examining case studies and examples from across the globe/world, Deak and Walravens provide details of the practical implications and potential outcomes of integrated smart mobility solutions. They emphasize the importance of collaboration among stakeholders, including governments, transportation providers, technology companies, and citizens, to realize the vision of future urban mobility.

Erlmann and Dantzig (2018) conducted a literature review to explore the potential of shared mobility services in alleviating urban traffic congestion. The study focuses to provide a detailed understanding of the existing research and insights on the effect of shared mobility on traffic congestion in urban areas. The review highlights that shared

³⁵ Mobility-as-a-Service

mobility services, such as ride-sharing, car-sharing, bike-sharing, and on-demand transportation, have gained significant attention as potential solutions to mitigate traffic congestion. These services have the potential to reduce the number of private vehicles on the road, promote more efficient use of transportation resources, and provide alternative transportation options to users. The findings of the review suggest that shared mobility services can indeed have a positive impact on traffic congestion. Studies indicate that these services have the potential to reduce VKT³⁶ and vehicle ownership, leading to a decrease in overall traffic volume and congestion levels. Shared mobility can also encourage the use of more sustainable transportation modes, such as public transport system and non-motorized vehicles.

Fintikakis and Bourka (2018) examine commuting patterns in the context of smart cities by conducting a comparative analysis of Stockholm and London. The researchers aim to understand the impact of smart city initiatives on commuting behavior and explore the differences between these two major European cities. The study utilizes data from various sources, including surveys and official statistics, to gather information on commuting patterns, transportation modes, and the use of smart technologies in commuting. The findings reveal several interesting insights. Both Stockholm and London have implemented smart city solutions to improve urban mobility and transportation efficiency. However, the two cities exhibit distinct commuting patterns and modal split. Stockholm demonstrates a higher reliance on sustainable modes of transportation, such as public transit, cycling, and walking, while London has a greater dependence on private vehicles. The researchers attribute these differences to various factors, including cultural preferences, urban form, and the availability and quality of transportation infrastructure. The study also highlights the role of smart technologies in shaping commuting behavior. Smart city initiatives, such as real-time traffic information systems, mobile applications, and integrated ticketing systems, contribute to more informed and efficient commuting experiences. These technologies enable travelers to make informed decisions, optimize their routes, and choose sustainable transportation options. However, the study emphasizes that the successful implementation of smart city solutions requires a comprehensive and

³⁶ Vehicle Kilometers Traveled

integrated approach that considers the needs and preferences of commuters. Overall, this comparative study sheds light on the relationship between commuting patterns and smart city initiatives in Stockholm and London.

Goggin and Gerard (2012) have done research on the intersection of mobile internet, cars, and social dynamics. The study investigates the social implications of mobile internet usage in vehicles, emphasizing the transformative role of technology in reshaping communication patterns and social interactions within the context of driving. By examining the relationship between mobile internet, cars, and society, Goggin's work sheds light on the evolving dynamics of human-technology interactions in the mobile age, offering valuable insights into the societal impact of connected mobility.

Gkiotsalitis and Cats (2018) investigate the potential implications of AVs³⁷ on urban environments. The study aims to assess the impact of AVs on various aspects, including traffic congestion, travel behavior, and urban form. The researchers conduct a detailed review and analysis of existing materials and models to evaluate the effects of AVs on cities. The findings of the study suggest that the introduction of AVs can have both positive and negative consequences for urban areas. On the positive side, AVs have the potential to enhance road safety, reduce traffic congestion, and improve the overall efficiency of transportation systems. They may also lead to changes in urban form, as the need for parking spaces decreases, potentially allowing for the repurposing of land for other uses. Furthermore, AVs could promote shared mobility services, potentially decreasing the reliance on private car ownership and promoting more sustainable transportation options. However, the study also highlights several challenges and concerns associated with the widespread adoption of AVs. One concern is the potential for increased VMT³⁸ as AVs may lead to greater convenience and reduced travel costs, potentially offsetting the anticipated reduction in congestion. The researchers also raise issues related to data privacy, cybersecurity, legal and regulatory frameworks, and public acceptance, which need to be addressed to ensure the successful integration of AVs into cities. The study emphasizes the need for further research and policy development to proactively address the potential impacts of AVs on cities. It

³⁷ Autonomous Vehicles

³⁸ Vehicle miles travelled

underscores the importance of considering various factors, such as urban planning, transportation infrastructure, and social and environmental impacts, to maximize the benefits and mitigate any negative consequences associated with AV deployment. By doing so, cities can better prepare for the transformative effects of AVs and create more sustainable and livable urban environments.

Ghasem-Aghaee (2019) explored the synergy between simulation, intelligence, and agent-based systems. The study delves into the integration of these technologies, offering insights into their collaborative potential. By examining current and future developments in artificial intelligence, the research contributes to the exploration of innovative approaches for creating intelligent systems. The study's findings provide a foundation for advancing AI-driven solutions in simulation and agent-based modeling, shaping the future of intelligent transportation systems.

Han, Liang and Zhang (2015) discussed the convergence of mobile cloud sensing, big data, and 5G networks in shaping intelligent and smart cities. They emphasize the transformative impact of these technologies, envisioning a future where interconnected data-driven systems enhance urban living. Through mobile cloud sensing, real-time data collection becomes feasible, and when combined with big data analytics and high-speed 5G networks, it enables intelligent decision-making processes in various aspects of urban life, paving the way for efficient, sustainable, and smart urban environments.

Hall and Pfeiffer (2016) have shown the influence of smart cities on travel behavior. The study takes an exploratory approach to understand how the implementation of smart city technologies affects people's transportation choices and patterns. In their research, the authors analyze the potential impacts of smart city initiatives, such as intelligent transportation systems, real-time data analysis, and urban planning strategies, on travel behavior. They examine how these technologies can enhance the efficiency and convenience of transportation options, promote sustainable modes of travel, and reduce congestion on urban road networks. The study employs both qualitative and quantitative research methods, including interviews, surveys, and data analysis, to gather insights into the travel behavior changes associated with smart city interventions. The authors investigate factors such as travel time, mode choice, trip frequency, and the adoption of new mobility services. The findings of the study shed light on the positive effects of smart city solutions on travel behavior. The

implementation of intelligent transportation systems and real-time data analysis were found to improve travel efficiency, reduce travel time, and enhance the reliability of transportation services. Moreover, the study suggests that smart city technologies have the potential to promote sustainable travel choices, such as walking, cycling, and public transportation, by providing real-time information and facilitating multi-modal integration. The research contributes to the understanding of how smart city initiatives can shape travel behavior and promote sustainable urban mobility. It highlights the importance of technological interventions in improving transportation systems and provides valuable insights for policymakers and urban planners in designing smart city strategies that prioritize efficient and sustainable travel.

Jung and Couclelis (2017) examined the intersection of smart cities and the politics surrounding urban data. The authors investigate how the collection, analysis, and use of data in smart cities can shape power dynamics, governance, and decision-making processes. The study explores the concept of smart cities, which leverage data and technology to improve urban services and enhance the quality of life for residents. It highlights the importance of data in smart city initiatives and how it becomes a valuable asset for urban planning, policy-making, and resource allocation. Jung and Couclelis argue that the politics of urban data in smart cities are characterized by issues of control, ownership, access, and privacy. They analyze the power dynamics between different stakeholders, including government agencies, private companies, citizens, and advocacy groups, in the collection, sharing, and utilization of urban data. The article examines the implications of data-driven decision-making in smart cities and raises concerns about potential biases, exclusion, and surveillance. It discusses the challenges of data governance, emphasizing the need for transparency, accountability, and citizen engagement in the management of urban data. Furthermore, the study highlights the role of data standards, interoperability, and data sharing protocols in shaping the politics of urban data. It explores the potential benefits of open data initiatives and the importance of ethical considerations in the use of personal data within smart city frameworks. Overall, the research sheds light on the political dimensions of urban data in smart cities. The article contributes to the understanding of the complexities surrounding data-driven urban governance, highlighting the need for inclusive and responsible approaches to the use of data in shaping the future of cities.

Jin, Ma, and Kosonen (2017) introduced an intelligent control system for traffic lights, incorporating simulation-based evaluations. By utilizing simulation techniques, the system optimizes traffic signal timings, aiming to enhance traffic flow and minimize congestion. This approach emphasizes the importance of simulation-based assessments in evaluating the effectiveness of intelligent traffic control systems. Through rigorous evaluation, the study contributes valuable insights into the performance and efficiency of intelligent traffic management solutions, guiding their implementation in real-world scenarios.

Janahan (2018) presented IoT based Smart Traffic Signal Monitoring System. By leveraging IoT technology, the system captures real-time vehicle count data, providing accurate and timely information for traffic analysis. The research underscores the importance of data-driven decision-making in traffic management. The system's ability to provide precise and up-to-date traffic information is instrumental in optimizing traffic signal timings, reducing congestion, and improving overall urban mobility.

Joo, Ahmed and Lim (2020) proposed a traffic signal control system for smart cities using reinforcement learning techniques. By employing machine learning algorithms, specifically reinforcement learning, the system dynamically adjusts traffic signals based on real-time traffic conditions. The research highlights the adaptive nature of the system, allowing it to respond dynamically to changing traffic patterns. This approach represents a significant advancement in traffic management, providing agile and responsive solutions to address congestion and optimize traffic flow within smart cities.

Kitchin (2014) explored the paradigm shifts induced by big data and new epistemologies. He critically examines the impact of big data on knowledge production and challenges traditional research methods. Kitchin's work highlights the transformative potential of big data, emphasizing the need for innovative approaches to understanding societal phenomena in the age of data-driven insights.

Kumar, Rahman, and Dhakad (2020) presented a traffic light control system that combines fuzzy inference and deep reinforcement learning techniques. By integrating fuzzy logic and deep learning, the system optimizes traffic signal timings with a high degree of precision. This innovative approach showcases the synergy between fuzzy inference and deep learning, offering a robust solution for intelligent transportation

systems. The system's ability to make informed decisions based on complex data sets contributes significantly to enhancing traffic flow and reducing congestion in urban environments.

Lakshmi (2016) explored big data analytics in the service industry. The study investigates the applications of big data analytics within the service sector, highlighting its potential to enhance operational efficiency, customer satisfaction, and strategic decision-making. By synthesizing existing knowledge, the literature survey provides a comprehensive overview of the impact of big data analytics on the service industry, offering valuable insights for businesses and researchers alike.

Lingani, Rawat and Garuba (2019) introduced a Smart Traffic Management System that utilizes advanced deep learning techniques for efficient traffic management within smart cities. By employing deep learning algorithms, the system can analyze vast amounts of traffic data, enabling real-time decision-making. This approach showcases the transformative power of artificial intelligence in revolutionizing traffic management strategies. By providing intelligent and adaptive solutions, the system contributes significantly to enhancing urban mobility, reducing congestion, and improving the overall quality of life in smart cities.

Lee and Chiu (2020) introduced a Smart Traffic Signal Control System tailored specifically for smart city applications. By integrating advanced technologies and smart algorithms, the system optimizes traffic signal timings, effectively managing traffic flow and reducing congestion. The research emphasizes the importance of tailored solutions, recognizing the unique challenges faced by smart cities. By addressing these challenges directly, the system offers a customized approach to traffic management, aligning with the specific needs and dynamics of smart urban environments.

Malek, Li, Yang, Hasan and Zhang (2012) discussed the improvement of energy efficiency in Ad Hoc On-Demand Distance Vector routing protocols. The authors propose enhancements to the Ad Hoc On-Demand Distance Vector routing protocol, aiming to optimize energy consumption in mobile ad hoc networks. Their work is instrumental in advancing the field of wireless communication, paving the way for more energy-efficient protocols in various applications, including smart city networks.

Misuraca, Gianluca, Francesco, Mureddu, and Osimo (2014) presented in the book "Policy-making 2.0: Unleashing the Power of Big Data for Public Governance," explores the integration of big data into public governance. The authors discuss the utilization of big data analytics for policy-making processes, emphasizing the potential for data-driven decision-making in the public sector. By harnessing the power of big data, governments can enhance their policy formulation, implementation, and evaluation strategies, fostering more efficient and effective governance practices.

Mondal and Rehena (2019) focused on developing an intelligent traffic congestion classification system utilizing ANNs³⁹. By employing ANNs, the system accurately classifies traffic congestion levels, providing valuable insights into traffic conditions. The study highlights the potential of artificial intelligence techniques, particularly ANNs, in developing precise and reliable traffic monitoring systems for smart cities. By accurately classifying congestion levels, the system contributes to data-driven decision-making, enabling proactive measures to alleviate congestion and enhance urban mobility.

Nasr, Elie, Kfoury and Khoury (2016) presented an IoT-based approach to vehicle accident detection, reporting, and navigation. The authors explore the utilization of IoT devices and sensors to detect vehicle accidents in real-time. By enabling swift accident reporting and navigation assistance, their system enhances road safety and emergency response mechanisms. The study illustrates the practical application of IoT technology in enhancing transportation safety and underscores its potential in shaping the future of intelligent transportation systems.

Psomakelis and Evangelos (2016) have shown the integration of big data and social networking data for smart cities. The authors explore the synergy between big data analytics and social networking data in shaping intelligent urban environments. By harnessing social network data, cities can gain valuable insights into public behavior and preferences, facilitating more informed decision-making processes. The study highlights the potential of combining big data and social networking data for enhancing smart city initiatives and improving urban quality of life.

³⁹ Artificial Neural Networks

Rizwan, Patan, Suresh, and Rajasekhara (2016) introduced a real-time smart traffic management system for smart cities, leveraging the IoT and big data technologies. The authors propose an intelligent system capable of real-time data analysis and traffic management. By integrating IoT devices and big data analytics, the system optimizes traffic flow, reduces congestion, and enhances overall urban mobility. Their innovative approach demonstrates the potential of IoT and big data in revolutionizing urban transportation systems.

Silva and Silva (2019) conducted a systematic review exploring the relationship between congestion charging and smart cities. The study aims to provide a comprehensive understanding of the impact of congestion charging policies in the context of smart city initiatives. By examining a range of relevant literature, the authors analyze the key findings and trends associated with the integration of congestion charging schemes and smart city technologies. The review highlights that congestion charging policies have been implemented in various cities worldwide as a means to alleviate traffic congestion and promote sustainable urban mobility. The integration of smart city technologies, such as advanced sensors, data analytics, and intelligent transportation systems, has played a crucial role in the successful implementation and management of congestion charging schemes. The findings indicate that smart city initiatives have enhanced the effectiveness and efficiency of congestion charging policies. By utilizing real-time data and predictive analytics, cities can optimize pricing structures, dynamically adjust congestion charges, and provide personalized travel information to users. The integration of smart technologies has also facilitated the seamless collection and processing of toll payments, improving the overall user experience and administrative processes. Furthermore, the systematic review identifies various benefits associated with the integration of congestion charging and smart city concepts. These include the reduction of traffic congestion, improvement of air quality, promotion of sustainable transportation modes, and generation of revenue for infrastructure investments. The review also discusses the potential challenges and barriers to the implementation of congestion charging schemes in smart cities, such as public acceptance, equity considerations, and technological requirements. The systematic review emphasizes the significant role of smart city technologies in enhancing the effectiveness of congestion charging policies. The findings provide valuable insights for policymakers and urban planners in understanding the synergies

between congestion charging and smart city initiatives. The study suggests that the integration of these approaches can contribute to more sustainable and efficient urban transportation systems.

Sochor and Dvorak (2020) provides a comprehensive overview of the effects of smart city solutions on urban mobility. The researchers conducted a systematic review of existing literature to analyze the impacts of various smart city technologies and initiatives on transportation systems and commuting patterns. The review highlights several key findings. Firstly, it emphasizes the positive effects of smart city solutions on improving transportation efficiency and reducing traffic congestion. Smart technologies such as intelligent transportation systems, real-time traffic management, and advanced data analytics have shown promise in optimizing traffic flow, enhancing public transportation services, and facilitating multi-modal integration. The study also recognizes the potential of smart mobility services, including ridesharing, carpooling, and bike-sharing, to encourage sustainable transportation choices and decrease private vehicle ownership. These services, often facilitated by mobile applications and digital platforms, have the potential to enhance connectivity and provide flexible alternatives to traditional commuting patterns. Furthermore, the review identifies the importance of data-driven decision-making and urban planning in achieving effective smart mobility solutions. The analysis highlights the role of big data, Internet of Things devices, and sensor networks in gathering real-time information about transportation patterns, which can inform policy interventions and enable responsive transportation management. However, the study also acknowledges the challenges associated with implementing smart city solutions. These include issues of data privacy and security, interoperability of different systems.

Zafar, Haq, Sohail, Chughtai and Muneeb (2022) revealed that in the future, as smart cities continue to evolve, the abundance of data from diverse sources and different levels of difficulty will be initiated, fused, treated, and employed. Within the realm of city traffic planning in smart cities, one of the most pressing challenges revolves around predicting and mitigating traffic congestion. Traffic congestion is a multifaceted phenomenon influenced by numerous factors. In addition to vehicular mobility, elements such as road network characteristics, weather conditions, holidays, and peak

hours significantly contribute to congestion, particularly on arterial roads within cities. This paper introduces a novel approach to addressing this challenge by proposing

a hybrid deep learning model based on the combination of GRU⁴⁰ and LSTM⁴¹ architectures. The model is applied to city-wide traffic data, which is aggregated from various heterogeneous sources. To support the implementation of the proposed model, a customized data pipeline has been developed. This pipeline incorporates a series of algorithms designed to handle tasks such as map matching, handling data sparsity, removing outliers, adjusting zero speeds, and mapping road segments using OSM⁴². Rigorous experimentation has been conducted to showcase the enhanced performance of the proposed method. Comparative analysis demonstrates that our methodology achieves an impressive 95% accuracy, outperforming other deep neural network models commonly used in this domain.

In summary, this literature review considered the dynamic interface of technology and transportation systems in the context of smart cities. The transformative potential of technological advances such as the IoT, artificial intelligence, and data analytics is being demonstrated in designing more efficient, sustainable, and connected urban mobility solutions. As smart cities seek to address challenges such as rapid urbanization, environmental issues, and increased demand for seamless transportation, the integration of innovative technologies is proving to be a key driver of change.

2.4 Technologies and Transportation Systems

Smart cities use a variety of technologies and transportation systems to improve mobility, reduce traffic congestion, and improve citizens' overall quality of life. By integrating different transportation technologies and systems such as smart transportation systems, mobility as a service, electric vehicles, smart parking solutions, fleet management systems, data analytics and artificial intelligence, smart cities aim to create a more efficient, resilient, accessible urban transportation network while improving the overall quality of life of residents.

⁴⁰ Gated Recurrent Unit

⁴¹ Long Short-Term Memory

⁴² Open Street Map

Cunha, Amaral, Silva and Pinheiro (2021) presented a comprehensive review of the existing literature on smart public transportation in smart cities. The authors recognize the importance of efficient and sustainable public transportation systems in the context of smart cities, where technological advancements and data-driven solutions are utilized to improve urban mobility. The study employs a systematic literature review methodology to analyze and synthesize relevant research articles from various sources. The authors identify key themes and trends in the literature, including the use of emerging technologies, data analytics, intelligent transportation systems, and innovative mobility services to enhance public transportation in smart cities. The findings of the literature review highlight several aspects of smart public transportation. Firstly, the integration of smart technologies, such as Internet of Things devices, sensors, and real-time data collection systems, enables the provision of real-time information to passengers, optimized route planning, and improved operational efficiency. Secondly, the utilization of data analytics and machine learning algorithms enables the prediction of demand patterns, congestion management, and personalized services for passengers. The review also emphasizes the importance of sustainable and multimodal transportation solutions, including the integration of public transit with other modes of transport, such as cycling and car-sharing. Furthermore, the authors discuss the role of policy and governance frameworks in promoting and supporting the development of smart public transportation systems. The study concludes by highlighting the need for further research and implementation of smart public transportation initiatives in real-world settings. The findings provide insights for policymakers, urban planners, and transportation authorities to develop strategies and interventions that enhance the quality, efficiency, and sustainability of public transportation in smart cities. Overall, the research work contributes to the understanding of smart public transportation in the context of smart cities and provides a valuable resource for researchers and practitioners interested in this field.

Gohar, Sagheer, Javaid, Anpalagan, and Khan (2019) provided an in-depth analysis of fog computing in the context of ITS⁴³. The authors explore the architecture, opportunities, and challenges associated with leveraging fog computing for transportation management in smart cities. The paper begins by highlighting the

⁴³ Intelligent Transportation Systems

limitations of traditional centralized cloud computing in handling the massive data generated by ITS. It then introduces the concept of fog computing as a decentralized paradigm that brings computation, storage, and networking closer to the edge of the network. The authors discuss the architecture of a fog-enabled ITS, which involves the integration of fog nodes, smart vehicles, road-side units, and cloud infrastructure. They emphasize the advantages of fog computing in enabling real-time data processing, reducing latency, enhancing privacy and security, and enabling context-aware decision making. The opportunities presented by fog computing in the context of transportation management are explored in detail. These include improved traffic management and congestion control, efficient routing and navigation, enhanced driver safety and assistance, and support for emerging applications such as autonomous vehicles and smart parking. The paper also addresses the challenges associated with fog computing in ITS, such as resource constraints, network scalability, interoperability, data management, and security issues. The authors discussed potential solutions and research directions to overcome these challenges. Overall, the paper provides a comprehensive overview of the architecture, opportunities, and challenges of fog computing in the context of intelligent transportation systems. It highlights the potential benefits of leveraging fog computing in smart cities to enable efficient and reliable transportation management. The insights provided in this paper can serve as a valuable resource for researchers, practitioners, and policymakers working in the field of smart transportation and urban planning.

Kaur and Kaur (2017) provided an in-depth analysis of the applications and advancements of the Internet of Things in the transportation sector. The authors aim to present a comprehensive understanding of the potential benefits, challenges, and future directions of IoT implementation in transportation systems. The article begins with an introduction to IoT and its relevance to the transportation domain. It highlights the key characteristics of IoT, such as connectivity, sensing capabilities, and data processing, that make it well-suited for improving transportation systems. The authors emphasize the importance of IoT in enabling the collection and analysis of real-time data, which can be utilized for enhancing various aspects of transportation, including traffic management, vehicle safety, and passenger experience. The review then discusses the applications of IoT in transportation, covering areas such as smart traffic management, intelligent vehicle systems, fleet management, and passenger information systems. The

authors provide detailed insights into how IoT technologies can be leveraged to improve efficiency, reduce congestion, enhance safety, and enable predictive maintenance in transportation systems. Additionally, the article examines the challenges and issues associated with the implementation of IoT in transportation. These challenges include data security and privacy concerns, interoperability of devices and systems, scalability, and the need for reliable connectivity. The authors discuss potential solutions and strategies to overcome these challenges, such as standardization efforts and the development of robust communication protocols. Furthermore, the authors discuss emerging trends and future directions in IoT for transportation. They highlight the integration of IoT with other technologies, such as artificial intelligence and big data analytics, to enable advanced transportation management and decision-making. In summary, Kaur and Kaur provide a comprehensive review of IoT for transportation, encompassing its applications, challenges, and future prospects. The article serves as a valuable resource for researchers, practitioners, and policymakers in understanding the potential of IoT in transforming transportation systems and addressing the associated challenges.

Khan, Salah and Zeadally (2019) explored the applications, opportunities, and challenges of fog computing in the context of intelligent transportation systems. The authors delve into the potential of fog computing to address the limitations of traditional centralized cloud computing in handling the massive data generated by ITS. The paper begins by providing an overview of ITS and the increasing demand for real-time data processing and decision-making capabilities. It introduces the concept of fog computing as a decentralized computing paradigm that extends cloud services to the edge of the network. The authors highlight the advantages of fog computing in terms of low latency, location awareness, mobility support, and efficient resource utilization. The applications of fog computing in ITS are discussed, including traffic management, vehicle-to-vehicle and vehicle-to-infrastructure communication, intelligent routing, and autonomous vehicles. The authors emphasize the potential benefits of fog computing in enabling real-time data analytics, adaptive traffic control, enhanced driver safety, and improved overall transportation efficiency. The challenges associated with fog computing in ITS are also addressed in the paper. These challenges include resource management, security and privacy concerns, network connectivity, interoperability, and scalability. The authors discuss various approaches and solutions to overcome these

challenges, including workload distribution, data caching, security mechanisms, and standardization efforts. In all the research work provides a comprehensive analysis of the opportunities and challenges of fog computing in the domain of intelligent transportation systems. It highlights the potential of fog computing to revolutionize transportation management by enabling real-time data processing and decision-making capabilities at the network edge. The insights presented in this paper can serve as a valuable resource for researchers, practitioners, and policymakers working on the integration of fog computing in intelligent transportation systems.

Pereira, Rodrigues and Saleem (2018) conducted a comprehensive survey on IoT-based transportation systems. The aim of the research is to provide an overview of the applications, challenges, and opportunities associated with IoT in the transportation domain. The authors begin by introducing the concept of IoT and its potential impact on transportation systems. They discuss how IoT technologies can be integrated into various components of transportation systems, including vehicles, infrastructure, and users. The study then presents a survey of existing literature, focusing on different applications of IoT in transportation, such as smart parking, traffic monitoring, fleet management, and road safety. Authors also highlight the key challenges and issues that arise in the implementation of IoT-based transportation systems, including data security, privacy concerns, interoperability, and scalability. They discuss the importance of addressing these challenges to ensure the successful deployment and operation of IoT solutions in transportation. Furthermore, the study identifies several opportunities and benefits that IoT brings to transportation systems, such as improved traffic management, reduced congestion, enhanced safety, and increased efficiency. The authors emphasize the potential of IoT to revolutionize the way transportation systems are designed, operated, and experienced. In conclusion, the paper provides a comprehensive survey of IoT-based transportation systems, covering applications, challenges, and opportunities. It serves as a valuable resource for researchers, practitioners, and policymakers interested in understanding the potential of IoT in transforming transportation systems and addressing the associated challenges.

Vlahogianni, Karlaftis and Golias (2014) in their comprehensive review, provided an overview of the state of the art in short-term traffic forecasting and shed light on future directions in this field. They recognize the importance of accurate traffic forecasting for

efficient transportation management and discuss the advancements and challenges in this area. The article begins by highlighting the significance of short-term traffic forecasting in various transportation applications, such as traffic control, congestion management, and travel time estimation. The authors emphasize the need for accurate and reliable forecasting models to enable efficient traffic management and improve overall transportation system performance. The article also addresses the challenges and limitations of existing traffic forecasting models, including data availability, model complexity, and uncertainty. The authors then review the different approaches and methodologies employed in short-term traffic forecasting. They discuss the traditional time series analysis methods, such as ARIMA⁴⁴, as well as more advanced techniques like ANN, SVM⁴⁵, and hybrid models that combine multiple methods. The authors highlight the need for real-time data, such as traffic sensor data and weather information, to improve the accuracy of forecasting models. They also discuss the importance of incorporating non-linear relationships and dynamic traffic patterns into the models. Furthermore, the authors explore emerging trends and future directions in short-term traffic forecasting. They discuss the potential of big data analytics, machine learning, and data fusion techniques in improving the accuracy and reliability of forecasting models. They also highlight the importance of considering the impact of emerging technologies, such as connected and autonomous vehicles, on traffic forecasting. In conclusion, Vlahogianni, Karlaftis, and Golias provide a comprehensive review of short-term traffic forecasting, discussing the existing models, challenges, and future directions. The article serves as a valuable resource for researchers, practitioners, and policymakers involved in transportation planning and management, highlighting the importance of accurate traffic forecasting for the development of efficient and sustainable transportation systems.

Zanella, Bui, Castellani, Vangelista and Zorzi (2014) provided a comprehensive overview of the Internet of Things and its role in the development of smart cities. The authors present the concept of the IoT as a paradigm where physical objects are connected to the internet and can interact with each other and with the environment to

⁴⁴ Autoregressive Integrated Moving Average

⁴⁵ Support Vector Machines

enable a range of applications and services in urban environments. The article begins by introducing the concept of smart cities and the key challenges faced by urban areas, including sustainability, energy efficiency, transportation, and public safety. The authors argue that the IoT can play a vital role in addressing these challenges by enabling the deployment of interconnected devices and systems for monitoring, control, and optimization of various urban services. The authors then delve into the technical aspects of the IoT, discussing the key enabling technologies such as wireless sensor networks, radio frequency identification, and cloud computing. They highlight the importance of communication protocols, data management, and security in ensuring the effective operation of IoT systems in smart cities. Furthermore, the article presents a comprehensive classification of IoT applications in smart cities, covering areas such as transportation, energy management, healthcare, environmental monitoring, and public safety. The authors provide detailed insights into the potential benefits and challenges associated with each application domain, highlighting the need for interoperability, scalability, and privacy protection. The article also discusses several case studies and initiatives where IoT technologies have been applied in real-world smart city projects. These examples illustrate the potential of the IoT to transform urban environments and improve the quality of life for citizens. In conclusion, Zanella et al. provide a comprehensive overview of the Internet of Things for smart cities, covering its concepts, technologies, applications, and challenges. The article serves as a foundational reference for researchers, practitioners, and policymakers interested in understanding the potential of the IoT in shaping the future of urban environments and addressing the complex challenges faced by cities.

Literature review on Technology and transportation system, reminded me the profound impact, technology has on transforming the transport environment. By exploring innovative solutions, data-driven insights, and integrating cutting-edge technologies, it is clear that our transportation system is on the verge of transformation. As we navigate the intersection of technology and transportation, we sincerely hope that this research will contribute to the ongoing debate and stimulate further research and advancement in this field. The potential to improve the efficiency, sustainability and safety of our transport networks is huge, and the path to a smarter, more connected future is both exciting and challenging.

2.5 IoT Based Traffic Prediction Models

IoT-based traffic prediction models are an integral part of smart cities as they can help manage traffic effectively, reduce traffic congestion, and improve overall mobility. IoT-based traffic prediction models for smart cities can significantly improve the efficiency of transportation systems, reduce traffic congestion, and improve citizens' overall quality of life. These models must be flexible and adaptable to accommodate changing traffic patterns and urban infrastructure. The steps involved in developing an IoT-based traffic forecasting model for a smart city are data collection, data processing, data analysis, predictive modeling, visualization and reporting, feedback loops, communication and evaluation. Review of work already done in this field is as follows.

Abbas (2011) researched article explores bio-inspired neuro-fuzzy-based dynamic route selection to avoid traffic congestion. Published in the International Journal of Scientific and Engineering Research, the study presents an innovative approach to optimizing traffic routes. By leveraging bio-inspired computational models and fuzzy logic, the research proposes adaptive routing strategies. These strategies enable vehicles to dynamically select routes, avoiding congested areas and optimizing travel times. The study contributes to the development of intelligent traffic management systems, promoting efficient and congestion-free urban mobility.

Amadeo (2016) the authors discuss information-centric networking for the IoT. The study explores the challenges and opportunities presented by IoT devices and their information-centric networking. By addressing the unique requirements of IoT communication, the research contributes valuable insights into optimizing data transmission and management, enhancing the efficiency of IoT-based applications, and fostering the development of smarter and more connected cities.

Bhardwaj, Malik, Chauhan, and Rana (2021) conducted a comprehensive survey on IoT-based traffic prediction models for smart cities. The study was published in the Journal of Ambient Intelligence and Humanized Computing. The authors aimed to provide an extensive overview of IoT-based traffic prediction models specifically designed for smart cities. They reviewed various research articles and studies in this domain to identify the current state-of-the-art approaches and advancements. The survey focused on the use of Internet of Things technologies for traffic prediction in

smart cities. It explored different models and techniques employed for this purpose. The authors discussed the application of machine learning and deep learning algorithms in traffic prediction, emphasizing the integration of IoT devices and sensors to gather real-time traffic data. Furthermore, the authors highlighted the challenges and limitations faced by existing IoT-based traffic prediction models. They identified areas of improvement and potential future research directions, such as incorporating additional data sources, considering heterogeneous traffic scenarios, and enhancing prediction accuracy. Overall, the survey provides a comprehensive analysis of IoT-based traffic prediction models in the context of smart cities.

Balasubramanian (2023) explained in the context of smart cities, the rapid increase in the number of vehicles has led to congestion, pollution, and disruptions in the transportation of goods. Additionally, road accidents continue to cause numerous fatalities and permanent injuries each year. To address these challenges, the implementation of an IoT-based TMS⁴⁶ has gained prominence. This system utilizes autonomous vehicles and intelligent devices equipped with sensors to collect, transmit, and analyze traffic data. Machine learning techniques are also employed to improve the efficiency of the transportation system. This research work focuses on the development of an ATM⁴⁷ system integrated with an AALS⁴⁸ to effectively manage traffic congestion and detect accidents. To ensure secure transmission of traffic-related data, the SEE-TREND⁴⁹ mechanism is utilized. The proposed design incorporates various scenarios to address potential issues in the transportation system. The ATM model continuously adjusts the timing of traffic signals based on the traffic volume and predicted movements from nearby junctions. By allowing vehicles to pass through green lights in a progressive manner, the system significantly reduces travel time and alleviates traffic congestion. The experimental results demonstrate that the proposed ATM system outperforms traditional traffic management methods and holds promise for enhancing transportation planning in smart city environments. Furthermore, the integration of ATM with SEE-TREND facilitates secure transmission of traffic data, resulting in

⁴⁶ Traffic Management System

⁴⁷ Adaptive Traffic Management

⁴⁸ Accident Alert Sound System

⁴⁹ Secure Early Traffic-Related Event Detection

reduced traffic congestion, minimized waiting times for vehicles, decreased accident rates, and an overall enhanced travel experience. In summary, the IoT-based Traffic Management System presented in this study offers an innovative solution to address traffic congestion and improve road safety in smart cities. The combination of adaptive traffic management, accident detection, and secure data transmission contributes to a more efficient and sustainable transportation system for the benefit of both commuters and city governance.

Coutard and Olivier(2014) collaborated research effort delves into urban megatrends and outlines a European research agenda. Published in a comprehensive report, the study explores the transformative trends shaping urban environments. By identifying key megatrends, such as demographic shifts and technological advancements, the research provides a strategic framework for future urban development initiatives. The report offers essential guidelines for policymakers, researchers, and urban planners to address the challenges and opportunities posed by urbanization and emerging technologies, guiding the creation of sustainable and resilient smart cities.

Chong, Hon Fong, and Kiat Ng (2016) presented at the IEEE Student Conference focuses on the development of IoT devices for traffic management systems. By leveraging IoT technology, the study explores innovative approaches to enhance traffic management efficiency. The research contributes to the growing field of intelligent transportation systems, emphasizing the integration of IoT devices to optimize traffic flow, reduce congestion, and improve overall urban mobility.

Chowdhury (2016) research focused on a priority-based and secured traffic management system for emergency vehicles using IoT. The study addresses the critical need for efficient traffic management to facilitate emergency vehicle movement. The research proposes solutions to ensure the swift and secure passage of emergency vehicles, improving response times and enhancing overall urban safety and security.

Mamoona and Humayun (2022) proposed an IoT-based architecture for smart traffic management in metropolitan areas, with a specific focus on Riyadh, Saudi Arabia. The aim is to address the problem of excessive traffic congestion during peak hours by utilizing modern technologies such as IoT, cloud computing, 5G, and big data. The proposed architecture involves the deployment of IoT devices and agents that collect

and count vehicles, with real-time data being stored in the cloud through message agents. The system utilizes messaging agents as actuators to broadcast real-time traffic information on dashboards and Google Maps, helping citizens make informed decisions and save time on the roads. Wi-Fi-enabled controllers facilitate timely message transmission. A case study is conducted to validate the accuracy and effectiveness of the proposed architecture. The proposed solution has the potential to significantly reduce traffic congestion and improve mobility in metropolitan areas. It leverages IoT technologies to provide real-time traffic updates and early warning messages to drivers, enabling them to choose the most efficient routes and avoid unexpected traffic incidents. The study also highlights future directions for the system, including optimizing route recommendations based on real-time data, enhancing traffic signal control in different environments, and addressing IoT security considerations. Overall, the proposed IoT-based architecture offers a promising approach to tackle traffic congestion in cities like Riyadh. By integrating advanced technologies, the system aims to provide efficient traffic management, enhance communication between drivers and the transportation network, and improve the overall travel experience in urban areas.

Neelakandan (2016) as author focused on large-scale optimization to minimize network traffic in big data applications using MapReduce. The study delves into the realm of computational power, energy information, and communication. By employing the MapReduce framework, the research aims to enhance the efficiency of processing vast volumes of data. This approach holds significance in the context of big data applications where optimizing network traffic is crucial for seamless data processing and communication. The paper discusses the methodologies and techniques applied to achieve these objectives, contributing valuable insights to the domain of big data analytics.

Neelakandan and Paulraj (2020) presented an innovative Automated Exploring and Learning Model for data prediction utilizing a Balanced Cellular Automata-Support Vector Machine approach. The study focuses on developing a robust model capable of predictive data analysis. By integrating Cellular Automata-Support Vector Machine, a hybrid approach leveraging cellular automata and machine learning, the authors propose an automated system for exploring data patterns and making predictions. The study emphasizes the importance of balancing the learning process to enhance the

accuracy and reliability of predictions. This model showcases advancements in automated predictive analytics, offering valuable applications in various fields.

Ou, Haoyuan, Zhang, and Wang (2016) presented the development of an intelligent traffic control system based on the Internet of Things and FPGA⁵⁰ technology in Proteus. Published in the Traffic journal, the research explores the integration of IoT and FPGA technology for traffic control applications. By leveraging these technologies, the system enhances traffic management capabilities, offering efficient solutions to address urban traffic challenges. The study showcases the potential of cutting-edge technologies in shaping intelligent transportation systems.

Pattanaik, Singh, Gupta and Singh (2016) introduced a Smart Real-time Traffic Congestion Estimation and Clustering Technique for urban vehicular roads. The research addresses the challenge of real-time traffic management in urban areas by proposing an efficient congestion estimation and clustering method. The study explores innovative techniques to assess traffic congestion levels and cluster vehicular data for effective traffic management. By leveraging advanced algorithms, the proposed approach aims to enhance the real-time decision-making process, ensuring optimal traffic flow in urban road networks. The research contributes valuable insights to the development of intelligent traffic management systems.

Ramachandra, Sujit, Reddy, Vivek, Vellore, Karanth, and Kamath (2016) introduced a novel dynamic traffic management system integrating on-board diagnostics and Zigbee protocol. Published in the International Conference on Communication and Electronics Systems, the study explores advanced technologies to optimize traffic management. By combining on-board diagnostics and wireless communication protocols, the system enhances real-time data collection and analysis, enabling data-driven decision-making for traffic management and contributing to more intelligent urban mobility solutions.

⁵⁰ Field Programmable Gate Array

Rego, Garcia, Sendra, and Lloret (2018) presented a SDN⁵¹ based control system designed for efficient traffic management during emergency situations in smart cities. The research emphasizes the significance of adaptive and responsive traffic management systems, especially during emergencies. By utilizing SDN technology, the proposed system offers dynamic control and optimization of traffic flow, ensuring swift response to emergency scenarios. The paper discusses the architecture and implementation details of the SDN-based control system, highlighting its potential to enhance urban safety and resilience during critical situations.

Saeed and Yousaf (2016) focused on the impact of cognition on user authentication schemes in vehicles. By employing fuzzy logic and artificial neural network technologies, the study enhances user authentication protocols. The research addresses the critical issue of vehicle security, offering advanced cognitive solutions to prevent unauthorized access. The proposed authentication scheme contributes to enhancing vehicle safety and security, promoting the development of intelligent and secure transportation systems.

Subbulakshmi and Prakash (2018) explained the challenges of mitigating eavesdropping in wireless CRNs⁵². The study introduces a novel approach combining fuzzy-based learning and multilevel Stackelberg game theory to enhance network security. The proposed method focuses on optimizing spectrum utilization and minimizing eavesdropping risks. By integrating fuzzy logic and game theory, the research provides a comprehensive solution for ensuring secure communication in CRNs. The study underscores the importance of intelligent security mechanisms in cognitive radio environments, emphasizing the role of advanced computational techniques in enhancing network resilience.

Satpathy, Mohan and Das (2020) introduced a novel healthcare diagnosis system utilizing an IoT-based fuzzy classifier with Field-Programmable Gate Array technology. The study focuses on advancing healthcare diagnostics through intelligent systems. By integrating fuzzy logic and FPGA, the proposed system enhances the accuracy and efficiency of disease diagnosis. The study explores the application of IoT

⁵¹ Software-Defined Network

⁵² Cognitive Radio Networks

devices in healthcare, emphasizing the synergy between IoT technology and advanced computational techniques. The research contributes to the evolution of smart healthcare systems, showcasing the potential of IoT-driven diagnostic solutions.

Sarrab, Pulparambil, and Awadalla (2020) in spite of the broad research conducted on smart traffic system frameworks, smart traffic observing remains an dynamic area of study, driven by emerging innovations just like the Internet of Things and Artificial Intelligence. The consideration of these innovations has the potential to revolutionize decision-making forms and contribute to the development of urban environments. In any case, most existing traffic estimation strategies primarily center on highway and city traffic administration, with limited consideration given to vital minister streets and closed campuses.

Subramani, Berlin, Tripathi, Sandesh, Devi, Bhardwaj, Natarajan and Arul Kumar (2021) found that with the ever-increasing population and stagnant traffic density, traffic prediction has emerged as a significant challenge in today's cities. This situation leads to wasted time, fuel, environmental damage, and even fatalities as people get trapped in congested traffic. Although the field of control system and estimation of traffic congestion has received limited attention from researchers, existing approaches often lack the desired accuracy. To address this issue, they proposes an improved IoT-based traffic estimation and traffic signal control system for smart cities, employing the OWENN algorithm and an Intel 80286 microprocessor. The proposed system consists of five phases: IoT data collection, feature extraction, classification, reducing traffic IoT values, and traffic signal control system. Initially, IoT traffic data is collected from a dataset. Then, the system extracts essential features such as traffic, weather, and direction information. These taken out features are fed into the OWENN classifier, which accurately identifies areas with high traffic congestion. Furthermore, if a specific direction within an area experiences heavy traffic, the system optimizes IoT values using the IBSO⁵³ technique. Finally, the traffic signal control is implemented through the utilization of an Intel 80286 microprocessor. The OWENN algorithm plays a vital role in both traffic signal control and traffic estimation stages. It demonstrates remarkable accuracy, achieving an impressive 98.23% accuracy rate, surpassing

⁵³ Improved Binary Swarm Optimization

existing models, and a high F-score of 96.69%. The experimental results reveal the superior performance of the proposed system compared to cutting edge methods. In conclusion, they presents an efficient IoT-framework approach for traffic signal control and traffic estimation in smart cities. By incorporating the OWENN algorithm and Intel 80,286 microprocessor, the system enhances accuracy and effectiveness, offering a encouraging solution to address the problems posed by traffic congestion in city areas. Effective communication and engagement with the public, particularly when users do not possess smart devices, poses a significant challenge. To address these issues, IoT-based system model was designed to collect, process, and store real-time traffic data in scenarios where traditional methods may be inadequate. Their main objective is to provide real-time traffic information on traffic congestion and rare incidents through roadside, thereby enhancing overall mobility. The citizen time can be save using these early-warning messages, particularly during rush hours, while also broadcasting traffic information from administrative offices. A prototype of the proposed system is implemented to evaluate its practicality, and experimental results demonstrate accurate vehicle detection and low relative error in road occupancy prediction. By leveraging IoT technology and roadside message units, this research contributes to improving the efficiency of traffic management and providing timely information to the public. The findings highlight the potential of the proposed system model in enhancing urban mobility and facilitating informed decision-making. However, further research and development are necessary to refine and expand the system's capabilities and ensure its seamless integration into smart city environments.

Theodoridis, Mylonas, and Chatzigiannakis (2013) focused on traffic components of smart city by exploring innovative approaches for creating intelligent urban environments. The authors emphasize the integration of IoT technologies to enhance city infrastructure and services. The framework proposed in the research aims to leverage IoT sensor, real-time data collection and analysis for efficient urban management. By emphasizing the synergy between IoT systems and city functionalities, the study underscores the potential for creating smarter and more responsive cities.

Thakur, Tanvi Tushar, Ameya Naik, Sheetal Vasari, and Gogate (2016) presented at the International Conference on Communication and Signal Processing, the authors propose a real-time traffic system utilizing the Internet of Things. By harnessing IoT-enabled sensors and communication protocols, the system enables dynamic traffic management strategies. The research highlights the potential of IoT technology in creating adaptive and responsive traffic management solutions, leading to more accurate and sustainable city transportation system. The research contributes valuable insights into the practical implementation of IoT solutions to transform urban spaces into intelligent, data-driven environments.

Tchuitcheu, Bobda and Pantho (2020) presented the implementation of an IoT smart camera system for optimizing traffic lights in smart cities. The smart cameras are used with traffic lights to enhance traffic control mechanisms. By leveraging IoT technology, the informed system offers real-time monitoring and flexible control of traffic signals. The research emphasizes the role of smart cameras in capturing real-time traffic data and enabling intelligent traffic management. The study contributes to the advancement of smart city infrastructures, showcasing the potential of IoT-driven traffic optimization solutions.

Yao, Gao, Wang, Zhang, Jiang and Han (2019) introduced a IoT Traffic Capsule Network tailored for smart cities. The study focuses on refining traffic classification methods using advanced deep learning techniques. By employing Capsule Networks, the proposed mechanism enhances the accuracy of IoT-driven traffic classification. The research emphasizes the importance of precise traffic data analysis for smart city applications. The study showcases the integration of cutting-edge machine learning algorithms with IoT technology, highlighting their synergy in optimizing traffic management strategies. The research contributes valuable insights to the field of smart city technologies, showcasing innovative approaches to traffic. By putting more emphases on the synergy between IoT systems and city functionalities, the study underscores the potential for creating smarter and more responsive cities.

Zhou, Wang, Ma and Zhang (2020) published a paper titled "An IoT-based traffic prediction model for smart cities using deep learning algorithms" in the journal *Sensors*. The authors presented a traffic prediction model that utilizes Internet of Things technology and deep learning algorithms in the context of smart cities. The goal of the study was to improve the accuracy and efficiency of traffic prediction by leveraging real-time data from IoT devices. The proposed model incorporated deep learning algorithms, such as CNNs⁵⁴ and LSTM⁵⁵ networks. These algorithms were employed to extract meaningful features from the collected IoT data and generate predictions for traffic conditions. The authors conducted experiments and evaluated the performance of their IoT-based traffic prediction model using real-world traffic data. They compared the results with other traditional prediction models and demonstrated the superiority of their approach in terms of accuracy and prediction capabilities. The findings of the study indicated that the integration of IoT devices and deep learning algorithms can significantly enhance the accuracy of traffic prediction in smart cities. The authors highlighted the potential applications and benefits of their model in improving traffic management and optimizing transportation systems in urban areas. Overall, the research provides insights into the development of an IoT-based traffic prediction model using deep learning algorithms. It showcased the potential of this approach in the context of smart cities and contributed to the advancement of intelligent transportation systems.

At the end of this literature review on "IoT-based traffic prediction models" I was reminded the transformative potential that the IoT offers to improve the efficiency and reliability of transportation systems. Research on the integration of data-driven approaches, machine learning algorithms, and IoT technologies has provided valuable insights into the dynamics of urban traffic patterns. This study highlights the importance of using the IoT in the context of traffic prediction and provides a glimpse into a future where cities are equipped with intelligent systems that can adapt to the ever-changing demands of urban mobility.

⁵⁴ Convolution Neural Networks

⁵⁵ Long short-term memory

2.6 Artificial Intelligence and Traffic Management

Giffinger and Gudrun (2010) critically examine the effectiveness of smart city ranking systems as instruments for city positioning. By evaluating various methodologies, metrics, and indicators used in city rankings, the study raises important questions about the relevance and accuracy of such assessments. The research prompts discussions on the nuances of smart city evaluations, challenging existing paradigms and advocating for more comprehensive and context-specific approaches to accurately gauge cities' smart capabilities.

Iker, Alessandro and Saioa (2016) comprehensively explored the concept of smart cities, delineating its current state and potential future trajectories. Investigating the essence of smart urban development, the research delves into advanced technologies, data-driven decision-making processes, and sustainable urban practices. By dissecting the multifaceted aspects of smart cities, the study illuminates the intricate interplay between technology, governance, and urban infrastructure, offering crucial insights for urban planners and policymakers striving to create intelligent and sustainable urban environments.

Kikuchi (2009) work, featured in Transportation Research Part C: Emerging Technologies, which explores the transformative role of artificial intelligence in transportation analysis. Kikuchi delves into diverse approaches, methodologies, and applications of AI in addressing intricate transportation challenges. By harnessing machine learning and data analytics, the research showcases how AI techniques optimize transportation systems, emphasizing efficiency, safety, and sustainability.

Navarathna, Pramathi, Vindhya and Malagi (2018) highlighted the diverse applications of artificial intelligence in smart city contexts. Focusing on post-2018 advancements, the study highlights emerging trends and innovative AI solutions adopted by cities to address urban challenges. Their work provides insights into the evolving landscape of AI technologies in shaping smart urban ecosystems.

Tarawneh (2023) explained, how the integration of artificial intelligence into smart cities has revolutionized urban mobility by enhancing traffic flow and reducing accidents. This transformative approach involves optimizing traffic management through intelligent decision-making, leveraging AI algorithms and IoT devices. A robust network infrastructure forms the backbone of successful smart cities, enabling diverse applications and efficient data analysis. In this context, ensuring driver safety is paramount. Remote monitoring of drivers and vehicles emerges as a vital strategy, empowering smart cities to prevent accidents effectively. His research proposed a holistic framework that utilizes real-time traffic data, AI algorithms, and IoT devices to analyze driver behavior, vehicle performance, and road conditions. By remotely monitoring these parameters, the framework not only enhances driver safety but also facilitates timely interventions, thereby saving lives. Additionally, it offers real-time road status updates to drivers and enables cost-effective vehicle maintenance, contributing significantly to a safer, smarter, and more efficient urban environment.

Kumar and Ratan (2023) Analyzed the usefulness of various machine learning algorithms in managing traffic and their real-world applications. Traffic framework is a critical aspect of modern transportation systems, and AI has the power to improve it significantly. they used a dataset from traffic cameras in Delhi to evaluate the performance of four machine learning algorithms: Linear Regression, Decision Tree, Random Forest, and Support Vector Regression. they compared the algorithms based on three performance metrics: Mean Absolute Error, Mean Squared Error, and R-squared. The results showed that Random Forest and Support Vector Regression performed better than Linear Regression and Decision Tree. The real-world applications of AI in traffic management are promising, but ethical considerations must be taken into account. Overall, this research contributes to the growing body of literature on AI for traffic management and provides insights into the potential of machine learning algorithms to improve traffic flow and reduce congestion.

Sreelatha , Mahalakshmi and Yadav(2023) clarified AI based independent traffic control which alludes to the administration and control of traffic stream. In order to gather real-time information on traffic conditions, sensors, cameras, and communication systems are utilized. This information is at that point assessed and handled by AI Algorithms to deliver experiences and make judgment. AI-fuelled

autonomous traffic regulation points to increase framework effectiveness by reducing vehicle blockage, expanding security, and all of the above. The advantage of utilizing independent activity control utilizing AI is the ability to handle and collect huge genuine time information and conclusions are drawn. This empowers the system to alter the activity flow quick in reaction to moving traffic circumstances. Calculations based on AI can moreover be used to learn from past traffic designs and circumstances to form future figures and conclusions that are more precise. For independent activity control, an assortment of AI calculations, which incorporates support learning machine learning, and profound learning, can be applied. Calculations based on Profound learning can be utilized to decipher photographs, video information from cameras, and spotting designs, and patterns in traffic information can be achieved through machine learning calculations. Algorithms for support, learning can be used to memorize from the past and make choices based on compensate signals. To ensure their dependability and security, it is vital to create beyond any doubt that these frameworks are designed and conveyed with the correct assurances. This AI-powered framework can also alter in real-time to moving activity designs and street conditions, making the activity directing handle more responsive and energetic. As a result, there may be a change in traffic-related outflows reductions and fuel productivity. In general, the AI is utilized for the improvement of clever transportation frameworks which has progressed noteworthy, which has the potential to revolutionize activity administration and guarantee a more successful, secure, and feasible transportation framework.

Literature review on "Artificial Intelligence and Traffic Management" reminded me that the integration of AI technologies, machine learning algorithms, and data-driven decision-making processes holds great potential for tackling the complex challenges of modern transportation systems. This literature review revealed new insights into the application of AI in traffic management, highlighting its ability to increase efficiency, reduce congestion, and improve overall transportation sustainability. As we stand at the threshold of a transformative era of smart cities, the research in this area will contribute to the development of intelligent transportation systems.

Chapter – 3

Research Methodology

- 3.1 Introduction
- 3.2 Significance of Research
- 3.3 Problem Statement
- 3.4 Objectives
- 3.5 Hypothesis
- 3.6 Scope of Study
- 3.7 Research Design
 - 3.7.1 Information Collection Procedure
 - 3.7.2 Comparative Analysis of Technologies and Models
 - 3.7.3 Machine Learning Predictive Model Development
 - 3.7.4 Performance Evaluation
 - 3.7.5 Machine Learning Approach Selection
- 3.8 Data Collection Sources
 - 3.8.1 Chicago Dataset: “Chicago_Traffic_1000”
 - 3.8.2 Udaipur Dataset: “Udaipur Traffic”
- 3.9 Tools and Techniques
 - 3.9.1 Weka Tool
 - 3.9.1.1 Components and Techniques
 - 3.9.2 Python
- 3.10 Summary

3.1 Introduction

The Internet of Things is a technological system that replaces human interaction with a variety of tools and gadgets. This makes it possible for smart cities to sprout up all over the world. The internet of things, which contains several technologies and allows interactions between them, has sped up the development of smart city systems for sustainable living, greater comfort, and increased productivity for people. A wide range of industries are significantly impacted by the Internet of Things for Smart Cities, and its operation depends on a number of different underlying technologies. The Internet of Things in Smart Cities is in-depth examined in this study. Following the architectures, networking, and artificial algorithms that support these domains in IoT-based Smart City systems, the major elements of the IoT-based Smart City environment are provided.

IOT, Data science, Artificial Intelligence, and Machine Learning are just a few of the technologies that are coming together to form the Fourth Industrial Revolution, which has the potential to dramatically alter how public transportation information systems are seen. Artificial intelligence technology is capable of self-correcting and has a brain like a person. A subset of artificial intelligence called machine learning replicates how people learn. With the aid of IoT, it is necessary to investigate the usage of artificial intelligence and machine learning to address issues with the public transportation information system. To improve users' experiences using public transportation, research must be conducted on various artificial intelligence and machine learning algorithms using IoT and GPS to show real-time bus arrival time information, bus interval information, and seat availability.

3.2 Significance of Research

Research methodology is the way in which research problems are solved systematically. Research methods are the strategies, processes or techniques utilized in the collection of data or evidence for analysis in order to uncover new information or create better understanding of a topic. It is a science of studying how research is conducted scientifically. Under it, the researcher acquaints himself/herself with the various steps generally adopted to study a research problem, along with the underlying logic behind them. Research methodology deals with scientific results, the hypothesis

which is the outcomes of the objectives with the results. Research methodology is defined by the source of information on which the work will be done. The tools and technology to work on the information are chosen carefully to result in the positive outcome of the hypothesis.

3.3 Problem Statement

Our research work is an attempt to understand the importance of IoT, Artificial Intelligence and machine learning in solving commuting problems. The central problem focuses on:

“To Explore and Analyze the Role of IOT, Artificial Intelligence and Machine Learning in Solving the Commuting Problems of Smart Cities”

The research work studies the level of automation work, commuting issues, explore the technologies for improving reliability of public transportation system of smart cities, simulate data using IOT, Artificial Intelligence and Machine Learning Algorithms to resolve commuting problems and find trends, issues, challenges, suggestions, and future potential of commuting problems in smart cities. Overall, the research work finds the usage of IOT, Artificial Intelligence and Machine Learning techniques in solving the commuting problems for hassle-free journey in smart cities.

3.4 Objectives

An objective creates vision and converts it into measurable targets. Objectives direct our activities toward achieving the goals and vision. Objectives bring motivation among stakeholders and help in decision making. It brings clarity, focus, provides direction and motivates us to measure progress. Objectives create the environment of clear and transparent communication to achieve the targets. It also helps us to continuously improve to reduce errors and bring better results close to set targets. The main urge behind our research is to solve commuting problems using IOT, Artificial Intelligence and Machine Learning technology and making smart public transport systems for smart cities. The objectives of research are:

1. To study the commuting and traffic congestion issues associated with smart cities.
2. To analyze the different technologies used for enhancing the transportation system for smart cities.
3. Comparative analysis of various existing IoT based traffic prediction models and traffic control systems for smart cities.
4. To develop a machine learning predictive model for Smart Transportation System.
5. To analyze the performance of machine learning predictive model for Smart Transportation System based on various performance measures.
6. To address the implementation issues related to deployment of Intelligent Transportation System.
7. To identify the most appropriate machine learning approach related to traffic congestion monitoring and transportation management system for Smart cities.

The main goal of research work is to use various analyzing tools to create model, and study commuting problems in smart cities and provide solutions with the help of IoT, AI and ML algorithms for stress free smart commuting. Research being focused on smart public transport information systems to achieve above mentioned objectives.

3.5 Hypothesis

Hypothesis is nothing but a tentative statement or proposed explanation made based on limited evidence as a starting point for further investigation. The following hypothesis is being tested for proposed research work.

H₀1: There is no significant difference between technologies used for enhancing the transportation system for smart cities.

H₀2: The Machine learning-based traffic prediction models have average performance scores of greater than or equal to 75%.

Hypothesis is playing vital role in the field of IoT, AI and ML for problem formulation, problem solving, testing, continuous learning, Decision making and predictive modelling. Hypothesis help us in successful completion of the project.

3.6 Scope of Study

Our study is focused on Implementation of IoT, Artificial Intelligence and Machine Learning Algorithms to solve commuting problems. Smart Transportation Systems are increasingly integrating IoT, Machine Learning and Artificial Intelligence to revolutionize transportation networks. Machine Learning algorithms optimize traffic management by predicting congestion and adapting traffic signals in real time. Predictive maintenance powered by Machine Learning prevents infrastructure failures, saving costs and enhancing safety. Artificial Intelligence-driven public transportation planning improves routes and schedules based on dynamic factors. The limiting factors or challenges are data collection complexity, data privacy and security, continuous data access, Interoperability issues, overloaded wireless networks and continuous model monitoring and adaptation. The scope of IoT combined with Artificial intelligence and Machine Learning is very vast and promising.

3.7 Research Design

The research design for the study encompasses elements of both quantitative and qualitative research designs. It incorporates literature review, data collection, comparative analysis, model development, performance evaluation, and addressing implementation issues. The quantitative aspect involves data collection on commuting patterns, traffic flow, congestion levels, transportation infrastructure, and performance metrics. Statistical analysis techniques are applied to compare technologies, models, and performance measures. Machine learning algorithms are used for predictive modelling and performance evaluation. The qualitative aspect includes the literature review, which synthesizes existing knowledge and identifies gaps. Interviews, surveys, and focus groups may be conducted to gather insights from commuters, transportation authorities, and urban planners, addressing implementation issues and obtaining feedback on the effectiveness of transportation systems. Overall, this research design can be considered a mixed methods approach, combining quantitative analysis and qualitative insights to provide a comprehensive understanding of smart transportation systems and traffic management in smart cities.

3.7.1 Information Collection Procedure

Information is gathered on commuting patterns, traffic flow, congestion levels, transportation infrastructure, and smart city initiatives from relevant sources such as transportation authorities, urban planning departments, and IoT sensor networks. Collected information on the performance of existing transportation technologies and traffic prediction models etc. The different Information collection methods used are:

1. By Observation

In observation procedure, various companies, traffic departments, public places were visited to understand the problems and solutions adapted. The following places were visited:

- a) Pyrotech company Udaipur Rajasthan India to understand Automation work of Street lights used in smart cities.
- b) AGV Ambernath Maharashtra India to get acquainted with IoT Enabled video Door phone, Smart IoT Enabled battery management system.
- c) The Brihan Mumbai Electric Supply and Transport Undertaking GM's office BEST Marg Colaba Mumbai to get traffic congestion data.
- d) BEST Planning department Wadala Mumbai to collect fleet management and traffic congestion data.
- e) Udaipole Udaipur Rajasthan India pay and park to understand automation of parking allotment and to collect parking allotment data.

Countless hours spent on observation and contacting people had given me insights of commuting problems in smart cities. This helped me to understand and analyze the actual traffic problems of smart cities. I am thankful to authorities who not only shared their knowledge and experience with me but also helped me in connecting with subject matter.

2. Through Interviews

Personal interactions were conducted with leading industry people like chairman of Pyrotech company Udaipur Rajasthan, Director of AGV Systems MIDC Ambernath Maharashtra, General manager of BEST Bhavan Colaba Mumbai Maharashtra and Deputy Head Planning Department Wadala Mumbai Maharashtra. All the dignitaries

were interviewed on usage of IoT, Artificial intelligence and Machine learning in smart city transportation. Their opinion and views were considered to carry out research in IoT, Artificial intelligence and Machine learning to solve commuting problems in smart cities.

3. Through Questionnaire

The Google survey form was designed for fog computing, IoT, Artificial intelligence and Machine Learning algorithm questionnaire and feedback was collected online using designed google forms. The data was collected from eminent personalities in the technical field and from different regions / places. This feedback data was used for Hypothesis testing using python as tool.

4. Through Participation in Conference

To collect insight of subjects and dive deeper into the details of IoT, Artificial intelligence and Machine learning algorithm following conferences were attended.

- a) 12th International Conference on **“Sustainable Global trends: Planet People and Profit”** on 16-17 April 2021 Organized by Pacific University Udaipur.
- b) Virtual International conference **“Emerging Era of Applications of Computers”** on 15-16 January 2022 Organized by Pacific University Udaipur.
- c) 4th Springer International Conference on **“Mobile Computing and Sustainable informatics”** on 11-12 January 2023 Organized by Tribhuvan University Nepal.
- d) 38th Indian Engineering Congress Conference on **“Re-imagining tomorrow: Shaping the future through Disruptive and Interdisciplinary technologies”** on 27-29 December 2023 Organized by The Institution of Engineers India at Jabalpur.

I am grateful to all Conference Organizers and my fellow presenters or researchers who not only provided me with the platform to showcase my talent but also helped me in gaining rich technical experience by actively participating in conferences to collect data. These gatherings have provided me the stage for scholarly exchange which helped me allot in coming out with Machine learning solutions for traffic congestion problems.

5. Through Participation in Competitive Exams

To set high research standards and equip ourselves with the latest trends and techniques various Machine learning courses were successfully completed from high ranking institutes listed below.

- a) EDUXLABS (Esoir Business Solution LLP) Certificate course on **“Applied Deep learning for Medical data analysis”** from 27.10.2020 to 9.11.2020.
- b) Faculty Development program on **“Research and Publication Ethics”** from 2.01.2021 to 7.01.2021 Organized by Pacific Business School Udaipur.
- c) Two days National Conclave on **“Intellectual Property rights”** on 26.07.2021 and 27.07.2021 Organized by Pacific University Udaipur.
- d) Twelve weeks Online course on **“Data Analytics with Python”** from January 2023 to April 2023, Organized by NPTEL (Funded by Govt of India) and successfully completed with consolidated score of 82% with **elite silver medal**.
- e) Four weeks Online course on **“Python for Data Science”** from July 2023 to August 2023, Organized by NPTEL (Funded by Govt of India) and successfully completed with consolidated score of 78% with elite silver medal and secured **All India rank One** (Top 5%).

Participation in certification courses and competitive exams helped me in sharpening my technical skills, enhanced my understanding about machine learning algorithms and encouraged me to use machine learning algorithms to solve real life traffic congestion problems. Competitive exams have produced the spirit of continuous learning in me and motivated me to take up the traffic congestion problems.

3.7.2 Comparative Analysis of Technologies and Models

Analyze various technologies used for enhancing transportation systems in smart cities, including intelligent traffic management systems, connected vehicles, smart parking systems, and transportation network optimization algorithms. Evaluate the strengths and weaknesses of different technologies based on factors such as effectiveness, scalability, cost implications, and integration capabilities. Compare and analyze existing IoT-based traffic prediction models and traffic control systems, considering accuracy, real-time capability, scalability, and adaptability.

3.7.3 Machine Learning Predictive Model Development

Designed and developed a machine learning predictive model for smart transportation systems using the gathered data and insights from the literature review. Utilize appropriate machine learning algorithms such as regression, decision trees to predict traffic congestion and optimize transportation routes. Train and validate the predictive model using historical traffic data and real-time traffic information.

3.7.4 Performance Evaluation

The performance of the developed machine learning predictive model are analyzed using various performance measures such as prediction accuracy, incorrectly classified instances, kappa score and various confusion matrix parameters such as TP rate, FP rate, precision, recall and F1-score. Compare the performance of the model with existing traffic prediction models and assess its effectiveness in predicting traffic congestion and optimizing transportation systems.

3.7.5 Machine Learning Approach Selection

Evaluate different machine learning approaches related to traffic congestion monitoring and transportation management systems for smart cities. Compare the suitability and performance of various machine learning algorithms, such as Random Forest, Random Tree, Bayes net, naïve Bays, SMO, IBK, Logistic, KStar and Multiclass classifier in addressing traffic congestion issues. Recommend the most appropriate machine learning approach for traffic congestion monitoring and transportation management in smart cities. Summarize the research findings, including the analysis of commuting patterns, traffic congestion issues, technologies, and machine learning predictive models. Provide recommendations for policymakers, urban planners, and transportation authorities on improving smart transportation systems, mitigating traffic congestion, and implementing intelligent transportation solutions in smart cities.

3.8 Data Collection Sources

The primary goal of the research work is to examine and find the technologies associated with the smart city project. To find new emerging technologies that can impact the transportation system in smart cities also improving the reliability of public

transportation system of smart cities. Predictive model based on Artificial Intelligence and Machine Learning Algorithms is being developed to resolve commuting problems. Following are the secondary sources which were being used for data collection:

- i. Research papers and articles from Springer, IEEE⁵⁶, Wiley, Sage, Elsevier etc.
- ii. Computer science engineering, IT⁵⁷ journals for the comparative analysis.
- iii. Government websites that offer information on training initiatives taking place in the area.
- iv. Books on IoT, Artificial Intelligence, Machine Learning, Deep Learning & Data Science.
- v. The datasets are being collected from:
 - a. Kaggle
 - b. Chicago Traffic data
 - c. tomtom.com
 - d. Data World etc.

I extend my gratitude to all Journal publishers, Book publishers and websites for providing valuable data sets and information, without which it would not have been possible to conclude my research.

3.8.1 Chicago Dataset: “Chicago_Traffic_1000”

Chicago Traffic Tracker is being used as a secondary source to collect data for model development. The Chicago Department of Transportation maintains and publishes the Chicago Traffic Tracker website, which provides real-time traffic updates along arterial streets. The site contains data on Average Daily Traffic volumes, traffic signal locations, intersections with automated red-light enforcement cameras, automated speed enforcement camera placements, dynamic messaging sign locations, and at-grade rail crossings that affect travel to and from Midway Airport. The website is constantly evolving as an ongoing project, with Chicago Department of Transportation anticipating future enhancements such as the addition of route-level alerts, live images from traffic cameras, truck routes etc. Following are the credentials of the Chicago data set.

⁵⁶ Institute of Electrical and Electronics Engineers

⁵⁷ Information Technology

- Instances: 1000
- Attributes: 21
- Details: Chicago Traffic Tracker: Congestion Estimates by Traffic Regions
- Data Owner: Chicago Department of Transportation
- Time Period: March 2018 - Current, with occasional gaps due to system maintenance and other temporary technical issues.
- Frequency: Data are updated every 10 minutes
- Data Reduction: Number of instances included were the latest 1000 records.

The Chicago Traffic Tracker shown in figure 3.1 utilizes real-time GPS traces from Chicago Transit Authority buses to estimate traffic congestion on nonfreeway arterial streets. Updated every 10 minutes, it provides two types of congestion estimates: Traffic Segments offer observed speeds for one-half mile segments in a specific traffic direction, covering around 300 miles of principal arterials, while Traffic Regions provide average traffic conditions for all arterial street segments within a region composed of two or three community areas with comparable traffic patterns.

Figure 3.1: Chicago Traffic Tracker

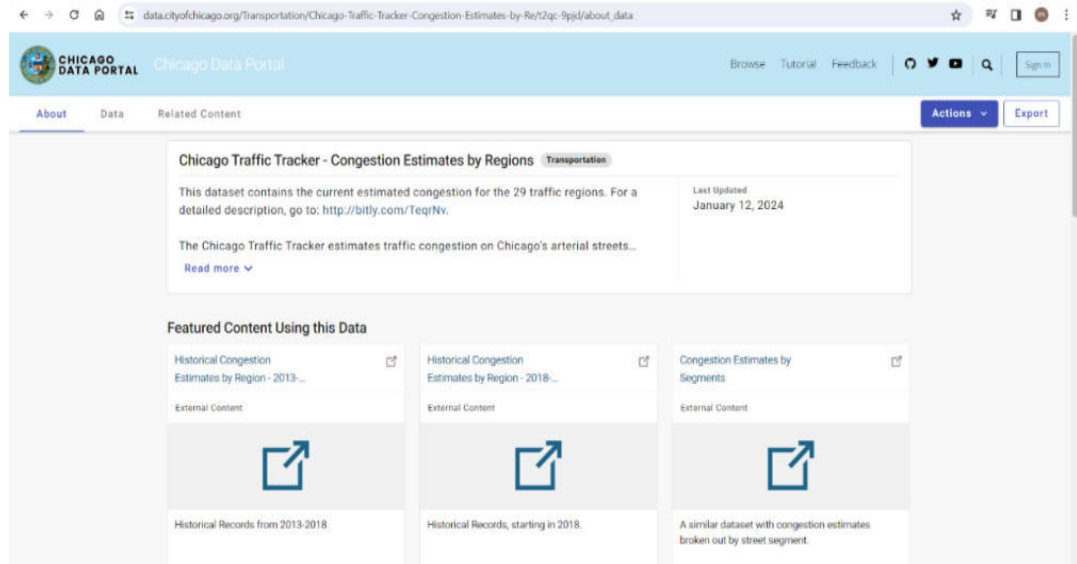


Source: Chicago Traffic Tracker [16]

The attributes such as Time, Region_ID, Speed, Region, Bus_Count, Num_Reads, Hour, Day_Of_Week, Month, Description, Record_Id, West, East, South, North, Nw_Location, Se_Location, Community Areas, Codes Assigned, Wards and Class

Label were being considered. Detail view of the traffic can be seen in the “Chicago Traffic Tracker” with various tools available in the real time application. The relevant dataset can also be downloaded from Chicago city portal [17] as shown in figure 3.2.

Figure 3.2: Chicago Data Portal



Source: Chicago City Portal [17]

The Chicago Traffic Tracker, managed by the Chicago Department of Transportation, offers real-time updates on arterial street traffic conditions, Average Daily Traffic volumes, and various relevant information such as signal locations and camera placements.

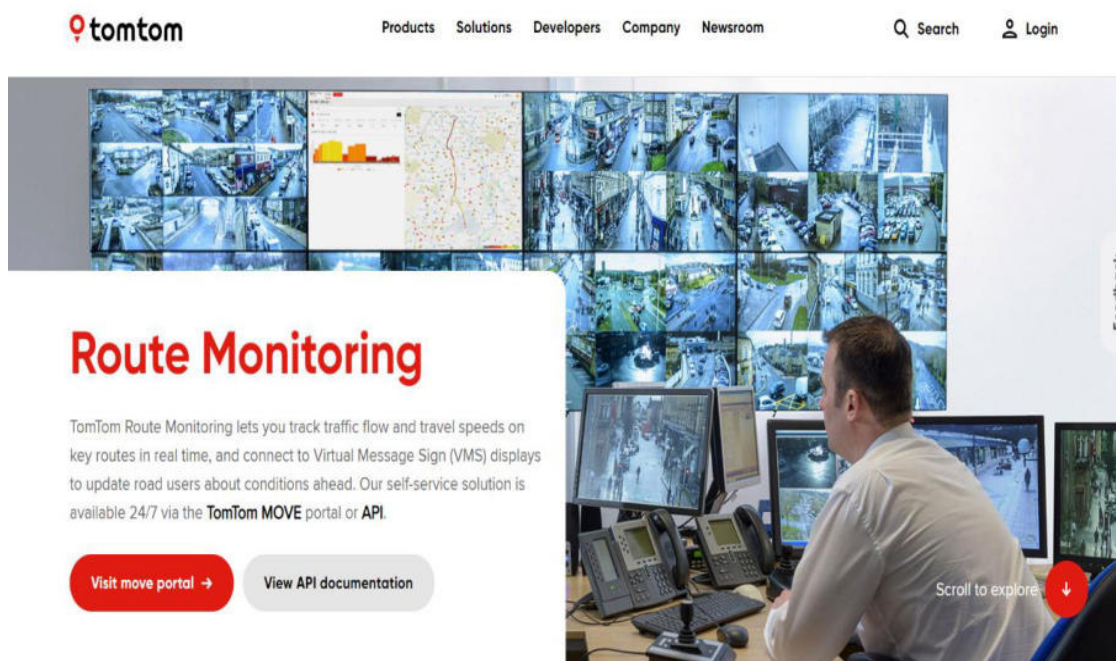
3.8.2 Udaipur Dataset: “Udaipur Traffic”

TomTom traffic server is used to collect real time traffic data of Udaipur city. TomTom traffic server provides information about traffic density, speed of vehicles, traffic congestion, number of vehicles on the road, delay time on the roads and historical traffic values at world level. TomTom server provides various traffic statistics through various products like Traffic stats, Route Monitoring, O/D Analysis, Junction Analytics and Road Analytics. These products are available through MOVE portal or via API. These products are available on trial basis for one month or also on usage basis. These products are helpful for transport planners, researchers to make smart decisions. Following are the credentials of tomtom server data set.

- Number of Instances: 1000
- Number of Attributes (After Feature Extraction): 7
- tomtom Site: <https://www.tomtom.com/traffic-index>
- Product Details: Traffic stats, Route Monitoring, Junction Analytics
- Data Owner: tomtom
- Time Period: Sept 2023 – October 2023, with occasional gaps due to system maintenance and other temporary technical issues.
- Frequency: Data are updated every 3 minutes (Approximately 40 samples collected / day)

Figure 3.3 shows Route monitoring product from tomtom server. All key routes of the world from different countries can be monitored in real time. It gives route information like vehicle speed and tracks traffic flows. Custom routes of any city can be defined to get detailed information of travel times, traffic delays and speed of vehicles. This will help traffic planners to take smart decisions.

Figure 3.3: tomtom Route Monitoring

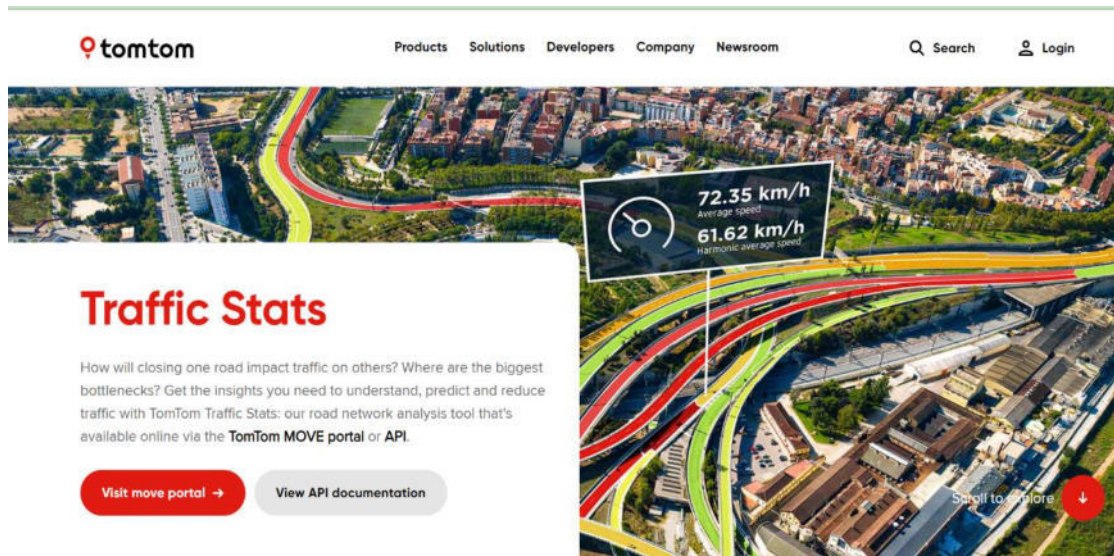


Source: Route Monitoring product [18]

Figure 3.4 shows Traffic Stats product from tomtom server. It gives information like traffic density, travel times and traffic density on the roads. Along with present real

time traffic data it also provides the largest historical traffic database. This present and past traffic data helps planners in traffic management.

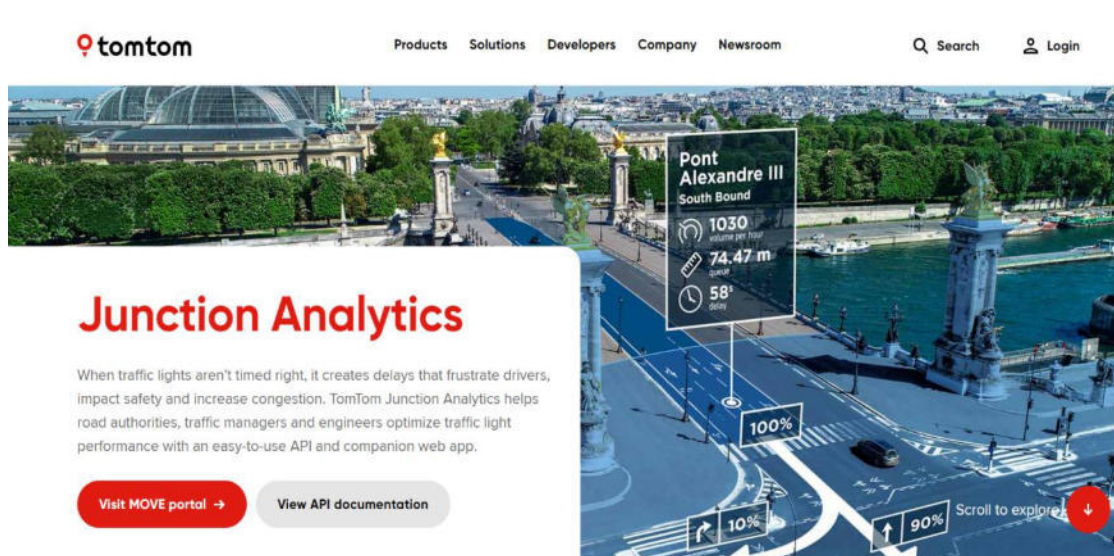
Figure 3.4: tomtom Traffic Stats



Source: Traffic Stats Product [19]

Figure 3.5 shows Junction Analytics product from tomtom server. It gives information about city junction traffic. This analytical tool gives understanding of how drivers move through interactions to control traffic signal timings on junction to reduce congestion.

Figure 3.5: tomtom Junction Analytics



Source: Junction Analytics Product [20]

3.9 Tools and Techniques

To evaluate the performance and effectiveness of the smart transportation systems, various metrics and statistical methods were employed. Percentage analysis, measures of central tendency, measures of dispersion, cumulative frequency, correlation coefficient, and regression analysis were used to analyze the collected mining data. Hypothesis testing was performed using the Chi-Square test. The study involved the simulation of data and model building, utilizing multiple regression analysis. A conceptual model based on regression was developed to examine the significance of different technologies and the reliability of public transportation systems in smart cities. Additionally, the study aimed to assess the usefulness of IoT, Artificial Intelligence, and Machine Learning-based models in addressing commuting problems. The Weka tool and Python were used for simulation and predictive analysis. Overall, the study employed a research design that combined qualitative and quantitative research approaches. The qualitative nature of the study facilitated the exploration of various concepts and ideas, leading to findings and recommendations for improving smart transportation systems in smart cities.

3.9.1 Weka Tool

The Weka Experimenter is a tool within the Weka software package that allows users to design, run, and analyze machine learning experiments systematically. It is particularly useful for comparing multiple machine learning algorithms and configurations on various datasets, helping researchers and practitioners make informed decisions about which algorithms work best for their specific tasks. Here's a more detailed explanation of the Weka Experimenter's key features and functionalities:

a. Experiment Design: The Experimenter allows users to design experiments by specifying different machine learning algorithms, datasets, and evaluation metrics. Users can choose from a wide range of classification, regression, and clustering algorithms available in Weka. They can also select multiple datasets to test the algorithms' performance across different data domains.

b. Parameter Sweeping: Users can explore the effect of different parameter settings on the performance of machine learning algorithms. The Experimenter enables parameter sweeping, where users can specify a range of values for certain parameters of the algorithms. The Experimenter then systematically runs experiments with different parameter combinations to find the optimal settings.

c. Cross-Validation and Evaluation Metrics: The Experimenter supports various techniques for evaluating machine learning models, including cross-validation (k-fold cross-validation, leave-one-out cross-validation, etc.). Users can select different evaluation metrics such as accuracy, precision, recall, F1-score, and others to assess the performance of the algorithms.

d. Batch Execution: The Experimenter can run experiments in batch mode, allowing users to schedule multiple experiments to run sequentially or concurrently. This feature is particularly useful for running large-scale experiments overnight or on computing clusters.

e. Result Analysis and Comparison: After the experiments are completed, the Experimenter provides detailed summary reports and visualizations of the results. Users can compare the performance of different algorithms on various datasets using statistical tests and visualizations like charts and graphs. This comparative analysis helps users identify the best-performing algorithms and configurations for their specific problem domains.

f. Reproducibility: The Experimenter ensures the reproducibility of experiments by allowing users to save the experiment configurations and results. Researchers can share these configurations and results with others, making it easier to validate and replicate experiments.

g. Integration with Other Weka Tools: The Experimenter seamlessly integrates with other Weka tools and interfaces, allowing users to utilize preprocessing techniques, attribute selection methods, and various machine learning algorithms available in Weka.

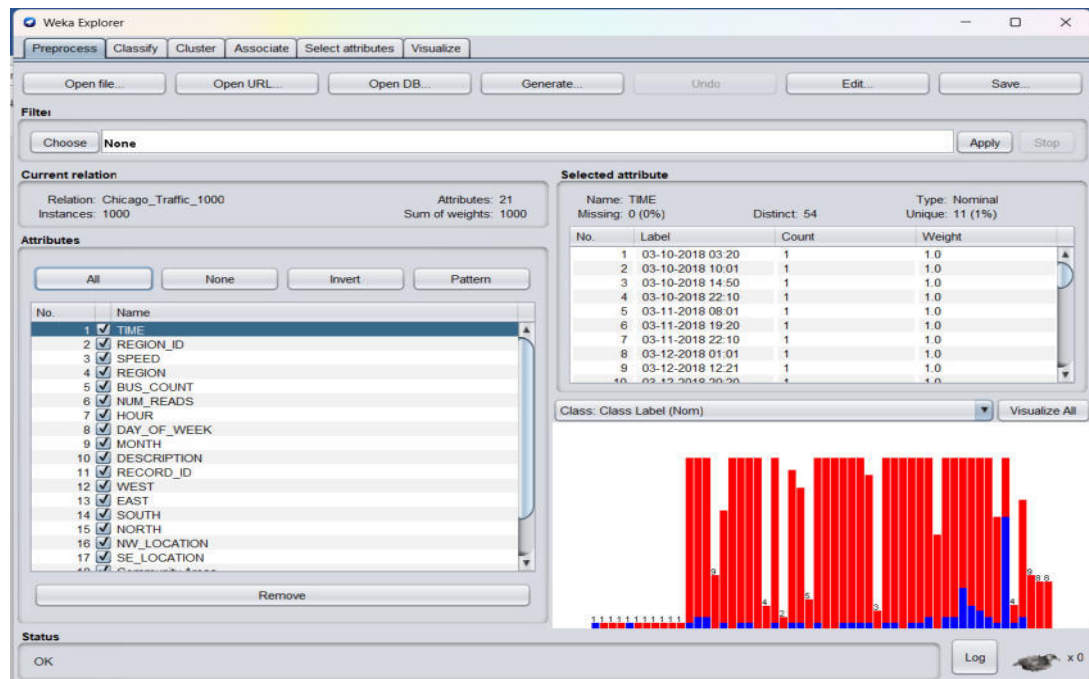
3.9.1.1 Components and Techniques

The Weka Experimenter provides a user-friendly environment for designing, running, and analyzing machine learning experiments. Its capabilities make it a valuable tool for researchers and practitioners who want to systematically evaluate and compare different machine learning algorithms and configurations on multiple datasets. Some key tools and techniques available in Weka Explorer include:

- Preprocessing Tools
- Classification Algorithms
- Clustering Algorithms
- Attribute Selection
- Evaluation Techniques
- Visualization Tools

Figure 3.6 shows the preprocessing tool in Weka applied to Chicago_Traffic_1000. This interface includes details about the number of instances, number of attributes, relation, selected attribute tab etc. In the present scenario the details of the Time attribute are shown in the selected attributes tab.

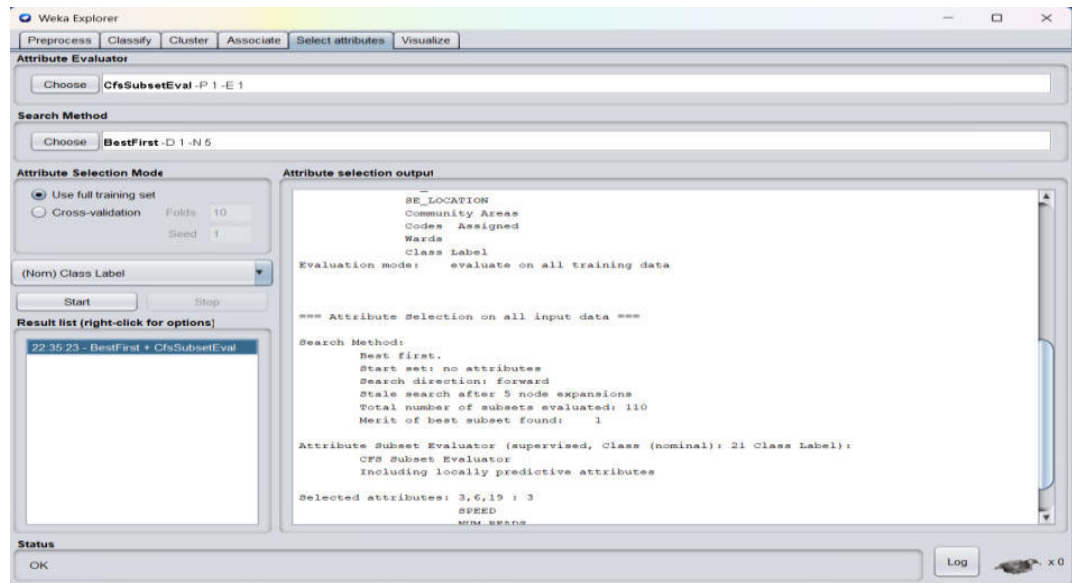
Figure 3.6: Preprocessing Step Tool



Source: Weka Tool

Figure 3.7 shown below is the attribute selector. Many times, data set contains redundant attributes which are insignificant in the analysis, therefore removing the unwanted attributes from the data set is necessary to develop good machine learning algorithms.

Figure 3.7: Attribute Selection



Source: Weka Tool

Weka has the facility to select required attributes for the designing machine learning algorithm. All selected attributes of Chicago_Traffic_1000 dataset can be seen graphically in the following figure 3.8.

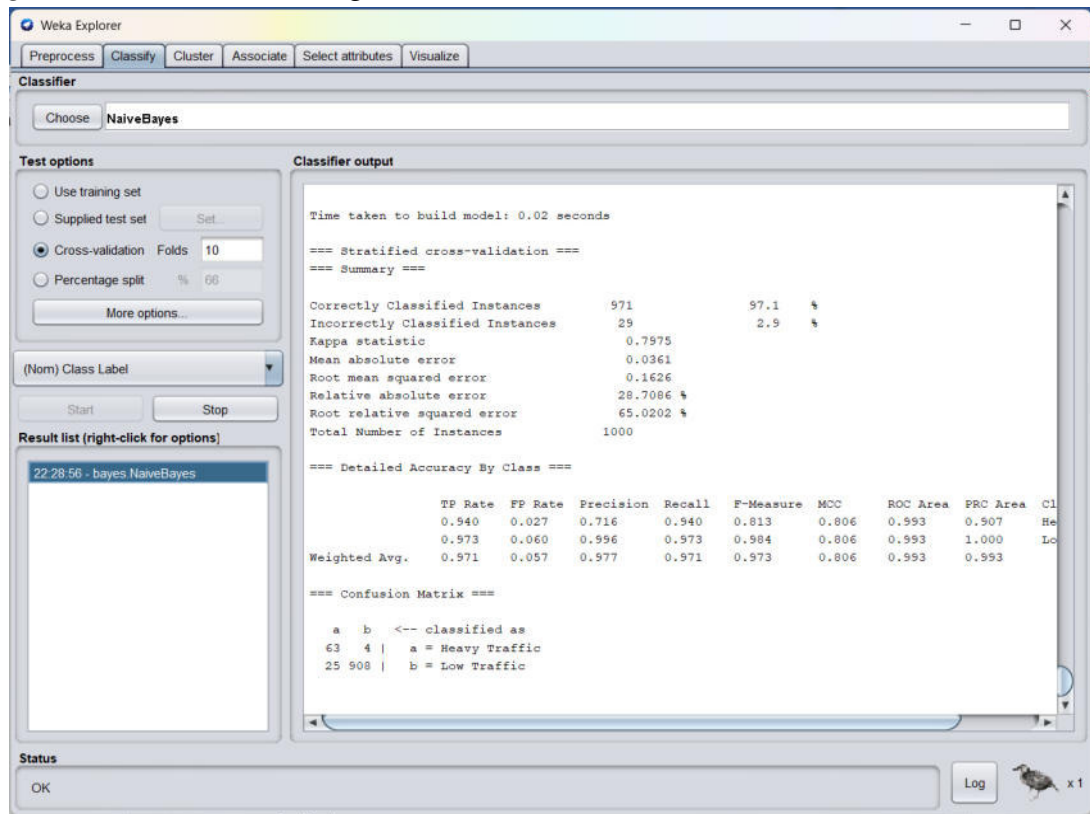
Figure 3.8: All Attribute (Applied on Dataset Chicago_Traffic_1000)



Source: Weka Tool

Figure 3.9 shown below is the Classifier window in Weka tool which is used to select classifier algorithm from the set of available Machine learning algorithms. This window has four testing options Training set, supplied test set, Cross-Validation and Percentage split. We have used number of folds and percentage split options under cross-validation option.

Figure 3.9: Classification Algorithms Tool

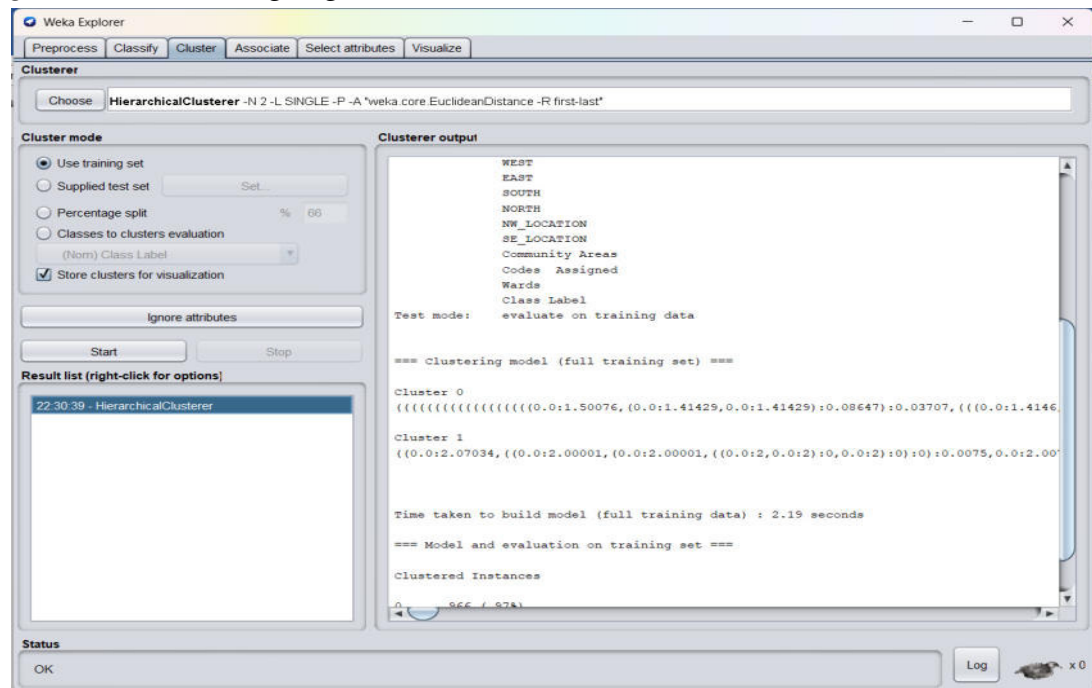


Source: Weka Tool

In the absence of your own training set or client supplied training set the cross-validation or percentage split options are selected. In our research 10-fold cross – validation, 25-fold cross – validation and 30% split options are used on nine different classification algorithms to develop machine learning algorithms for traffic forecasting.

Figure 3.10 shows the Clustering Algorithm window which is used to find groups of similar types of instances in the dataset. Weka allows execution of several clustering algorithms such simple KMeans, Hierarchical Clusters and so on.

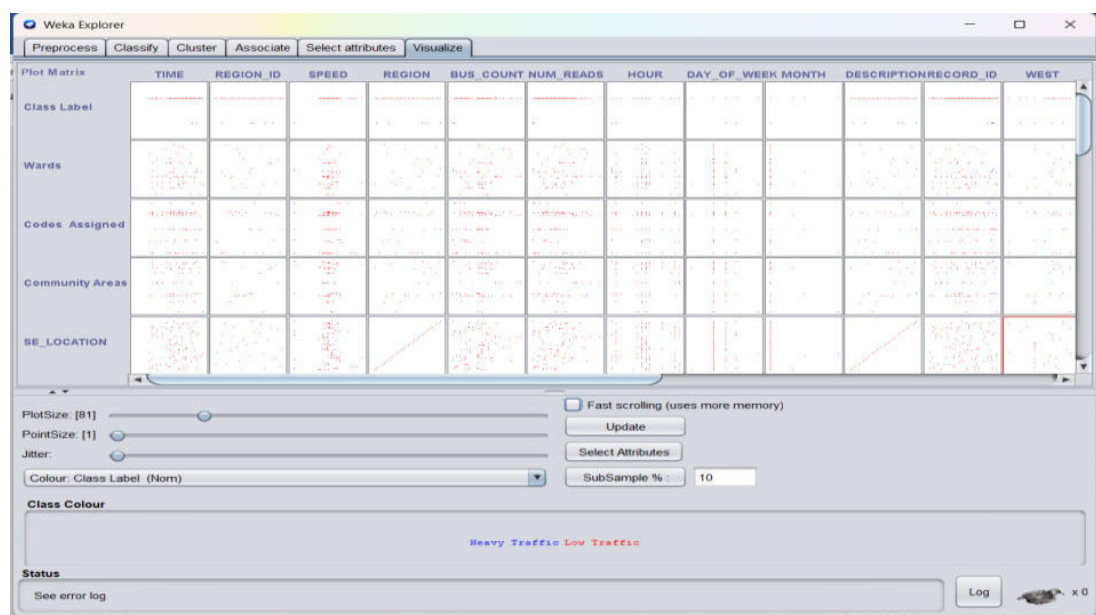
Figure 3.10: Clustering Algorithm



Source: Weka Tool

Figure 3.11 shown below is Visualization Tool is the Graphical user interface tool which allows users to visualize their processed data after execution of Machine learning algorithms. There are various graph attributes which can be used to control graphical plots. It has various options like plot size, attribute select, class color selection to control graphical display output.

Figure 3.11: Visualization



Source: Weka Tool

These figures show the application of various tools and techniques Weka being applied to the considered data set Chicago_Traffic_1000. The classification and clustering algorithms were being applied based on configuration settings either cross validation or split percentage.

3.9.2 Python

Python offers a wide range of libraries and tools that make feature extraction from various types of data (such as text, images, and numerical data) efficient and convenient. Here are some popular libraries or tools/techniques used for feature extraction in Python:

- Scikit-learn.
- Natural Language Toolkit
- OpenCV
- Pandas
- Feature-engine

The combined featured extraction matrix being designed using the python programming language. Feature extraction techniques are essential for transforming raw data into a format suitable for machine learning. For numerical data in Pandas Data Frames, specific features are extracted by selecting relevant columns. These methods enable effective preparation of data for machine learning models. Also, other statistical analysis is done using Python.

a. Hypothesis Testing Tools:

To test the framed null hypotheses, three types of statistical methods were used. The applied tests were Pearson Chi-Square test, ANOVA Test and T-Test.

b. Chi-Square Test:

The Chi-Square test is a statistical method used to determine if there is a significant association between categorical variables. It compares observed frequencies with expected frequencies, assessing whether any differences are statistically significant.

c. ANOVA Test:

ANOVA is a statistical technique used to compare means of three or more groups to determine if there are statistically significant differences. It assesses the variability within and between groups, helping to understand whether observed differences are likely due to random chance or actual group effects.

d. T-Test:

The t-test is a hypothetical test which is used to find out whether the average calculated for criteria from sample data is different from a value claimed by researchers. The one sample t test left tail or right tail is used to compare the mean of sample data with claimed value of researchers.

3.10 Summary

This research work aims to address various aspects of smart transportation systems and traffic management in smart cities. It encompasses a mixed methods approach, combining quantitative analysis and qualitative insights. The research objectives include studying commuting and traffic congestion issues, analyzing technologies for enhancing transportation systems, comparing IoT-based traffic prediction models and traffic control systems, developing a machine learning predictive model, evaluating its performance, addressing implementation issues, and identifying the most appropriate machine learning approach for traffic congestion monitoring and transportation management. The research begins with a comprehensive literature review to identify gaps in existing research and frameworks. Data collection involves gathering information on commuting patterns, traffic flow, congestion levels, transportation infrastructure, and smart city initiatives. Comparative analysis is conducted to evaluate different technologies and IoT-based traffic prediction models, considering factors such as effectiveness, scalability, cost implications, and integration capabilities. The research also involves the development of a machine learning predictive model for smart transportation systems. Historical and real-time traffic data are utilized to train and validate the model. Its performance is evaluated using various performance measures, comparing it with existing models. Implementation issues related to the deployment of intelligent transportation systems are addressed, including infrastructure requirements, data privacy and security, scalability, and user acceptance.

Recommendations and strategies are proposed to overcome these challenges. Furthermore, different machine learning approaches for traffic congestion monitoring and transportation management in smart cities are evaluated. The suitability and performance of various algorithms, such as neural networks, support vector machines, or ensemble methods, are compared to identify the most appropriate approach.

In conclusion, this research aims to provide a comprehensive understanding of commuting patterns, traffic congestion, technologies, IoT-based models, machine learning predictive models, implementation issues, and machine learning approach selection for smart transportation systems in smart cities. The findings will contribute to the development of effective transportation strategies, the mitigation of traffic congestion, and the successful implementation of intelligent transportation solutions in smart cities.

Chapter – 4

Feature Extraction and Data Processing Using IoT

- 4.1 Overview
- 4.2 Feature Selection Methods
 - 4.2.1 Info Gain Attribute Eval
 - 4.2.2 Correlation Attribute Eval
 - 4.2.3 Classifier Attribute Eval
 - 4.2.4 Cfs Subset Eval
 - 4.2.5 Gain Ratio Attribute Eval
 - 4.2.6 OneR Attribute Eval
 - 4.2.7 ReliefF Attribute Eval
 - 4.2.8 Symmetrical Uncert Attribute Eval
- 4.3 Feature Extraction Using Multiple Regression
- 4.4 Combined Feature Selection Matrix
- 4.5 Rank and Percentile
- 4.6 Summary

4.1 Overview

Dataset contains hundreds of attributes. However, not all attributes are required to complete the machine learning task. A feature extraction and selection algorithms are used to determine the importance of the attributes. Rather than processing all attributes, only relevant attributes are included in the machine learning process. This reduces processing time and also improves the performance of the task. Therefore, attribute extraction and selection algorithms are applied before applying data learning tasks such as classification, clustering, and outlier analysis.

Feature extraction is used to extract features from the collected data. Feature selection is the extension of feature extraction. Feature selection is the process of selecting a subset of relevant features (variables, predictors) to use in model building [Chong and Ng 2016]. In normal situations, domain knowledge plays a key role and allows you to select the features that seem most important. For example, when predicting sales of Furniture, the type of furniture, size of furniture and budget of customer may be important. Feature selection follows Feature extraction which simply selects required features and remove unwanted or redundant features from the Data set[Coutard 2014]. Feature selection performs following functions.

1. Remove Features with missing values
2. Remove highly uncorrelated features
3. Remove Features with low variance

Feature extraction and data processing using IoT involves extracting relevant information or features from the data collected by IoT devices and processing them for further analysis or application. Feature extraction is an important step in machine learning, and its goal is to reduce the dimensionality of input data while preserving important information. Extracting relevant features is important to improve the efficiency of machine learning algorithms, reduce computational complexity, and improve model performance. The step-by-step overview of the process is given below:

Data Collection: IoT devices collect data from various sensors, such as temperature, humidity, motion, or light sensors. The data can be collected continuously or at regular intervals and transmitted to a central server or cloud platform for processing.

Preprocessing: Raw data collected from IoT devices often requires preprocessing to remove noise, handle missing values, or normalize the data. This step ensures that the data is in a suitable format for further analysis.

Feature Extraction: Feature extraction involves identifying and extracting relevant features from the preprocessed data. Features are specific measurements or characteristics that capture the essential information for the intended analysis or application. For example, in a smart home scenario, features could include temperature, occupancy status, or energy consumption patterns.

Selection and Dimensionality Reduction: Depending on the application, it may be necessary to select a subset of features or reduce the dimensionality of the data. This step aims to eliminate irrelevant or redundant features, improving computational efficiency and reducing the risk of overfitting in machine learning models.

Data Integration: In some cases, data from multiple IoT devices or sources may need to be integrated to derive meaningful insights. Integration can involve combining data from various sensors, time synchronization, or merging data from different locations or devices.

Data Analytics: Once the relevant features have been extracted and processed, various analytics techniques can be applied to gain insights or make predictions. This can include statistical analysis, data mining, machine learning algorithms, or artificial intelligence models.

Visualization and Reporting: The processed data and analytics results can be visualized using charts, graphs, or dashboards to provide a clear representation of the information. Visualizations aid in understanding patterns, trends, or anomalies in the data. Additionally, reports or alerts can be generated to notify users or stakeholders of important findings or events.

Real-Time Processing: IoT systems often require real-time processing to enable timely decision-making or immediate actions based on the collected data. Real-time processing involves analyzing data as it arrives and generating responses or triggers in near real-time.

Feedback Loop: The insights or actions derived from the processed data can be used to provide feedback and optimize the IoT system's performance. For example, adjusting sensor thresholds, improving predictive models, or triggering automated responses based on the analysis results.

Overall, feature extraction and data processing using IoT play a crucial role in transforming raw data collected from IoT devices into meaningful information and actionable insights for various applications such as smart homes, industrial monitoring, healthcare, or environmental monitoring.

Algorithm: Combined Feature Selection, Evaluation, and Attribute Selection

1. Load Data:

- Load the dataset from a given file path.

2. Evaluate Feature Selection Methods:

- For each chosen feature selection method (like CfsSubset Eval, GainRatio Attribute Eval, etc.):
- Apply the method to the dataset to find the most important attributes.
- Store the list of selected attributes for each method.

3. Combine Selected Attributes:

- Create an empty list to hold combined selected attributes.
- For each list of selected attributes from different methods:
 - Add the attributes to the combined list, avoiding duplicates.

4. Apply Feature Selection on Combined Data:

- Use the CfsSubsetEval method on the dataset with the combined selected attributes.
- Create a new dataset with the chosen attributes.

5. Train Classifier:

- Choose a classifier, like Linear Regression.
- Train the classifier using the data obtained after combined feature selection.

6. Rank and Percentile Calculation:

- Calculate the attribute performance scores based on the classifier's coefficients.
- Rank the attributes based on their performance scores.
- Calculate the percentile of each attribute's performance score.

7. Select Most Appropriate Attributes:

- Choose a threshold percentile for attribute selection (e.g., 80%).
- Select attributes that have performance percentiles above the threshold.

8. Display Results:

- Display the matrix of selected attributes from each method.
- Display the list of combined attributes.
- Display the coefficients of the trained classifier.
- Display the ranked attributes along with their performance scores and percentiles.
- Display the final selected attributes based on the threshold.

9. End.

Machine learning algorithms are the core components of machine learning systems. They allow computers to learn from data and make predictions and decisions without being explicitly programmed for the task. Overall, the model offers an advanced approach to attribute selection, enabling the creation of more accurate and interpretable classification models across various domains and applications. The various Algorithms for extracting useful information are discussed in the following paragraphs.

4.2 Feature Selection Methods

Feature selection methods are essential techniques in data analysis and machine learning that streamline datasets by identifying the most pertinent attributes. Among these methods, CfsSubsetEval evaluates attribute relevance while considering inter-

feature relationships, Gain Ratio AttributeEval assesses attributes' class discrimination ability, One R creates simple rules to predict classes, Relief F handles noisy data by evaluating attribute influence on nearest neighbors' classification, and Symmetrical Uncertainty calculates mutual information between attributes and classes while accounting for class imbalance. Employing these methods enhances data quality, improves model performance, and ensures more effective feature selection, leading to more accurate and interpretable results. When choosing a feature selection method, it is important to consider the type of data, the characteristics of the problem, and the specific requirements of the machine learning task. It is often useful to try multiple techniques and compare their effects on model performance to determine the most effective approach for a particular scenario. The goal is to improve model performance, reduce computational complexity, and reduce the risk of overfitting. 21 attributes used for Analysis are shown in the Table 4.1.

Table 4.1: Attribute Names and ID

Attribute ID	Attribute Name
1	TIME
2	REGION_ID
3	SPEED
4	REGION
5	BUS_COUNT
6	NUM_READS
7	HOUR
8	DAY_OF_WEEK
9	MONTH
10	DESCRIPTION
11	RECORD_ID
12	WEST
13	EAST
14	SOUTH
15	NORTH
16	NW_LOCATION
17	SE_LOCATION
18	COMMUNITY AREAS
19	ZIP CODES
20	WARDS
21	CLASS LABEL

4.2.1 Info Gain Attribute Eval

Information gain is a metric used in the context of decision tree-based algorithms, specifically for feature selection. This helps determine the relevance of features in classifying or predicting the target variable. Attribute evaluation using information gains is commonly used in decision tree algorithms. Information gain measures the effectiveness of an attribute in classifying a dataset. It is based on the concept of entropy, which quantifies the uncertainty or disorder in a dataset. The information gain of an attribute is calculated by comparing the entropy of the dataset before and after partitioning based on that attribute.

In Weka, the 'InfoGainAttributeEval' feature selection method is utilized to assess attribute importance within a dataset. This technique quantifies the value of attributes by measuring the reduction in uncertainty they bring to the classification or regression task, particularly in decision tree algorithms. By comparing the entropy before and after splitting the data based on an attribute, the 'InfoGainAttributeEval' method determines the most informative attributes, aiding in improving model performance by focusing on key features. This approach is an essential tool in Weka's arsenal for enhancing data preprocessing and model building processes.

Evaluator: `weka.attributeSelection.InfoGainAttributeEval`

Search: `weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1`

Relation: `Chicago_Traffic_1000`

Instances: 1000

Attributes: 21

Evaluation mode: Evaluate on all training data

Search Method: Attribute ranking.

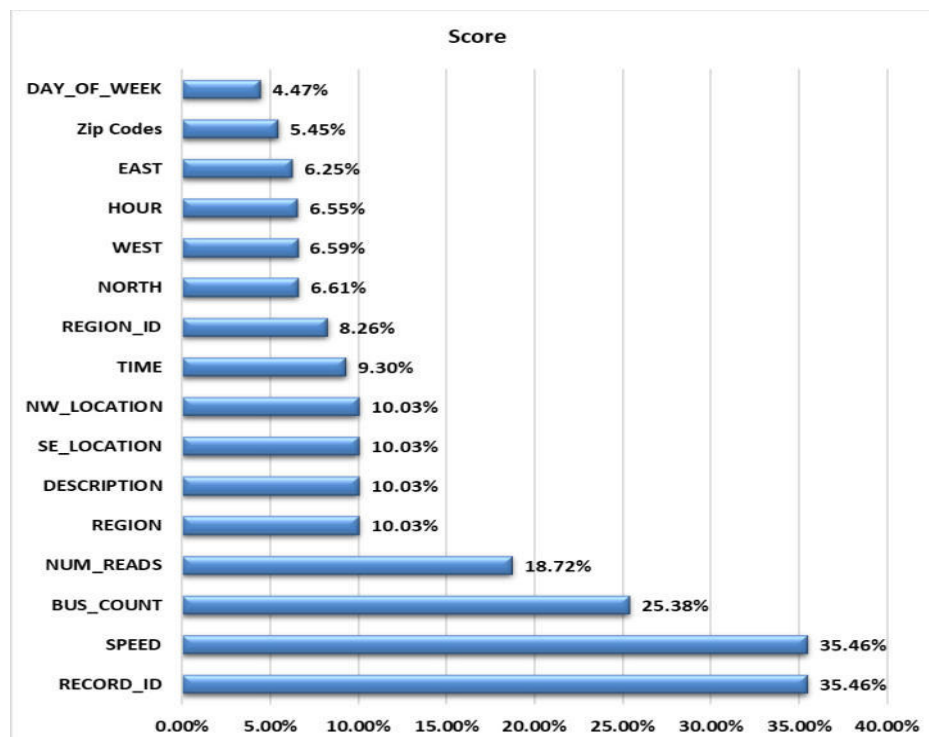
Attribute Evaluator (supervised, Class (nominal)): Information Gain Ranking Filter.

Attribute Scores: Shown in the Table 4.2

Table 4.2 : Attribute Score for Info Gain Attribute Eval

Score	Attribute/ Feature	Attribute ID
0.3546	RECORD_ID	11
0.3546	SPEED	3
0.2538	BUS_COUNT	5
0.1872	NUM_READS	6
0.1003	REGION	4
0.1003	DESCRIPTION	10
0.1003	SE_LOCATION	17
0.1003	NW_LOCATION	16
0.093	TIME	1
0.0826	REGION_ID	2
0.0661	NORTH	15
0.0659	WEST	12
0.0655	HOUR	7
0.0625	EAST	13
0.0545	ZIP CODES	19
0.0447	DAY_OF_WEEK	8

Figure 4.1: Attribute Score for Info Gain Attribute Eval



The results confirm that the top five high scoring attributes are RECORD_ID, SPEED, BUS_COUNT, NUM_READS and REGION with values 0.3546, 0.3546, 0.2538, 0.1872 and 0.1003 respectively. The low scoring attributes are TIME, REGION_ID, NORTH, WEST, HOUR, EAST, ZIP CODES and DAY_OF_WEEK with values 0.093, 0.0826, 0.0661, 0.0659, 0.0655, 0.0625, 0.0545 and 0.0447 respectively. Based on the figure above the 16 selected attributes are as shown below:

Selected attributes: 11,3,5,6,4,10,17,16,1,2,15,12,7,13,19 and 8.

Total No. of selected attributes: 16

4.2.2 Correlation Attribute Eval

Correlation Attribute Eval method serves as a feature selection technique aimed at evaluating attribute significance within a dataset by gauging their correlation with the class variable. By calculating the correlation between each attribute and the class labels, this approach helps identify attributes that bear the most relevance to the classification task. This process aids in refining model performance by retaining attributes that demonstrate strong connections to the class variable and discarding those with weaker correlations.

Evaluator: weka.attributeSelection.CorrelationAttributeEval

Search: weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N 16

Relation: Chicago_Traffic_1000

Instances: 1000

Attributes: 21

Evaluation mode: Evaluate on all training data

Search Method: Attribute ranking.

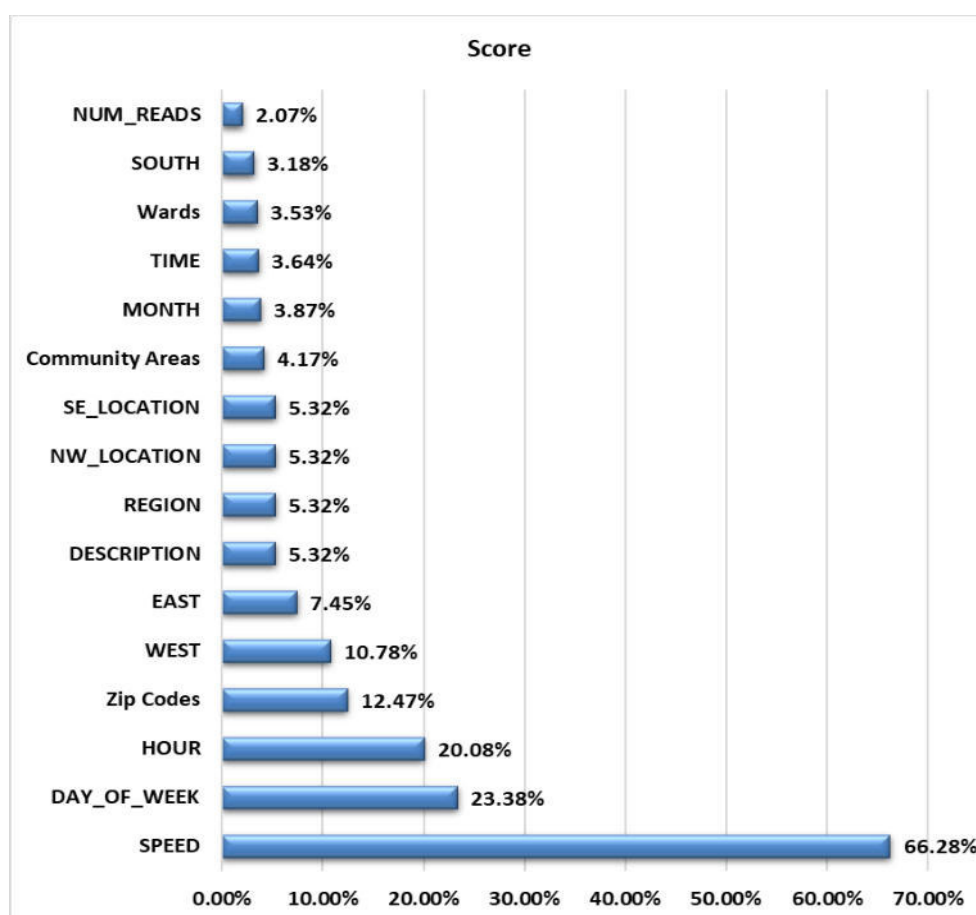
Attribute Evaluator (supervised, Class (nominal)): Correlation Ranking Filter.

Attribute Scores: Shown in Table 4.3

Table 4.3 : Attribute Score for Correlation Attribute Eval

Score	Attribute/ Feature	Attribute ID
0.6628	SPEED	3
0.2338	DAY_OF_WEEK	8
0.2008	HOUR	7
0.1247	ZIP CODES	19
0.1078	WEST	12
0.0745	EAST	13
0.0532	DESCRIPTION	10
0.0532	REGION	4
0.0532	NW_LOCATION	16
0.0532	SE_LOCATION	17
0.0417	COMMUNITY AREAS	18
0.0387	MONTH	9
0.0364	TIME	1
0.0353	WARDS	20
0.0318	SOUTH	14
0.0207	NUM_READS	6

Figure 4.2: Attribute Score for Correlation Attribute Eval



The outcome confirm that the top five high scoring attributes are SPEED, DAY_OF_WEEK, HOUR, ZIP CODES and WEST with values 0.6628, 0.2338, 0.2008, 0.1247, 0.1078 respectively. The low scoring attributes are EAST, DESCRIPTION, REGION, NW_LOCATION, SE_LOCATION, COMMUNITY AREAS, MONTH, TIME, WARDS, SOUTH and NUM_READS with values 0.0745, 0.0532, 0.0532, 0.0532, 0.0532, 0.0417, 0.0387, 0.0364, 0.0353, 0.0318 and 0.0207 respectively. Based on the figure above the 16 selected attributes are as shown below:

Selected attributes: 3,8,7,19,12,13,10,4,16,17,18,9,1,20,14 and 6.

Total No. of selected attributes: 16

4.2.3 Classifier Attribute Eval

Attribute selection algorithms are applied before applying data mining tasks such as classification, clustering, and outlier analysis. Classifier Attribute Eval is a feature selection method used to evaluate the importance of attributes (features) with respect to a classifier's performance. This technique helps to identify and select the most relevant attributes for building a predictive model.

Evaluator: weka.attributeSelection

Search: weka.attributeSelection

Relation: Chicago_Traffic_1000

Instances: 1000

Attributes: 21

Evaluation mode: Evaluate on all training data

Search Method: Attribute ranking.

Attribute Evaluator (supervised, Class (nominal)): Classifier feature evaluator

Attribute Scores: Shown in Table 4.4

Table 4.4 : Attribute Score for Classifier Attribute Eval

Rank	Attribute/ Feature	Attribute ID
1	WARDS	20
2	HOUR	7
3	DAY_OF_WEEK	8
4	ZIP CODES	19
5	NUM_READS	6
6	BUS_COUNT	5
7	REGION	4
8	SPEED	3
9	REGION_ID	2
10	MONTH	9
11	DESCRIPTION	10
12	RECORD_ID	11
13	NW_LOCATION	16
14	COMMUNITY AREAS	18
15	SE_LOCATION	17
16	NORTH	15

The results confirm that the top five high scoring attributes or ranked attributes are Wards, HOUR, DAY_OF_WEEK, Zip Codes and NUM_READS with ranks 1, 2, 3, 4, and 5 respectively. The low ranked attributes are BUS_COUNT, REGION, SPEED, REGION_ID, MONTH, DESCRIPTION, RECORD_ID, NW_LOCATION, COMMUNITY AREAS, SE_LOCATION and NORTH with ranks 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 and 16 respectively. Based on the table above the 16 selected attributes are as shown below:

Selected attributes: 20,7,8,19,6,5,4,3,2,9,10,11,16,18,17 and 15

Total No. of selected attributes: 16

4.2.4 Cfs Subset Eval

Correlation-based Feature Selection Subset Evaluator it is feature selection method that evaluates the relevance and redundancy of attributes within a dataset based on their correlation with the class variable. The goal is to select a subset of attributes that are highly correlated with the class variable while minimizing redundancy among them.

Evaluator: weka.attributeSelection.CfsSubsetEval

Search: weka.attributeSelection.GreedyStepwise

Relation: Chicago_Traffic_1000

Instances: 1000

Attributes: 21

Evaluation mode: Evaluate on all training data

Search Method: Greedy Stepwise (forwards).

Attribute Subset Evaluator (supervised, Class (nominal)): CFS Subset Evaluator

The results confirm that the top three high scoring attributes are SPEED, NUM_READS and MONTH, DAY_OF_WEEK, Zip Codes and NUM_READS with ranks 1, 2 and 3 respectively. The following 3 selected attributes are as follows:

Selected attributes: 3,6 and 19 (SPEED, NUM_READS and Zip Codes)

Total No. of selected attributes: 3

4.2.5 Gain Ratio Attribute Eval

Gain Ratio Attribute Eval is one of the method used for evaluating the attributes and finding the most appropriate one using the ability to discriminate between various classes. This method takes the concept of gain ratio basically which take into consideration the intrinsic information of the attribute and also the potential information gain.

Evaluator: weka.attributeSelection.GainRatioAttributeEval

Search: weka.attributeSelection.

Relation: Chicago_Traffic_1000

Instances: 1000

Attributes: 21

Evaluation mode: Evaluate on all training data

Search Method: Attribute ranking.

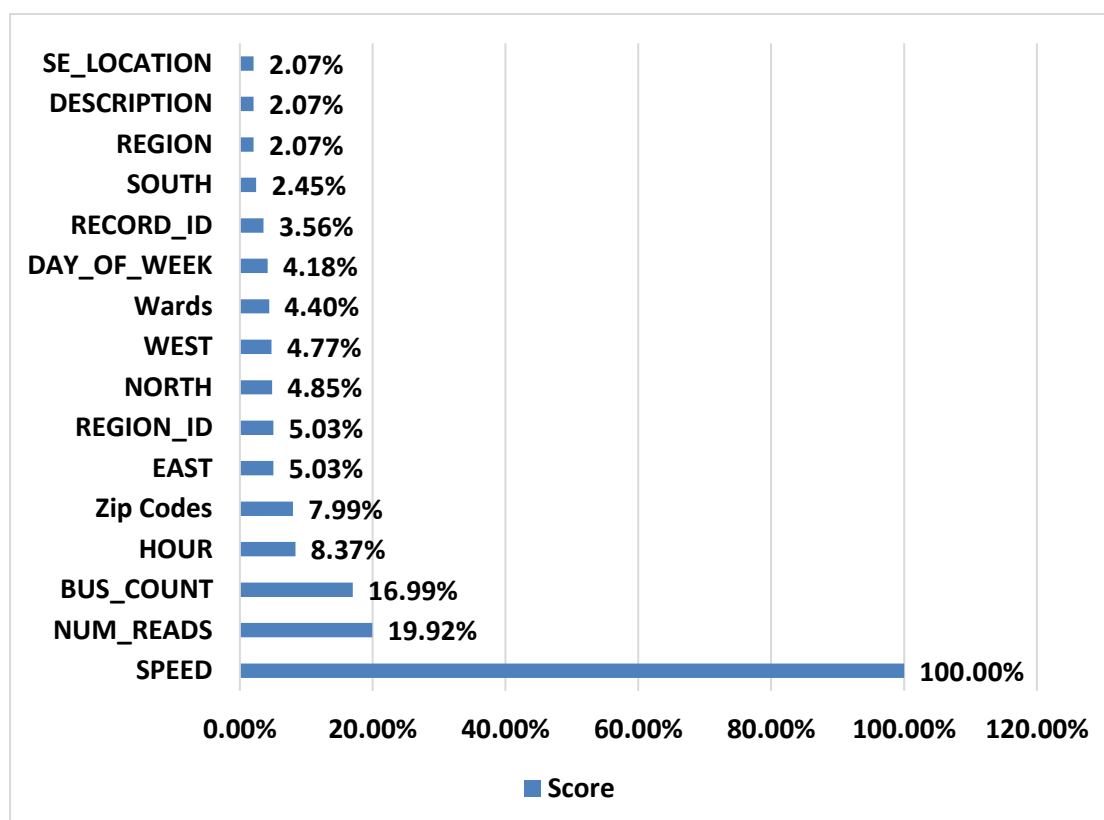
Attribute Evaluator (supervised, Class (nominal)): Gain Ratio feature evaluator

Attribute Scores: Shown in Table 4.5

Table 4.5: Attribute Score for Gain Ratio Attribute Eval

Score	Attribute/ Feature	Attribute ID
1	SPEED	3
0.1992	NUM_READS	6
0.1699	BUS_COUNT	5
0.0837	HOUR	7
0.0799	ZIP CODES	19
0.0503	EAST	13
0.0503	REGION_ID	2
0.0485	NORTH	15
0.0477	WEST	12
0.044	WARDS	20
0.0418	DAY_OF_WEEK	8
0.0356	RECORD_ID	11
0.0245	SOUTH	14
0.0207	REGION	4
0.0207	DESCRIPTION	10
0.0207	SE_LOCATION	17

Figure 4.3: Attribute Score for Gain Ratio Attribute Eval



The outcome confirm that the top five high scoring attributes are SPEED, NUM_READS, BUS_COUNT, HOUR and ZIP CODES with values 1, 0.1992, 0.1699, 0.0837 and 0.0799 respectively. The low scoring attributes are EAST, REGION_ID,

NORTH, WEST, Wards, DAY_OF_WEEK, RECORD_ID, SOUTH, REGION, DESCRIPTION and SE_LOCATION with values 0.0503, 0.0503, 0.0485, 0.0477, 0.044, 0.0418, 0.0356, 0.0245, 0.0207, 0.0207 and 0.0207 respectively. Based on the figure above the 16 selected attributes are as shown below:

Selected attributes: 3,6,5,7,19,13,2,15,12,20,8,11,14,4,10 and 17

Total No. of selected attributes: 16

4.2.6 OneR Attribute Eval

OneR Attribute Eval is a feature selection method that's based on the One Rule (OneR) classifier. The OneR algorithm is a simple and interpretable rule-based classification algorithm that selects a single attribute as the best predictor for classifying instances. The OneR Attribute Eval method evaluates the quality of attributes by measuring how well they serve as rules for classifying instances. OneR or "One Rule" is a simple, interpretable classification algorithm that is often used as a fast base model or as a benchmark for more complex algorithms. Attribute evaluation in the context of OneR refers to the process of selecting the best attributes to create classification rules. The goal is to find the attribute that by itself provides the most accurate prediction. For each attribute value, this algorithm creates a simple rule based on that value. Created rule is used to count the number of correct and incorrect classifications for calculating total error. The attribute with lowest error is selected as the One Rule. Finally classification rule is created on chosen attribute.

Evaluator: weka.AttributeSelection.OneRAttributeEval.

Search: weka. Attribute Selection.

Relation: Chicago_Traffic_1000

Instances: 1000

Attributes: 21

Evaluation mode: Evaluate on all training data

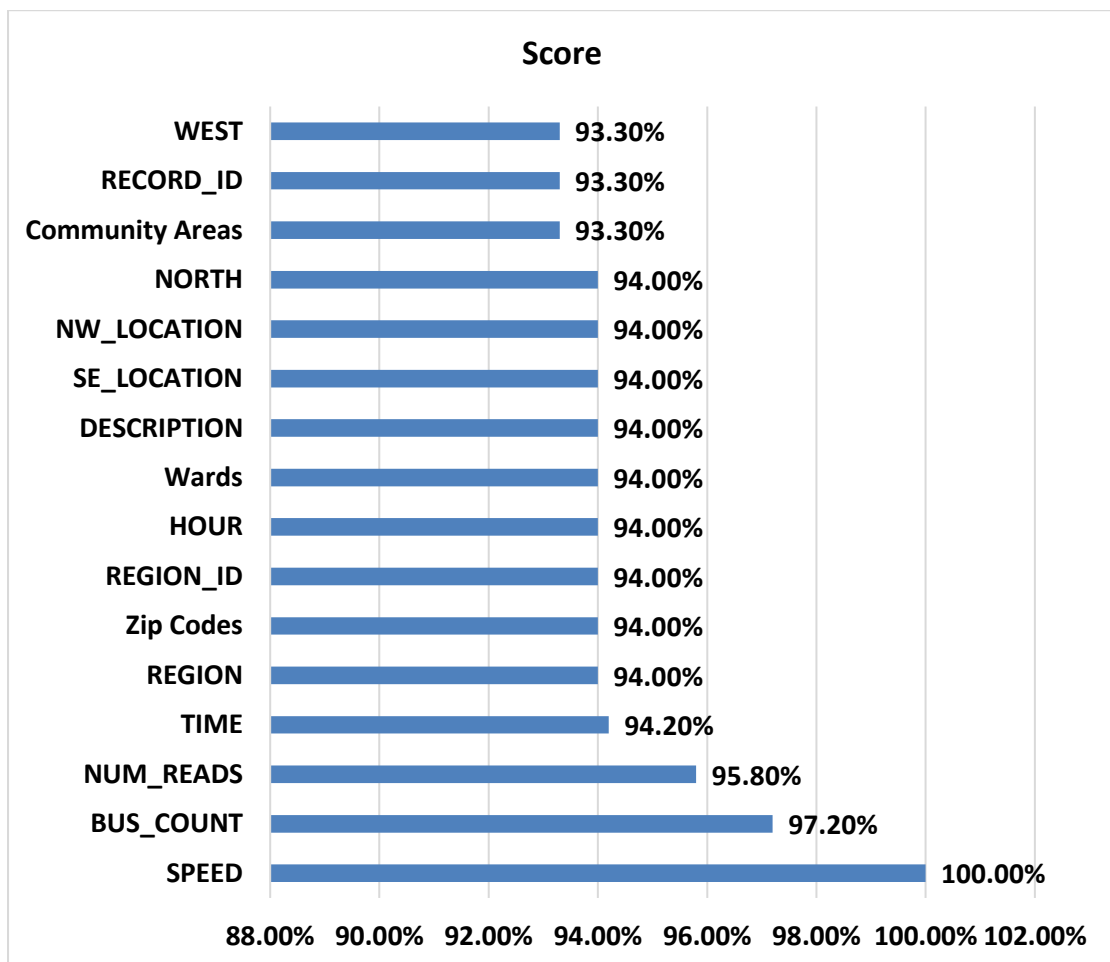
Search Method: Attribute ranking.

Attribute Evaluator (supervised, Class (nominal)): OneR feature evaluator.

Table 4.6: Attribute Score for OneR Attribute Eval

Score	Attribute/ Feature	Attribute ID
1.00	SPEED	3
0.972	BUS_COUNT	5
0.958	NUM_READS	6
0.942	TIME	1
0.94	REGION	4
0.94	ZIP CODES	19
0.94	REGION_ID	2
0.94	HOUR	7
0.94	WARDS	20
0.94	DESCRIPTION	10
0.94	SE_LOCATION	17
0.94	NW_LOCATION	16
0.94	NORTH	15
0.933	COMMUNITY AREAS	18
0.933	RECORD_ID	11
0.933	WEST	12

Figure 4.4: Attribute Score for OneR Attribute Eval



The results confirm that the top five high scoring attributes are SPEED, BUS_COUNT, NUM_READS, TIME and REGION with values 1.00, 0.972, 0.958, 0.942 and 0.94 respectively. The low scoring attributes are ZIP CODES, REGION_ID, HOUR, WARDS, DESCRIPTION, SE_LOCATION, NW_LOCATION, NORTH, COMMUNITY AREAS, RECORD_ID and WEST with values 0.94, 0.94, 0.94, 0.94, 0.94, 0.94, 0.94, 0.94, 0.933, 0.933 and 0.933 respectively. Based on the figure above the 16 selected attributes are as shown below:

Selected attributes: 3,5,6,1,4,19,2,7,20,10,17,16,15,18,11 and 12

Total No. of selected attributes: 16

4.2.7 ReliefF Attribute Eval

ReliefF Attribute Eval is a feature selection method based on the ReliefF algorithm. The ReliefF algorithm is designed to assess the importance of attributes in a dataset for classification tasks, particularly in the context of feature selection and ranking. It focuses on measuring the relevance and quality of attributes by considering the nearest neighbors of instances.

Evaluator: weka.attributeSelection.ReliefFAttributeEval

Search: weka.attributeSelection.

Relation: Chicago_Traffic_1000

Instances: 1000

Attributes: 21

Evaluation mode: Evaluate on all training data

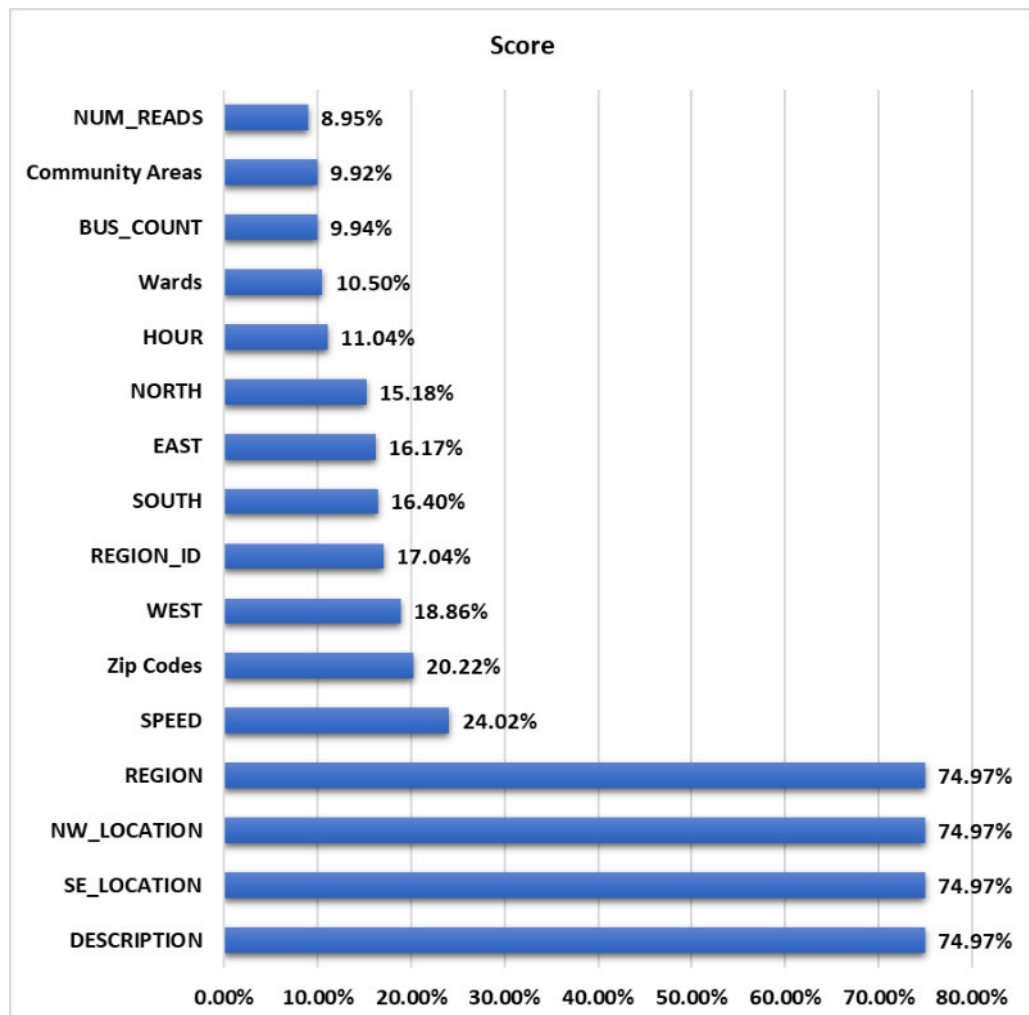
Search Method: Attribute ranking.

Attribute Evaluator (supervised, Class (nominal)): ReliefF Ranking Filter

Table 4.7: Attribute Score for ReliefF Attribute Eval

Score	Attribute/ Feature	Attribute ID
0.7497	DESCRIPTION	10
0.7497	SE_LOCATION	17
0.7497	NW_LOCATION	16
0.7497	REGION	4
0.2402	SPEED	3
0.2022	ZIP CODES	19
0.1886	WEST	12
0.1704	REGION_ID	2
0.164	SOUTH	14
0.1617	EAST	13
0.1518	NORTH	15
0.1104	HOUR	7
0.105	WARDS	20
0.0994	BUS_COUNT	5
0.0992	COMMUNITY AREAS	18
0.0895	NUM_READS	6

Figure 4.5: Attribute Score for ReliefF Attribute Eval



The outcome confirm that the top five high scoring attributes are DESCRIPTION, SE_LOCATION, NW_LOCATION, REGION and SPEED with values 0.7497, 0.7497, 0.7497, 0.7497 and 0.2402 respectively. The low scoring attributes are ZIP CODES, WEST, REGION_ID, SOUTH, EAST, NORTH, HOUR, WARDS, BUS_COUNT, COMMUNITY AREAS and NUM_READS with values 0.2022, 0.1886, 0.1704, 0.164, 0.1617, 0.1518, 0.1104, 0.105, 0.0994, 0.0992 and 0.0895 respectively. Based on the figure above the 16 selected attributes are as shown below:

Selected attributes: 10,17,16,4,3,19,12,2,14,13,15,7,20,5,18 and 6

Total No. of selected attributes: 16

4.2.8 Symmetrical Uncert Attribute Eval

Symmetrical Uncert Attribute Eval Is one of the selection methods which follows the symmetric uncertainty principle. The symmetric uncertainty is basically a measure that finds the amount of information which is being shared between the two variables and that is two being very useful for finding the relevance of various attributes in a classification context. This approach mainly identifies the attributes or features which have a strong relationship with the class label variable While taking in consideration the potential interactions.

Evaluator: weka.attributeSelection.SymmetricalUncertAttributeEval

Search: weka.attributeSelection.

Relation: Chicago_Traffic_1000

Instances: 1000

Attributes: 21

Evaluation mode: Evaluate on all training data

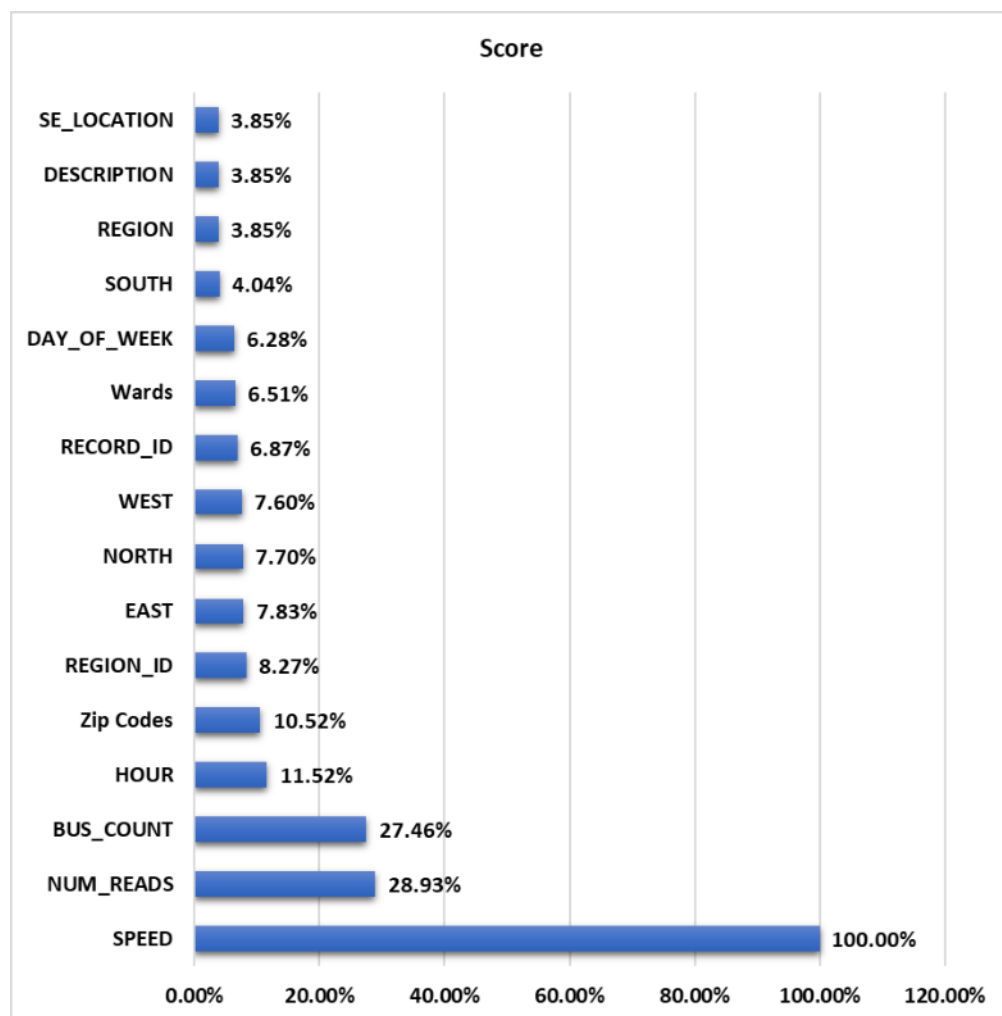
Search Method: Attribute ranking.

Attribute Evaluator (supervised, Class (nominal)): Symmetrical Uncertainty Ranking Filter

Table 4.8: Attribute Score for Symmetrical Uncert Attribute Eval

Score	Attribute/ Feature	Attribute ID
1	SPEED	3
0.2893	NUM_READS	6
0.2746	BUS_COUNT	5
0.1152	HOUR	7
0.1052	ZIP CODES	19
0.0827	REGION_ID	2
0.0783	EAST	13
0.077	NORTH	15
0.076	WEST	12
0.0687	RECORD_ID	11
0.0651	WARDS	20
0.0628	DAY_OF_WEEK	8
0.0404	SOUTH	14
0.0385	REGION	4
0.0385	DESCRIPTION	10
0.0385	SE_LOCATION	17

Figure 4.6: Attribute Score for Symmetrical Uncert Attribute Eval



The result confirm that the top five high scoring attributes are SPEED, NUM_READS, BUS_COUNT, HOUR and ZIP CODES with values 1, 0.2893, 0.2746, 0.1152 and 0.1052 respectively. The low scoring attributes are REGION_ID, EAST, NORTH, WEST, RECORD_ID, WARDS, DAY_OF_WEEK, SOUTH, REGION, DESCRIPTION and SE_LOCATION with values 0.0827, 0.0783, 0.077, 0.076, 0.0687, 0.0651, 0.0628, 0.0404, 0.0385, 0.0385 and 0.0385 respectively. Based on the figure above the 16 selected attributes are as shown below:

Selected attributes: 10,17,16,4,3,19,12,2,14,13,15,7,20,5,18 and 6

Total No. of selected attributes: 16

4.3 Feature Extraction Using Multiple Regression

The scope of multiple regression can be expanded from finding the relationship between dependent and independent variables to getting insights for finding the appropriate attributes that is for doing feature selection.

Table 4.9: Variables Entered / Removed

Variables Entered/Removed ^a			
Model	Variables Entered	Variables Removed	Method
1	Wards, Hour, North, Month, Community Areas, Speed, Day_Of_Week, West, Num_Reads, East, Bus_Count, South ^b	.	Enter
a. Dependent Variable: Class Label			
b. All requested variables entered.			

Table 4.10: Model Summary

Model Summary				
Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.675 ^a	.455	.445	.182
a. Predictors: (Constant), Wards, HOUR, NORTH, MONTH, Community Areas, SPEED, DAY_OF_WEEK, WEST, NUM_READS, EAST, BUS_COUNT, SOUTH				

Table 4.11: ANOVA Summary

ANOVA ^a						
Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	18.323	12	1.527	46.262	.000 ^b
	Residual	21.949	665	.033		
	Total	40.273	677			
a. Dependent Variable: Class Label						
b. Predictors: (Constant), Wards, HOUR, NORTH, MONTH, Community Areas, SPEED, DAY_OF_WEEK, WEST, NUM_READS, EAST, BUS_COUNT, SOUTH						

Table 4.12: Coefficients Summary

Coefficients ^a						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	27.656	12.608		2.194	.029
	Speed	-.023	.001	-.561	-15.761	.000
	Bus_Count	.003	.001	.276	1.828	.068
	Num_Reads	.000	.000	-.179	-1.167	.244
	Hour	-.010	.002	-.212	-5.860	.000
	Day_Of_Week	.048	.012	.143	4.159	.000
	Month	-.040	.012	-.102	-3.223	.001
	West	1.955	.442	.502	4.419	.000
	East	-1.953	.536	-.461	-3.643	.000
	South	2.759	.766	.923	3.604	.000
	North	-3.396	.821	-1.072	-4.134	.000
	Community Areas	.000	.000	-.023	-.759	.448
	Wards	.000	.001	-.011	-.354	.724
a. Dependent Variable: Class Label						

The provided regression model summary indicates the relationships between predictor variables (Speed, Bus_Count, Num_Reads, Hour, Day_Of_Week, Month, West, East, South, North, Community Areas, Wards) and a dependent variable ("Class Label"). The coefficients show how changes in predictor variables affect the predicted outcome. Notably, variables such as "Speed" and "Hour" have moderate negative impacts, while "Day_Of_Week" and "West" have moderate positive impacts, all with statistical significance. "East" and "North" have strong negative impacts, while "South" has a

strong positive impact, all statistically significant. Other variables like "Bus_Count," "Num_Reads," "Month," "Community Areas," and "Wards" have smaller effects with varying statistical significance. These interpretations aid in understanding the importance and direction of influence of each predictor on the dependent variable. So indirectly the identified selected attributes are as follows:

Selected attributes: 3,5,6,7,8,9,12,13,14,15,18 and 20

Total No. of selected attributes: 12

4.4 Combined Feature Selection Matrix

The proposed combined feature selection matrix basically is an arrangement between types of feature selection method and the identified attributes after analysis. The Attribute names and their ID's are given below for the reference.

Table 4.13: Attribute Names and ID

Attribute ID	Attribute Name
1	TIME
2	REGION_ID
3	SPEED
4	REGION
5	BUS_COUNT
6	NUM_READS
7	HOUR
8	DAY_OF_WEEK
9	MONTH
10	DESCRIPTION
11	RECORD_ID
12	WEST
13	EAST
14	SOUTH
15	NORTH
16	NW_LOCATION
17	SE_LOCATION
18	COMMUNITY AREAS
19	ZIP CODES
20	WARDS
21	CLASS LABEL

Table 4.14: Combined Feature Selection Matrix

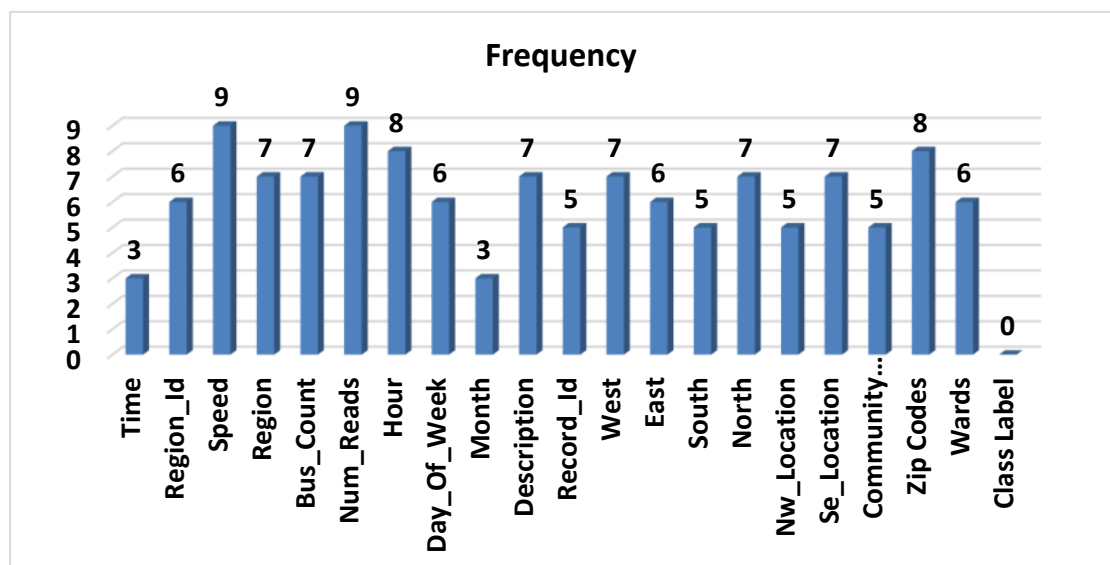
Info Gain Attribute Eval	Correlation Attribute Eval	Classifier Attribute Eval	Cfs Subset Eval	Gain Ratio Attribute Eval	OneR Attribute Eval	ReliefF Attribute Eval	Symmetrical Uncert Attribute Eval	Multiple Regression
11	3	20	3	3	3	10	3	3
3	8	7	6	6	5	17	6	5
5	7	8	19	5	6	16	5	6
6	19	19	-	7	1	4	7	7
4	12	6	-	19	4	3	19	8
10	13	5	-	13	19	19	2	9
17	10	4	-	2	2	12	13	12
16	4	3	-	15	7	2	15	13
1	16	2	-	12	20	14	12	14
2	17	9	-	20	10	13	11	15
15	18	10	-	8	17	15	20	18
12	9	11	-	11	16	7	8	20
7	1	16	-	14	15	20	14	-
13	20	18	-	4	18	5	4	-
19	14	17	-	10	11	18	10	-
8	6	15	-	17	12	6	17	-

The above table shows the 9 feature selection methods and various attributes being selected by them which are basically shown through the attribute ID in Table 4.14

Table 4.15: Attribute and Count

Attribute ID	Attribute Name	Count
1	TIME	3
2	REGION_ID	6
3	SPEED	9
4	REGION	7
5	BUS_COUNT	7
6	NUM_READS	9
7	HOUR	8
8	DAY_OF_WEEK	6
9	MONTH	3
10	DESCRIPTION	7
11	RECORD_ID	5
12	WEST	7
13	EAST	6
14	SOUTH	5
15	NORTH	7
16	NW_LOCATION	5
17	SE_LOCATION	7
18	COMMUNITY AREAS	5
19	ZIP CODES	8
20	WARDS	6
21	CLASS LABEL	0

Figure 4.7: Attribute and Frequency



The number of times the frequency of the particular attribute is identified from the combined feature selection matrix. From the figure above it is clear that attributes or features Speed and Num_Reads is having the highest frequency or occurrence with value 9 followed by attributes Hour, Zip_Codes with value 8 whereas the number of occurrences of attribute Class_Label, Month and Time is lowest.

Table 4.16: Overall Attribute Performance in Percentage (%)

Applied Attribute Evaluator ID	Attribute Name	Count	Overall Attribute Performance in Percentage (%)
1	TIME	3	14.29
2	REGION_ID	6	28.57
3	SPEED	9	42.86
4	REGION	7	33.33
5	BUS_COUNT	7	33.33
6	NUM_READS	9	42.86
7	HOURL	8	38.10
8	DAY_OF_WEEK	6	28.57
9	MONTH	3	14.29
10	DESCRIPTION	7	33.33
11	RECORD_ID	5	23.81
12	WEST	7	33.33
13	EAST	6	28.57
14	SOUTH	5	23.81
15	NORTH	7	33.33
16	NW_LOCATION	5	23.81
17	SE_LOCATION	7	33.33
18	COMMUNITY AREAS	5	23.81
19	ZIP CODES	8	38.10
20	WARDS	6	28.57
21	CLASS LABEL	0	0.00

The overall attribute contribution is being evaluated in percentage as shown above in the table. The performance percentages demonstrate the assessed significance of each attribute, ranging from 0% to 42.86%. Attributes like "Speed," "Num_Reads," "Hour," and "Zip Codes" received relatively higher performance percentages, implying they have notable influence or relevance in the context of the evaluation.

4.5 Rank and Percentile

Rank and percentile approach is one of the valuable techniques for the feature selection which also supports the identification of the most relevant attributes for the data analysis and modelling.

Table 4.17: Rank & Percentile Approach Results

Attribute Name	Attribute ID /Point	Overall Attribute Performance in Percentage (%)	Rank	Percentile
SPEED	3	42.86	1	95.00%
NUM_READS	6	42.86	1	95.00%
HOURL	7	38.10	3	85.00%
ZIP CODES	19	38.10	3	85.00%
REGION	4	33.33	5	55.00%
BUS_COUNT	5	33.33	5	55.00%
DESCRIPTION	10	33.33	5	55.00%
WEST	12	33.33	5	55.00%
NORTH	15	33.33	5	55.00%
SE_LOCATION	17	33.33	5	55.00%
REGION_ID	2	28.57	11	35.00%
DAY_OF_WEEK	8	28.57	11	35.00%
EAST	13	28.57	11	35.00%
WARDS	20	28.57	11	35.00%
RECORD_ID	11	23.81	15	15.00%
SOUTH	14	23.81	15	15.00%
NW_LOCATION	16	23.81	15	15.00%
COMMUNITY AREAS	18	23.81	15	15.00%
TIME	1	14.29	19	5.00%
MONTH	9	14.29	19	5.00%
CLASS LABEL	21	0.00	21	0.00%

The rank and percentile method were being use for further analysis of the various attributes using the overall attribute performance. Accordingly, the table above shows the evaluated rank and percentile. Evaluation seems to gauge the significance of each attribute in the context of the analysis. The "Rank" column indicates the attribute's rank based on performance, while the "Percent" column indicates the percentile of its performance among all attributes. Attributes like "Speed" and "Num_Reads" achieved the highest overall performance of 42.86%, securing the top rank and a percentile of 95.00%. "Hour" and "Zip Codes" follow closely with 38.10% performance and a joint

rank of 3, corresponding to an 85.00% percentile. Attributes such as "Region," "Bus_Count," "Description," "West," "North," and "Se_Location" share a performance of 33.33%, ranking 5th with a 55.00% percentile. Some attributes, including "Time," "Month," and "Class Label," had lower performance percentages, ranking lower with 14.29% and 0.00% performance, respectively. This ranking and percentile analysis offers insights into the relative importance of attributes within the dataset, helping to identify attributes that strongly contribute to the analysis and those with lesser impact.

4.6 Summary

Eight feature selection methods within WEKA machine learning software were executed to select and extract highly correlated features from Chicago_Traffic_1000 dataset with twenty one features. First six features along with class label were short listed after removing redundant and uncorrelated features for traffic congestion prediction model development. The selected features listed below.

1. SPEED
2. NUM_READS
3. HOUR
4. ZIP CODES
5. REGION
6. BUS_COUNT
7. CLASS LABEL

Chapter – 5

Utilization of AI and ML Prediction Algorithms

- 5.1 Traffic Control Systems for Smart Cities
 - 5.1.1 IoT-Based Traffic Prediction Models
 - 5.1.2 Machine Learning-Based Traffic Prediction Models
- 5.2 Machine Learning Predictive Model for Smart Transportation
- 5.3 Analysis of Machine Learning Models for Smart Transportation
 - 5.3.1 Performance Measure
 - 5.3.2 Error Measures
 - 5.3.3 Cross-Validation Configuration Setting (10-folds) Results
 - A. Performance Measures
 - i. Accuracy Measures
 - ii. Confusion Matrix Parameters – Low Traffic
 - iii. Confusion Matrix Parameters – Heavy Traffic
 - B. Error Measure Results
 - C. Execution Time Results
 - 5.3.4 Cross-Validation Configuration Setting (25-folds) Results
 - A. Performance Measures
 - i. Accuracy Measures
 - ii. Confusion Matrix Parameters – Low Traffic
 - iii. Confusion Matrix Parameters – Heavy Traffic
 - B. Error Measure Results
 - C. Execution Time Results
 - 5.3.5 Cross-Validation Configuration Setting (30% Split) Results
 - A. Performance Measures
 - i. Accuracy Measures
 - ii. Confusion Matrix Parameters – Low Traffic
 - iii. Confusion Matrix Parameters – Heavy Traffic
 - B. Error Measure Results
 - C. Execution Time Results
 - 5.3.6 Consolidated Result
 - 5.3.7 Dominance Chart
 - 5.3.8 Weighted Sum Model Analysis Using Python
 - 5.3.9 Rank and Percentile Method
- 5.4 Hypothesis Testing Results
- 5.5 Summary

Artificial Intelligence and Machine Learning prediction algorithms are revolutionizing industries and domains by harnessing data to make precise forecasts and informed decisions. In smart cities it can be used to solve traffic congestion problems, traffic prediction and vehicle maintenance insights, In healthcare, they aid in disease diagnosis and drug discovery, In finance it benefits from risk assessment and stock market predictions, e-commerce relies on recommendation systems and demand forecasting, manufacturing optimizes operations with predictive maintenance and quality control, Energy sector uses it for forecasting consumption and renewable energy utilization, Agriculture improves crop yields and pest detection, Customer service employs chatbots and sentiment analysis, Weather forecasting becomes more accurate, Education adopts personalized learning and student success prediction, all contributing to enhanced efficiency, cost reduction, and better decision-making across various sectors.

5.1 Traffic Control Systems for Smart Cities

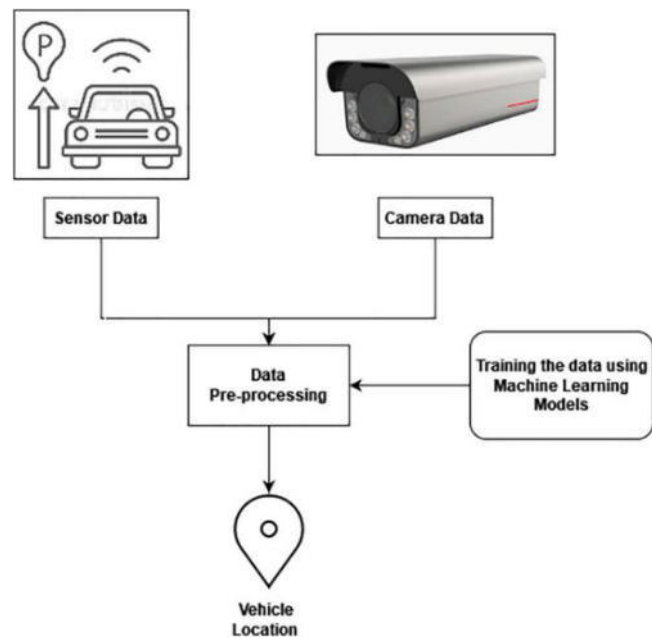
Traffic prediction and control systems in smart cities are essential for managing urban congestion and improving overall transportation efficiency. Various machine learning and IoT-based models have been developed to address these challenges. Some of the existing approaches in this field are:

Adaptive Traffic Signal Control: Using real-time traffic data collected from IoT sensors, adaptive traffic signal control systems can adjust signal timings based on current traffic conditions. These systems aim to minimize congestion and improve traffic flow efficiency.

Intelligent Transportation Systems (ITS): ITS integrates various technologies, including IoT, machine learning, and data analytics, to manage traffic in real-time. It involves strategies such as dynamic route guidance, incident detection, and congestion pricing to optimize traffic control in smart cities.

Predictive Traffic Control: By combining machine learning-based traffic prediction models with control algorithms, predictive traffic control systems can anticipate traffic conditions and adjust signal timings proactively. These systems help prevent congestion before it occurs.

Figure 5.1: Vehicle Location Tracking Using IoT and Machine Learning



The Figure 5.1 shown above explains the vehicle tracking system using IOT and Machine learning. Various IoT sensors, like Camera, GPS Probes, Motion Sensors etc. are used to collect raw traffic data, which is preprocessed for missing values and other non-linearities. Collected data is used for training the machine learning model, which will be used to forecast vehicle location on live data in future.

5.1.1 IoT-Based Traffic Prediction Models

Today traffic density is increasing in smart cities due to rise in population. This traffic rise results in time wastage, fuel wastage, environmental problems and casualties. Many solutions and methods are suggested in the past by many researchers but they lack in accuracy and reliability. IoT based traffic prediction models have shown us new ray of hope to overcome congestion problems in smart cities. Some of the IoT methods are described below.

a. Sensor Networks: IoT devices and sensors deployed across road networks can collect real-time data on traffic flow, vehicle speeds, and occupancy. By analyzing this data, traffic prediction models can provide accurate and up-to-date traffic forecasts.

b. **Vehicle-to-Infrastructure Communication:** IoT-enabled vehicles can communicate with smart infrastructure systems, such as traffic lights and road sensors, to gather real-time data. This information can be used to predict traffic patterns and optimize traffic control strategies.

These IoT models integrates data from different sources, including traffic cameras, GPS devices, and proximity sensors to predict and optimize vehicle density, traffic congestion, vehicle routing and improve the overall transportation experience.

5.1.2 Machine Learning-Based Traffic Prediction Models

In last one decade machine learning algorithms are increasingly becoming popular to solve traffic congestion problems due to static and error prone traditional statistical methods. Today's ever increasing city limits and population growth, has made city traffic management very difficult and demanding. Improvement in Machine learning technology is a new ray of hope for us. Some of the machine learning technologies used are described below.

a. **Time-Series Forecasting Models:** Models like ARIMA⁵⁸ and SARIMA⁵⁹ use historical traffic data to predict future traffic patterns. They consider factors like time of day, day of the week, and seasonality to forecast traffic conditions accurately.

b. **Artificial Neural Networks:** ANNs, such as MLP⁶⁰ and RNNs⁶¹ like Long Short-Term Memory, are capable of learning complex patterns in traffic data. These models can capture temporal dependencies and perform well in long-term traffic prediction.

c. **Support Vector Machines:** SVMs are used for both classification and regression tasks. They can be employed to predict traffic conditions based on historical data, considering features like weather, events, and road characteristics.

⁵⁸ Auto Regressive Integrated Moving Average

⁵⁹ Seasonal Auto Regressive Integrated Moving Average

⁶⁰ Multilayer Perceptron

⁶¹ Recurrent Neural Networks

d. Random Forests: Random Forest models combine multiple decision trees to make predictions. They can handle both numerical and categorical features, making them suitable for traffic prediction tasks involving multiple input variables.

In summary above set of algorithms have the capability to learn to perform tasks such as prediction and classification effectively using data. Learning is achieved using additional data and or additional models. A machine learning algorithm uses the following steps:

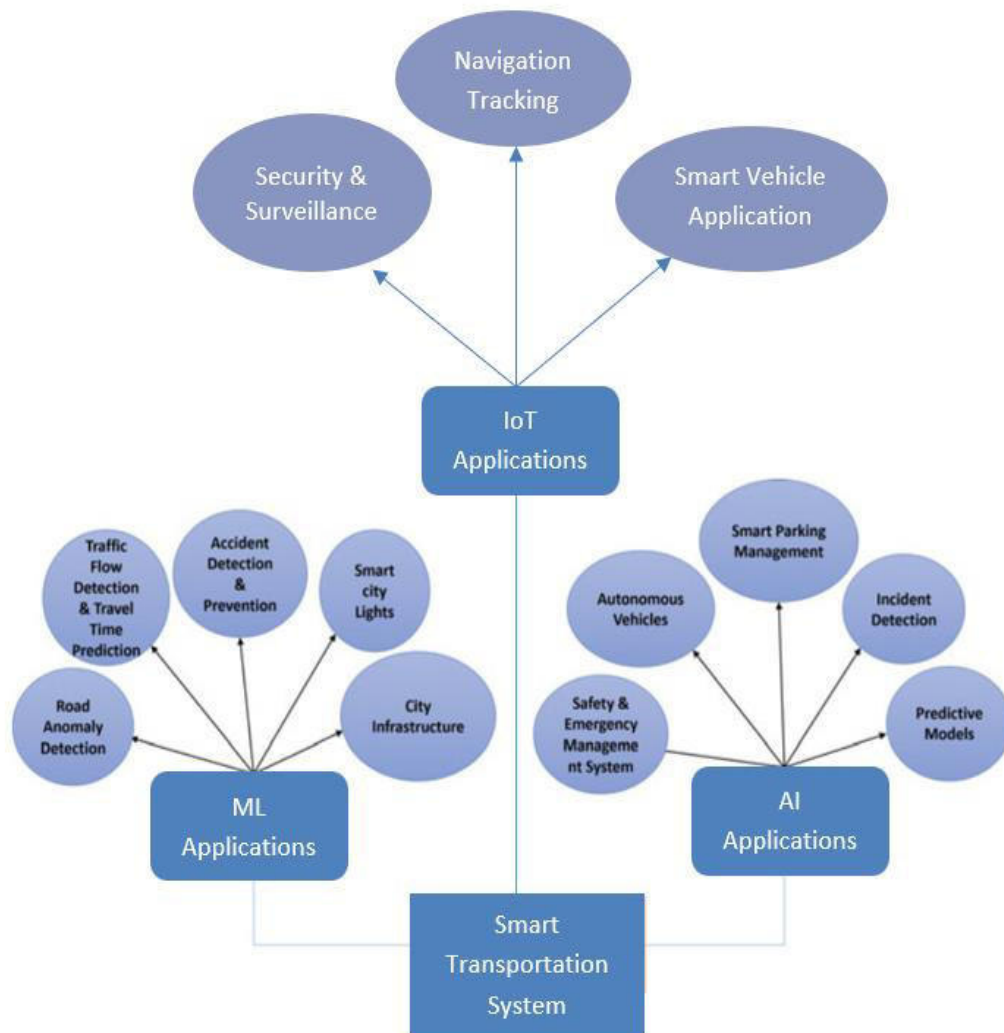
1. Identify the problems.
2. Identify sources of information / data.
3. Pre-process the data for missing and incorrect data and transform the data if required.
4. Divide the data into training and testing datasets.
5. Build ML models and identify the best model performance in validation data.
6. Implement solution / develop product.

An algorithm can be called as learning algorithm when it improves on a performance metric while performing a task.

5.2 Machine Learning Predictive Model for Smart Transportation

A Machine Learning Predictive Model for a Smart Transportation System integrates data from diverse sources, including traffic cameras, GPS devices, and weather forecasts, to predict traffic congestion, optimize vehicle routing, enhance public transportation efficiency, and improve the overall transportation experience. By analyzing historical and real-time data, these models offer solutions such as rerouting traffic, suggesting efficient routes for logistics, predicting public transportation demand, enabling predictive maintenance, promoting sustainability, enhancing security, and informing government policies, ultimately revolutionizing the way transportation is managed and transforming urban mobility for the better.

Figure 5.2: Smart Transportation System



The Smart Transportation System is being divided into three modules which includes ML applications, AI applications and IoT applications as shown in Figure 5.2. These modules are further being divided into submodules such as the ML application module includes traffic flow detection and travel time prediction, accident detection and prevention, smart city lights city, infrastructure and road anomaly detection. Similarly, the AI application is being subdivided into safety and emergency management, autonomous vehicles, smart parking management, incident detection and predictive models. The IoT applications include security surveillance, smart vehicle application and navigation.

5.3 Analysis of Machine Learning Models for Smart Transportation

Machine Learning Predictive Models for Smart Transportation provide data-driven insights and solutions to address complex urban mobility challenges. These models harness vast datasets from traffic sensors, GPS devices, and various sources to forecast traffic patterns, optimize routes, and improve public transportation systems. By leveraging historical and real-time data, they enable more efficient traffic management, reduce congestion, enhance user experience, and promote sustainable transportation practices. Additionally, these predictive models have the potential to play a pivotal role in shaping future transportation policies and infrastructure development for smarter and more accessible cities. Various Algorithms were used for feature/attribute extraction and selection. Based on the results provided by Algorithms out of twenty one attributes following seven attributes are selected for Machine Learning Algorithms.

1. SPEED
2. NUM_READS
3. HOUR
4. ZIP CODES
5. REGION
6. BUS_COUNT
7. CLASS LABEL

5.3.1 Performance Measure

To analyze different prediction models, the performance measure like accuracy, incorrectly classified instances, Kappa statistic, precision, recall, F-measure , ROC Area ,TP Rate, FP Rate, precision and recall were being used, which are explained below.

Accuracy: It is a commonly used metric to evaluate the performance of machine learning classification models. It measures the ratio of correctly predicted instances to the total number of instances in the dataset. The formula for precision is:

$$\text{Accuracy} = \frac{\text{Number of Correct Predicted values}}{\text{Total Number of instances in the data set}} \quad [i]$$

Incorrectly Classified Instances: In machine learning, a misclassified instance is a data point or instance in a data set that is incorrectly predicted or labeled by a machine learning model. These instances represent errors that the model made in its predictions. There are two types of errors:

- **False Positive:** Negative Instances are predicted as Positive by Model. In terms of Traffic Congestion this can be low traffic is predicted as heavy traffic.
- **False Negative:** Positive Instances are predicted as Negative by Model. In terms of Traffic Congestion this can be heavy traffic is predicted as low traffic.

Kappa Statistics: The kappa statistic, also known as Cohen's kappa, is a measure of agreement or reliability between classification algorithms. It is often used to assess agreement between two classification algorithms. Kappa value ranges from -1 to +1. Positive 1 indicates perfect agreement and Negative 1 indicates worst agreement. The various agreement interpretations are given below.

- $\text{Kappa} > 0.8$: Excellent Agreement
- $0.6 < \text{Kappa} < 0.8$: Good Agreement
- $0.4 < \text{Kappa} < 0.6$: Moderate Agreement
- $\text{Kappa} \leq 0.4$: Poor Agreement

Confusion Matrix Parameters: Confusion matrix shows the different ways in which the classification model gets confused when making predictions. The predicted values are compared with Actual values to find out various performance parameters. The Parameter like TP Rate, FP Rate, Precision, Recall and ROC Area are derived from a confusion matrix.

Figure 5.3: Confusion Matrix

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Source: Towards Data Science [23]

TP⁶²: It refers to the number of predictions where the classifier correctly predicts the positive class as positive. For example in terms of Traffic Congestion this can be heavy traffic is predicted as heavy traffic.

TN⁶³: It refers to the number of predictions where the classifier correctly predicts the negative class as negative. For example in terms of Traffic Congestion this can be low traffic is predicted as low traffic.

FP⁶⁴: It refers to the number of predictions where the classifier incorrectly predicts the negative class as positive. For example in terms of Traffic Congestion this can be low traffic is predicted as heavy traffic.

FN⁶⁵: It refers to the number of predictions where the classifier incorrectly predicts the positive class as negative. For example in terms of Traffic Congestion this can be heavy traffic is predicted as low traffic.

Precision: It is the quality of a positive prediction made by the model. Precision refers to the number of True Positives divided by the total number of Positive predictions.

$$\text{Precision} = \frac{TP}{TP+FP} \quad [\text{ii}]$$

Recall: It is the measures of how well a machine learning model can detect positive instances. It is also called as Sensitivity. Sensitivity refers to the number of true

⁶² True Positive

⁶³ True Negative

⁶⁴ False Positive

⁶⁵ False Negative

positives divided by the sum of True Positives and False Negatives. The model with high Sensitivity will have significantly fewer False Negatives.

$$\text{Recall} = \frac{TP}{TP+FN} \quad [\text{iii}]$$

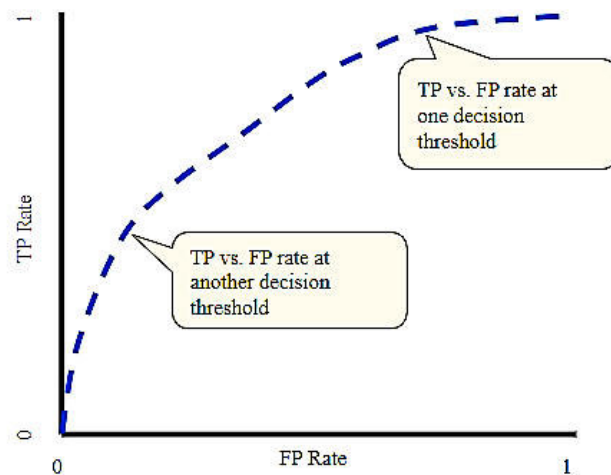
F-Measure: It also known as F1-score, It is a machine learning metric that combines precision and recall into one value.

$$\text{F-Measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad [\text{iv}]$$

F-Measure value changes between 0 and 1, Where 1 indicates ideal Precision and Recall and 0 indicates poor performance.

ROC⁶⁶: It is a graph showing the performance of a classification mode. This curve plots two parameters: TPR⁶⁷ and FPR⁶⁸. TP Rate is used to measure the percentage of actual positives which are correctly identified by model . TPR is synonym for Recall. FP Rate also known as Type - I error is used to measure the percentage of actual positives which are incorrectly identified by model.

Figure 5.4: ROC Curve



Source: Google developer site [27]

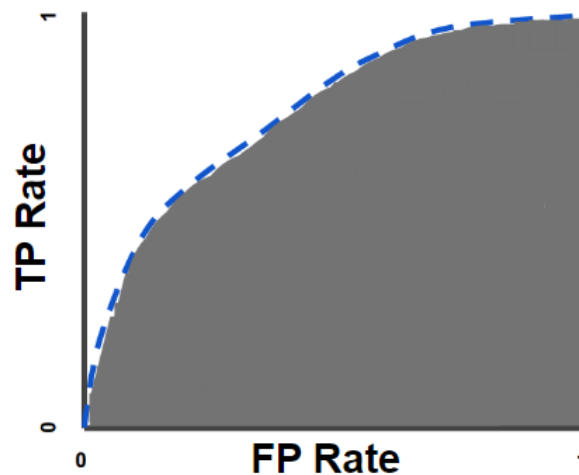
⁶⁶ Receiver Operating Characteristics

⁶⁷ True Positive Rate

⁶⁸ False Positive Rate

ROC Area: AUC⁶⁹ measures the entire two-dimensional area underneath the entire ROC curve. AUC indicates how well predictions are ranked. AUC ranges in value from 0 to 1. A model whose predictions are 100% wrong has an AUC of 0.0; one whose predictions are 100% correct has an AUC of 1.0.

Figure 5.5: Area Under ROC Curve



Source: Google developer site [27]

In summary, performance measurements play a critical role in evaluating the effectiveness of machine learning algorithms, providing insight into their ability to make accurate predictions and transform appropriately to new, unseen data. Choosing the most appropriate metric depends on the nature of your particular problem, the characteristics of your data, and your analysis goals.

5.3.2. Error Measures

Machine learning uses various error measures to evaluate the performance of algorithms and models. These measurements help quantify the difference between predicted and actual values and provide insight into model performance. Common error remedies includes:

Mean Absolute Error: it is Average of absolute differences between Actual values and Predicted values. Lower the value of Mean Absolute Error, better the Prediction Algorithm.

⁶⁹ Area Under ROC Curve

$$\text{Mean Absolute Error} = \frac{1}{n} \sum_{i=1}^n |y_{a_i} - y_{p_i}| \quad [\text{v}]$$

Where

- y_a = Actual Output Value
- y_p = Predicted Output Value

Root Mean Square Error: It is the root of mean square error. It considers the effect of negative as well positive errors in considerations.

$$\text{Root Mean Square Error} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_{a_i} - y_{p_i})^2} \quad [\text{vi}]$$

Relative Absolute Error: It is used to measure accuracy of predictions. It compares the distance of actual values from predicted values with distance of actual values from average values. Its value lie between 0 to positive infinite. A lower Relative Absolute Error indicates better performance.

$$\text{Relative Absolute Error} = \frac{\sum_{i=1}^n |y_{a_i} - y_{p_i}|}{\sum_{i=1}^n |y_{a_i} - \bar{y}|} \quad [\text{vii}]$$

Where

$$\bar{y} = \text{Mean of Actual Values}$$

Root Relative Square Error: It is the root of relative square error. The relative square error is the ratio of total square error to average of actual values. Relative square error normalizes total square error.

$$\text{Relative Square Error} = \frac{\sum_{i=1}^n (y_{a_i} - y_{p_i})^2}{\bar{y}} \quad [\text{viii}]$$

By Taking root of Relative Square Error we are reducing the normalize square error value and bringing it closer to predicted value.

$$\text{Root Relative Square Error} = \sqrt{\text{Relative Square Error}} \quad [\text{ix}]$$

$$\text{Root Relative Square Error} = \sqrt{\frac{\sum_{i=1}^n (y_{a_i} - y_{p_i})^2}{\bar{y}}} \quad [\text{x}]$$

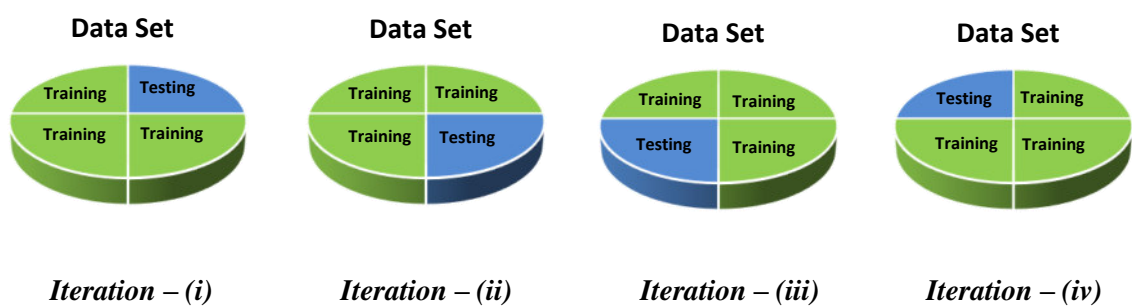
Lower the value of Root Relative Square Error, better the performance of Machine Learning prediction model. The choice of error measure depends on the nature of the problem and the specific objectives of the analysis. When choosing an appropriate error measure, it is important to consider the characteristics of the data and the objectives of the modeling task.

In summary, understanding and effectively using error counter measures is a fundamental aspect of creating and improving machine learning algorithms. Error measurements provide a quantitative means of assessing model accuracy and performance, allowing practitioners to make informed decisions, compare different algorithms, and optimize predictive capabilities.

5.3.3 Cross-Validation Configuration Setting (10-folds) Results

The Weka tool was being used for the analysis of various classification machine learning algorithms. K-folds Cross-Validation Approach is used for evaluating the Performance of Machine learning Algorithms, where K value is changed to study different cases. For example in 4 folds Cross-Validation K value is 4 where data set is divided into 4 parts, out of which 3 parts are used for training the Machine learning Algorithm and only One part is used for testing the Algorithm.

Figure 5.6: 4-fold Cross-Validation Example



As shown in above figure 5.6, four iterations are executed for 4-folds, In first iteration part one will be used for testing and remaining three parts will be used for training. In second iteration part two will be used for testing and remaining three parts will be used for training. This recursion will continue up to the last iteration to complete the Cross-Validation. For 10-fold Cross-Validation K value is 10 and data set will be divided into 10 parts, Nine parts for Training the algorithm and One part for testing the algorithm.

Similarly for 25-fold Cross-Validation K value is 25 and data set will be divided into 25 parts, Twenty four parts for Training the algorithm and One part for testing the algorithm.

Data Analysis Credentials:

Dataset: Udaipur_Traffic

Source: TOMTOM Server

Date: October 2023

Duration: One Month

Number of Instances: 1000

Number of Attributes (After Feature Extraction and Selection): 7

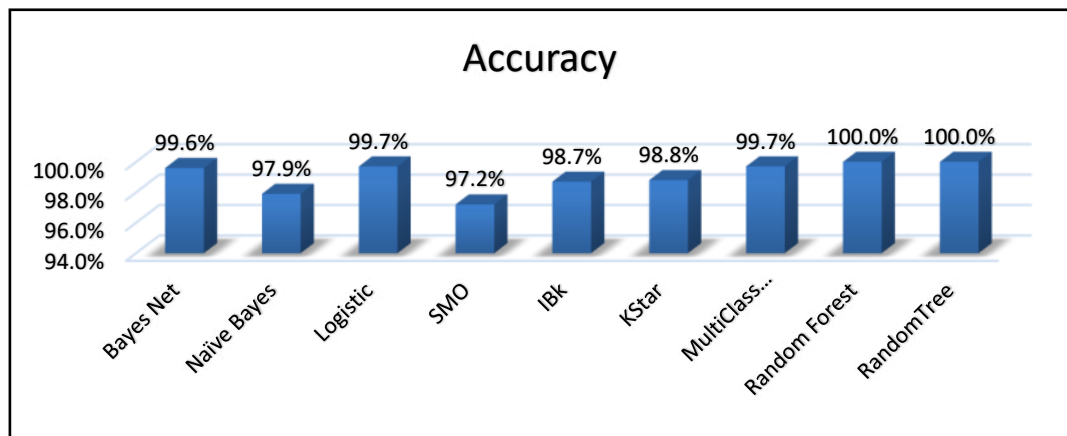
A. Performance Measures

i. Accuracy Measures

Table 5.1: Classifiers and Accuracy Measures (Cross Validation: 10-Folds)

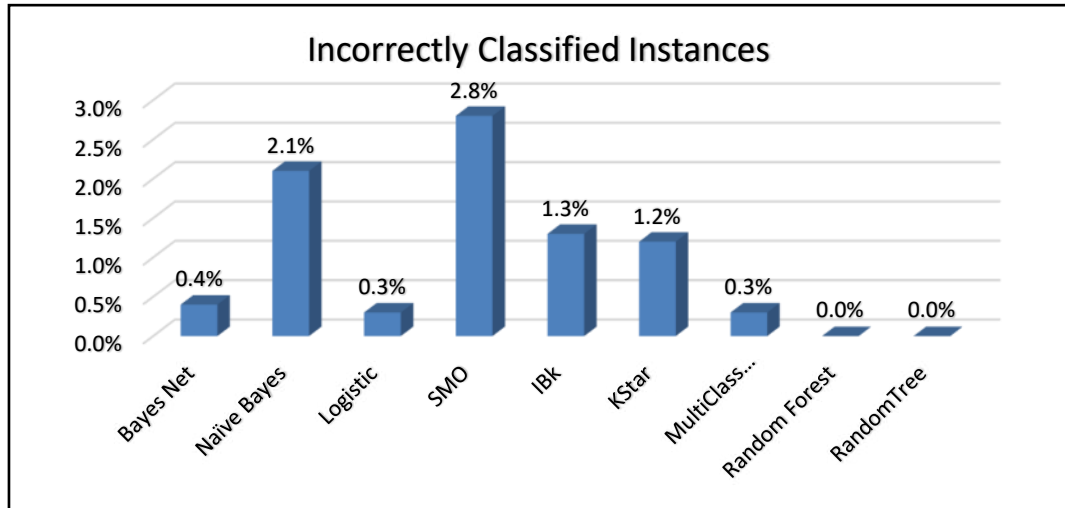
Classifier	Accuracy	Incorrectly Classified Instances	Kappa statistic
Bayes Net	99.6%	0.4%	0.967
Naïve Bayes	97.9%	2.1%	0.815
Logistic	99.7%	0.3%	0.976
SMO	97.2%	2.8%	0.722
IBk	98.7%	1.3%	0.898
KStar	98.8%	1.2%	0.905
MultiClass Classifier	99.7%	0.3%	0.976
Random Forest	100.0%	0.0%	1.000
RandomTree	100.0%	0.0%	1.000

Figure 5.7: Performance Measure Accuracy (Cross-Validation: 10 Folds)



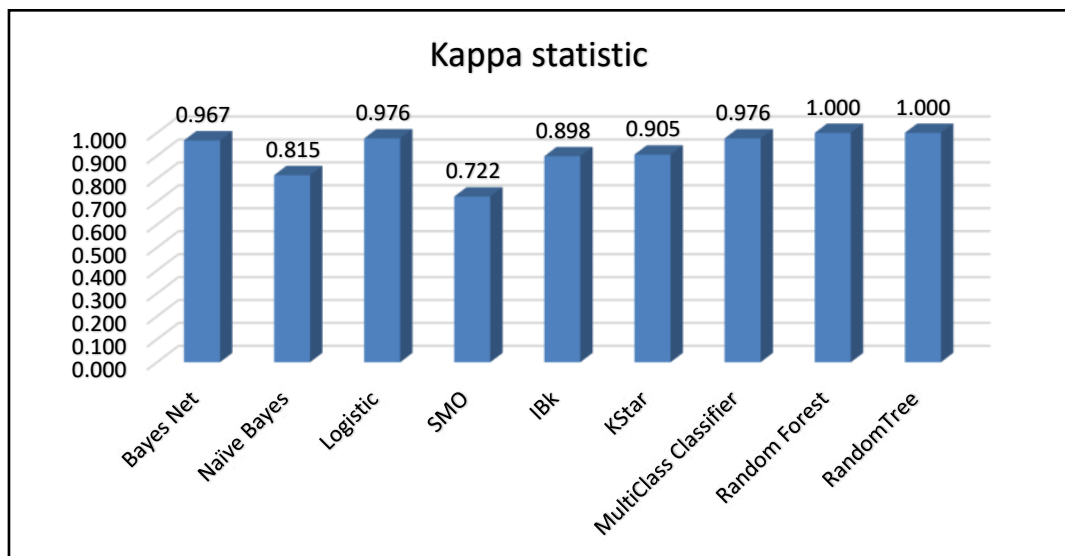
Based on the performance measure accuracy it can be interpreted that Random Forest classifier was the most appropriate one as it was having the highest accuracy value of 100% whereas SMO and Naïve Bayes were having the lowest value of accuracy 97.2% and 97.9%.

Figure 5.8: Incorrectly Classified Instances (Cross-Validation: 10 Folds)



According to the performance measure incorrectly classified instances it can be interpreted that Random Forest and Random Tree classifiers were the most appropriate one as these were having the lowest number of incorrectly classifies instances accounting for 0% each whereas SMO and Naïve Bayes classifiers were having the highest number of incorrectly classifies instances accounting for 2.8% and 2.1% respectively.

Figure 5.9: Kappa Statistic Values (Cross-Validation: 10 Folds)



The provided data consists of a set of Kappa statistic values, which are used to assess the agreement or consistency between classifiers in different situations. These Kappa values range from 0.722 to 1.000, indicating varying levels of agreement. The highest Kappa value, 1.000, suggests a very good level of agreement between the classifiers in that particular scenario, while the lowest value, 0.722, falls into the good to moderate agreement range. Overall, the data suggests that there is a generally positive level of agreement in the assessed situations, with some instances demonstrating higher agreement than others.

In correctly classified instances and kappa statistics are performance metrics which are important to explore to get more comprehensive insights to develop accurate machine learning model.

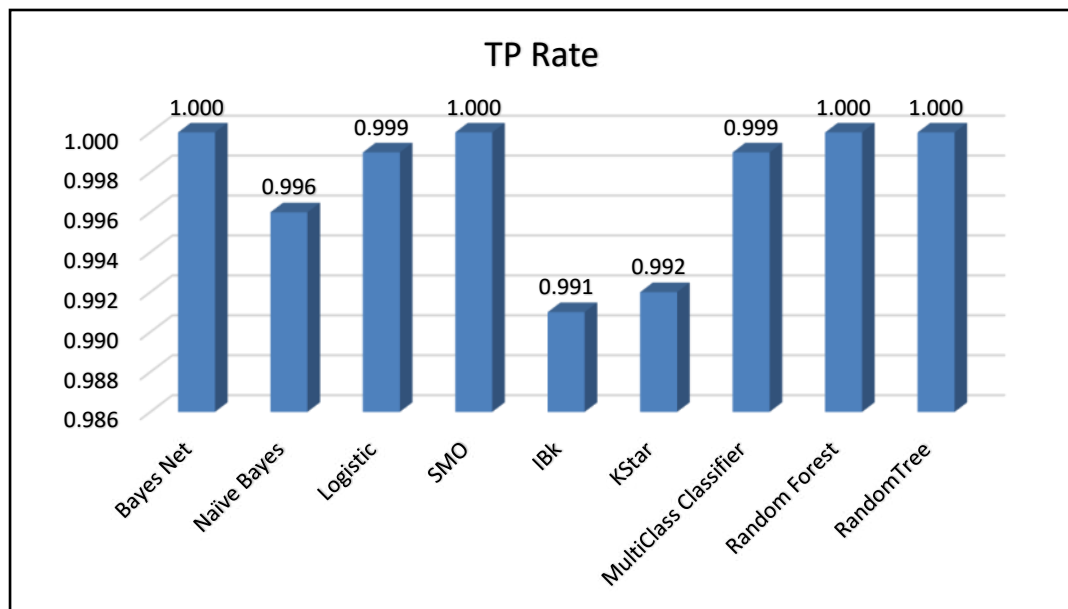
ii. Confusion Matrix Parameters – Low Traffic

Table 5.2 shows confusion matrix parameters TP Rate, FP Rate, Precision, Recall, F-Measure and ROC Area for Low Traffic case. Machine learning often requires a limited amount of data when dealing with low-traffic scenarios. In such cases, the challenge is to create a robust model despite data limitations. Data Augmentation techniques are used to artificially increase the size of data set which can be helpful especially in low traffic scenarios.

*Table 5.2: Classifiers and Performance Measures Class Label: Low Traffic
Cross Validation: 10-Folds*

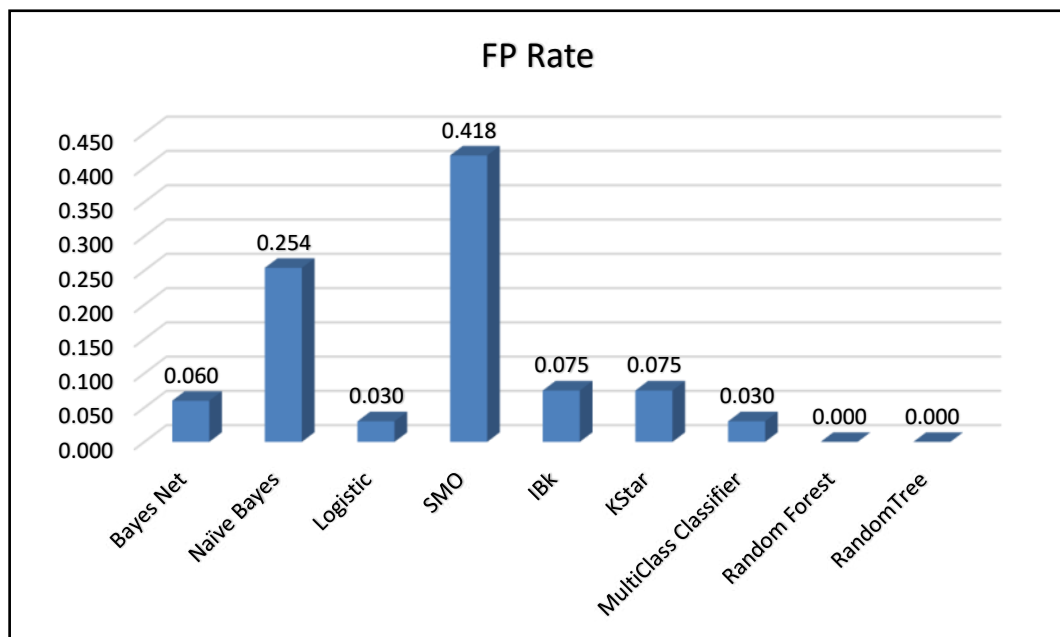
Classifier	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
Bayes Net	1.000	0.060	0.996	1.000	0.998	1.000
Naïve Bayes	0.996	0.254	0.982	0.996	0.989	0.988
Logistic	0.999	0.03	0.998	0.999	0.998	1.000
SMO	1.000	0.418	0.971	1.000	0.985	0.791
IBk	0.991	0.075	0.995	0.991	0.993	0.958
KStar	0.992	0.075	0.995	0.992	0.994	0.997
MultiClass Classifier	0.999	0.030	0.998	0.999	0.998	1.000
Random Forest	1.000	0.000	1.000	1.000	1.000	1.000
RandomTree	1.000	0.000	1.000	1.000	1.000	1.000

Figure 5.10: TP Rate (Cross Validation: 10-Folds – Low Traffic)



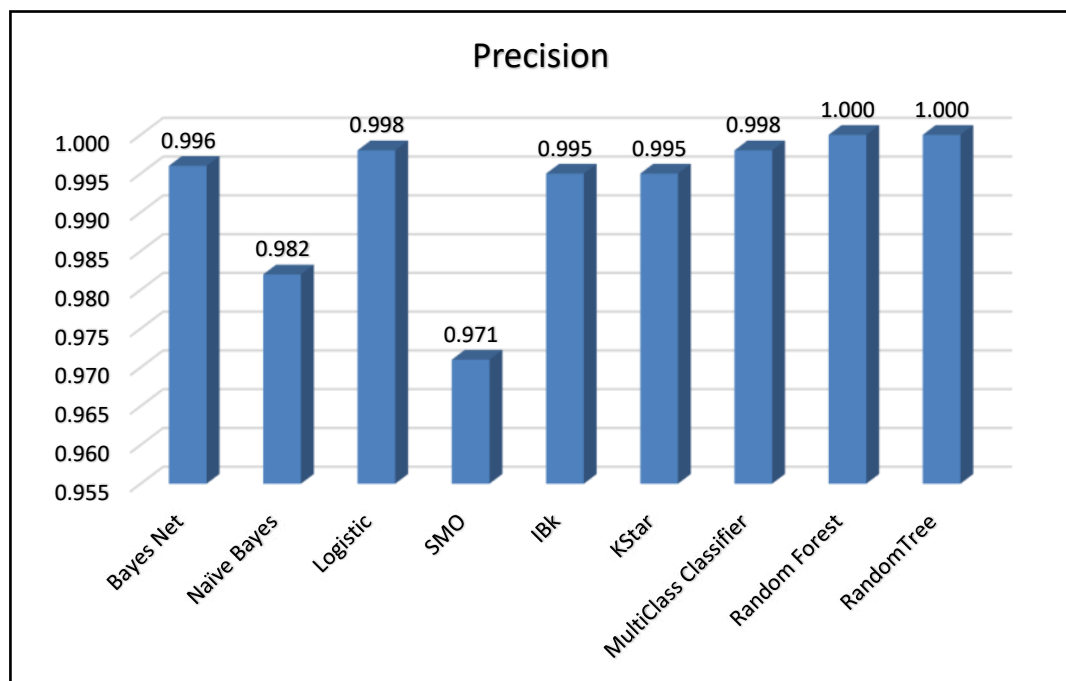
According to the performance measure TP rate it was found that the highest true positive rate was of the classifiers Bayes Net, Random Forest and Random Tree with value 1.000, followed by 0.999 and 0.999 of Logistic and MultiClass Classifier respectively whereas the lowest TP rate was found to be of the classifiers IBK with value 0.991. Overall it can be interpreted that there are three most appropriate classifiers based on the performance measure TP rate with the value of 1.000.

Figure 5.11: FP Rate (Cross Validation: 10-Folds – Low Traffic)



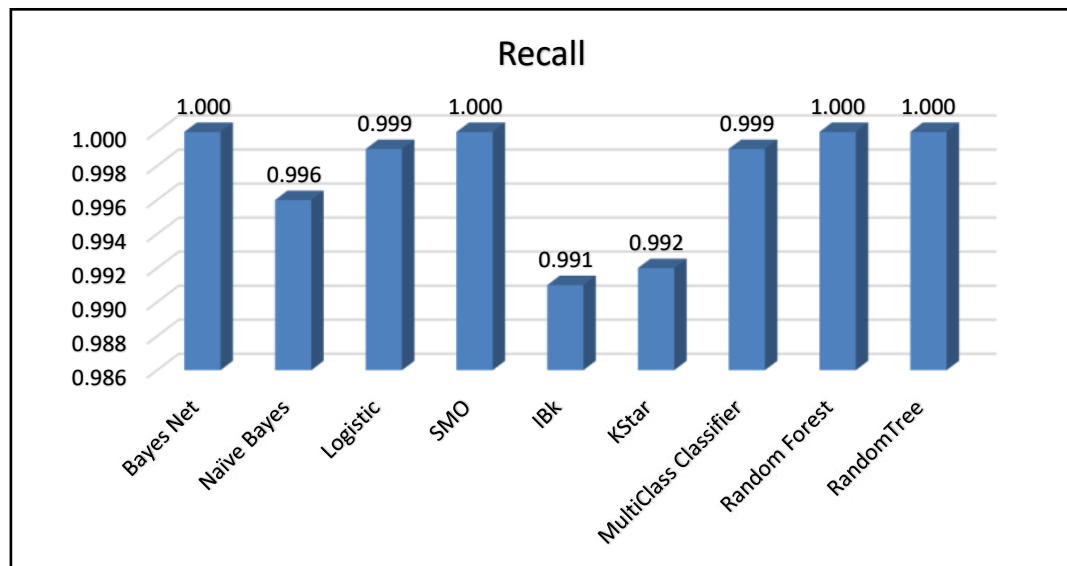
Based on the performance measure FP rate it was found that the lowest false positive rate was of the classifiers Random Forest and Random Tree with value 0.000, followed by 0.030 of Logistic and MultiClass Classifier whereas the highest FP rate was found to be of the classifiers SMO with value 0.418. Overall, it can be interpreted the most appropriate classifier based on the performance measure FP rate is found to be Random Forest and Random Tree with lowest FP rate value.

Figure 5.12: Precision (Cross Validation: 10-Folds – Low Traffic)



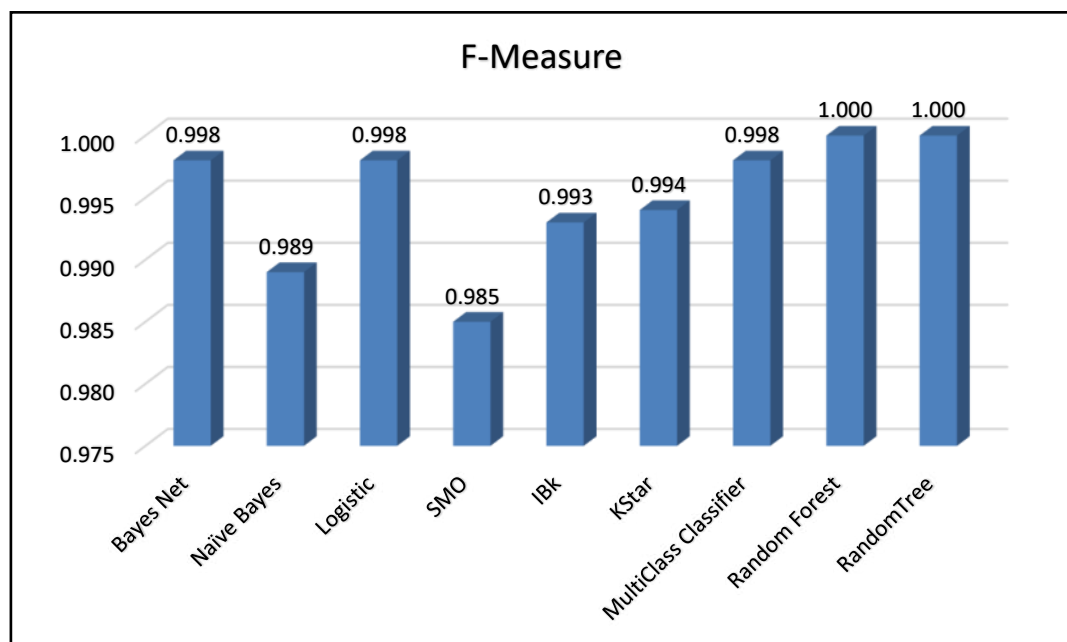
According to the performance measure precision it was found that the highest precision value was of the classifier Random Forest and Random Tree with value 1.000, followed by 0.998 of Logistic and MultiClass Classifier respectively whereas the lowest precision value was found to be of the classifier SMO with value 0.971. Overall, it can be interpreted the most appropriate classifiers based on the performance measure precision are found to be Random Forest and Random Tree.

Figure 5.13: Recall (Cross Validation: 10-Folds – Low Traffic)



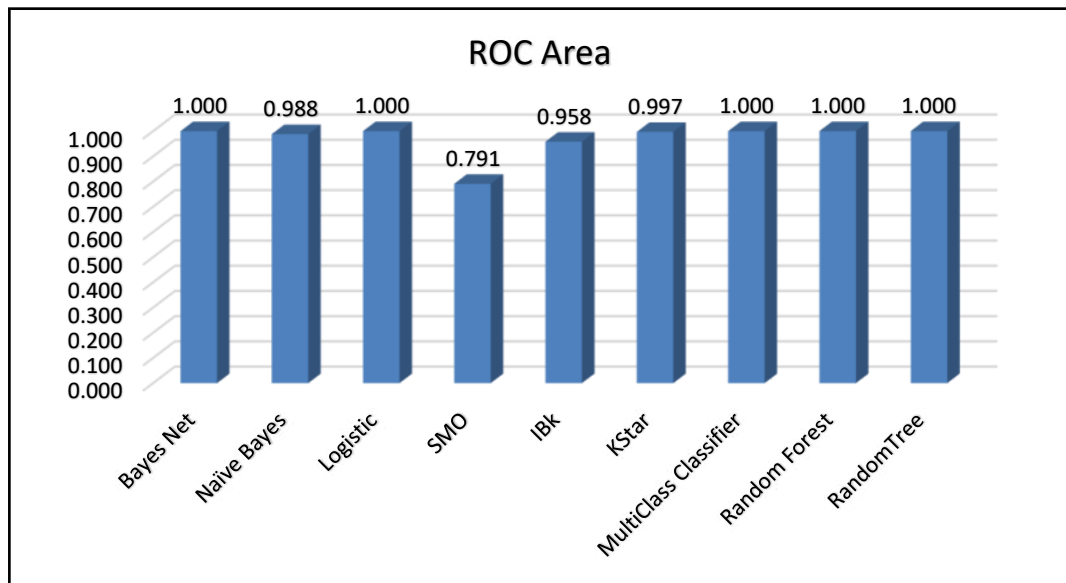
Based on the performance measure recall it was found that the highest recall value was of the classifiers Bayes Net, SMO, Random Forest and Random Tree with value 1.000, followed by 0.999 of Logistic and MultiClass Classifier respectively whereas the lowest recall value was found to be of the classifier IBK with value 0.991. Overall, it can be interpreted the most appropriate classifier based on the performance measure recall is found to be four algorithms Bayes Net, SMO, Random Forest and Random Tree.

Figure 5.14: F-Measure (Cross Validation: 10-Folds – Low Traffic)



According to the performance measure F-Measure it was found that the highest F-Measure values were of the classifier Random Forest and Random Tree with value 1.000, followed by 0.998 value of Bayes Net, Logistic and MultiClass Classifier respectively whereas the lowest F-Measure value was found to be of the classifier SMO classifier with values 0.985. Overall, it can be interpreted the most appropriate classifiers based on the performance measure F-Measure is found to be Random Forest and Random Tree.

Figure 5.15: ROC Area (Cross Validation: 10-Folds – Low Traffic)



Based on the performance measure ROC it was found that the highest ROC Area value was of the classifiers Bayes Net, Logistic, MultiClass Classifier, Random Forest and Random Tree with value 1.000, followed by 0.997 and 0.988 of KStar and Navie Bayes respectively whereas the lowest ROC Area value was found to be of the classifier SMO with value 0.791 respectively. Overall, it can be interpreted the most appropriate classifiers based on the performance measure ROC Area are found to be five Algorithms.

In summary, performance measurements play a critical role in evaluating the effectiveness of machine learning algorithms, providing insight into their ability to make accurate predictions and transform appropriately to new, unseen data. Choosing the most appropriate metric depends on the nature of your particular problem, the characteristics of your data, and your analysis goals.

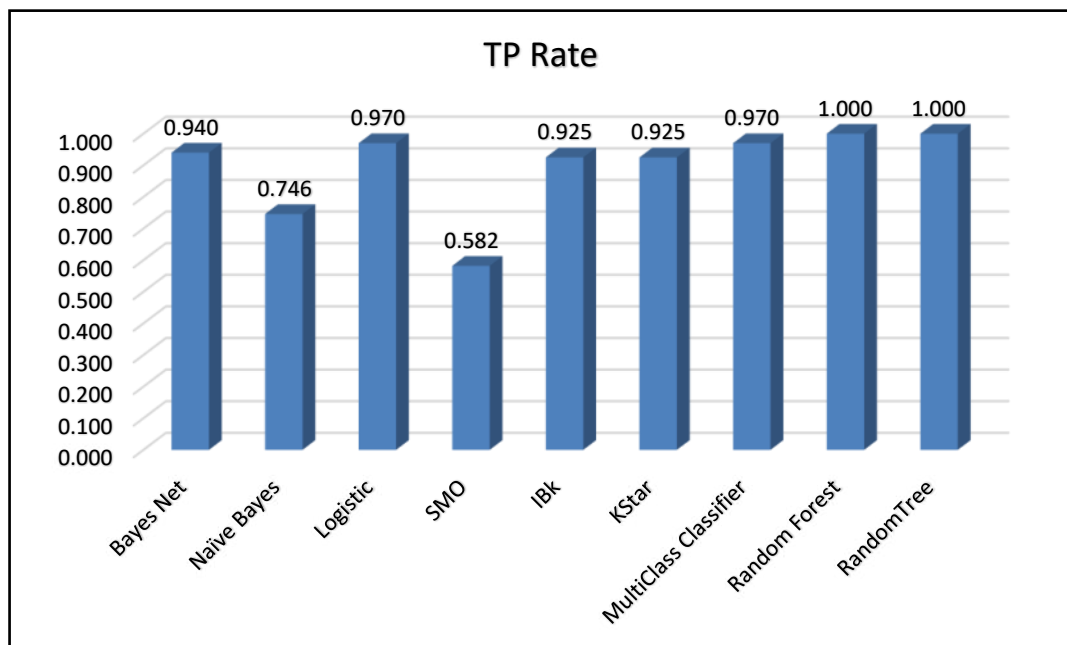
iii. Confusion Matrix Parameters – Heavy Traffic

Heavy Traffic generates huge amounts of data from the various IOT sensors. These data sets can be used by Machine Learning Algorithms to develop prediction models. Table 5.3 shows the Confusion Matrix parameters obtained for Heavy Traffic conditions.

Table 5.3: Classifiers Performance Measure Class Label: Heavy Traffic
Cross Validation: 10-Folds

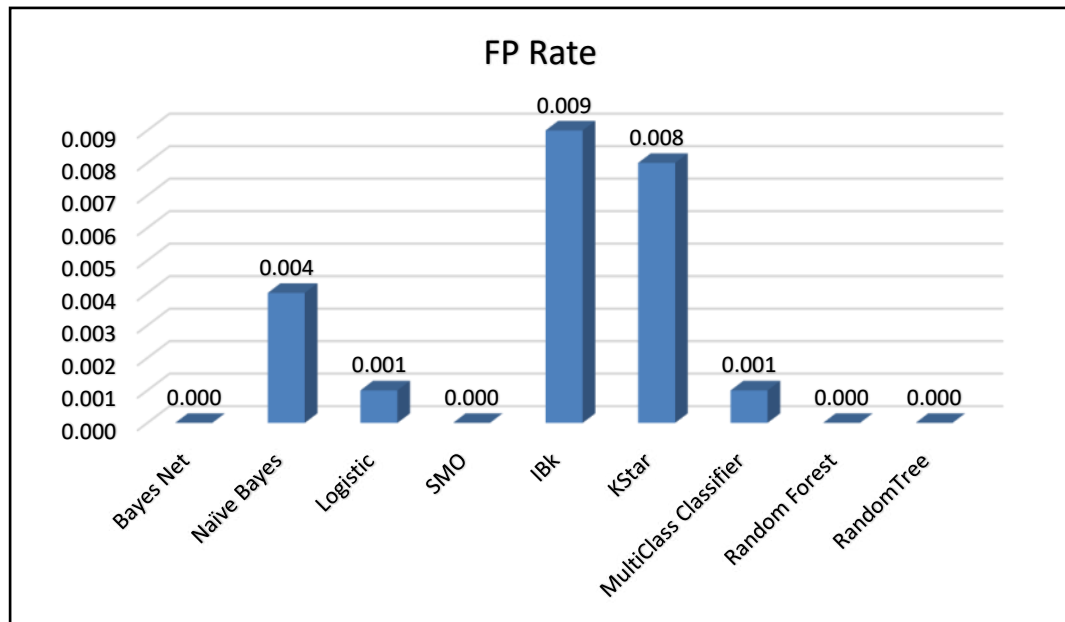
Classifier	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
Bayes Net	0.940	0.000	1.000	0.940	0.969	1.000
Naïve Bayes	0.746	0.004	0.926	0.746	0.826	0.988
Logistic	0.970	0.001	0.985	0.970	0.977	1.000
SMO	0.582	0.000	1.000	0.582	0.736	0.791
IBk	0.925	0.009	0.886	0.925	0.905	0.958
KStar	0.925	0.008	0.899	0.925	0.912	0.997
MultiClass Classifier	0.970	0.001	0.985	0.970	0.977	1.000
Random Forest	1.000	0.000	1.000	1.000	1.000	1.000
RandomTree	1.000	0.000	1.000	1.000	1.000	1.000

Figure 5.16: TP Rate (Cross Validation: 10-Folds – Heavy Traffic)



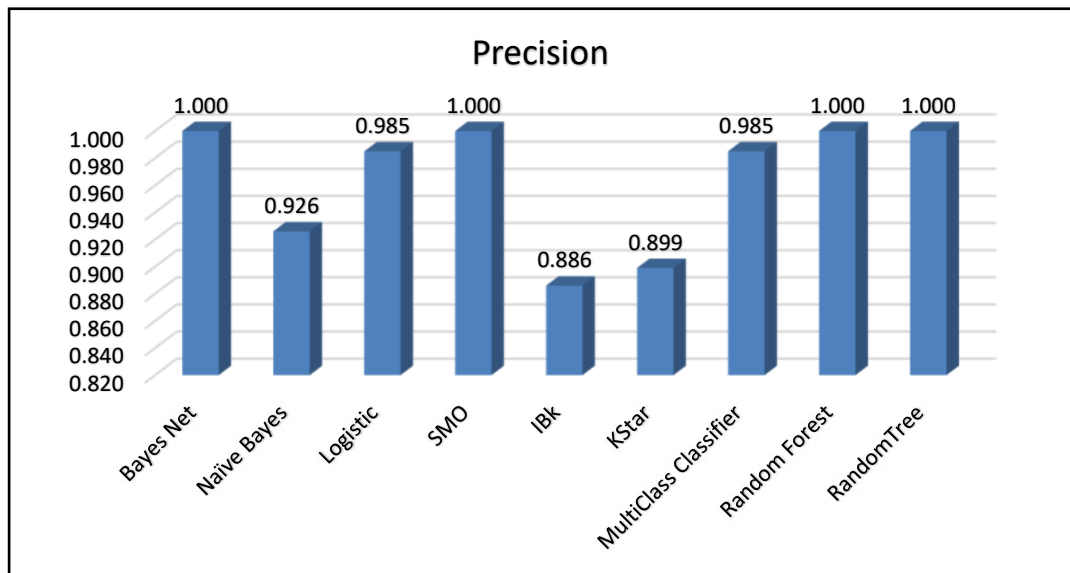
According to the performance measure TP rate for class label: Heavy Traffic it was found that the highest true positive rate was of the classifiers Random Forest and Random Tree 1.000, followed by 0.970 of Logistic and MultiClass Classifier respectively whereas the lowest TP rate was found to be of the classifiers SMO with values 0.582. Overall, it can be interpreted the most appropriate classifier based on the performance measure TP rate are Random Forest and Random Tree.

Figure 5.17: FP Rate (Cross Validation: 10-Folds – Heavy Traffic)



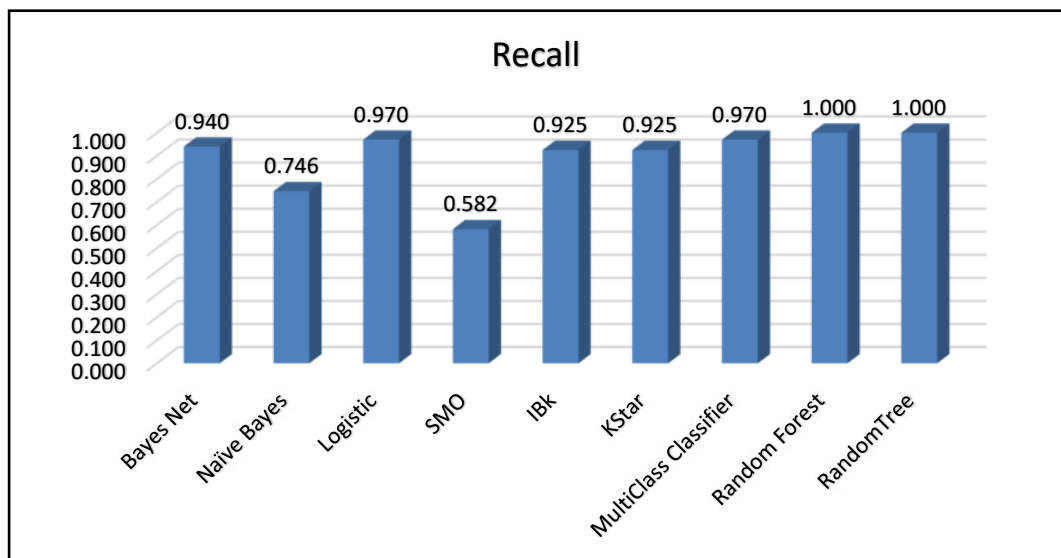
Based on the performance measure FP rate for class label: Heavy Traffic it was found that the lowest false positive rate was of the classifiers Bayes Net, SMO, Random Forest and Random Tree with value 0.00, followed by 0.001 of Logistic and MultiClass Classifier, whereas the highest FP rate was found to be of the classifiers KStar and IBK with values 0.008 and 0.009 respectively. Overall, it can be interpreted the most appropriate classifier based on the performance measure FP rate are found to be four Algorithms with lowest FP rate value 0.000.

Figure 5.18: Precision (Cross Validation: 10-Folds – Heavy Traffic)



According to the performance measure precision class label: Heavy Traffic it was found that the highest precision value was of the classifiers Bayes Net, SMO, Random Forest and Random Tree with value 1.000, followed by 0.985 and 0.926 of Logistic, MultiClass Classifier and Naïve Bayes respectively whereas the lowest precision value was found to be of the classifier IBK with values 0.886 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure precision is found to be Four Algorithms.

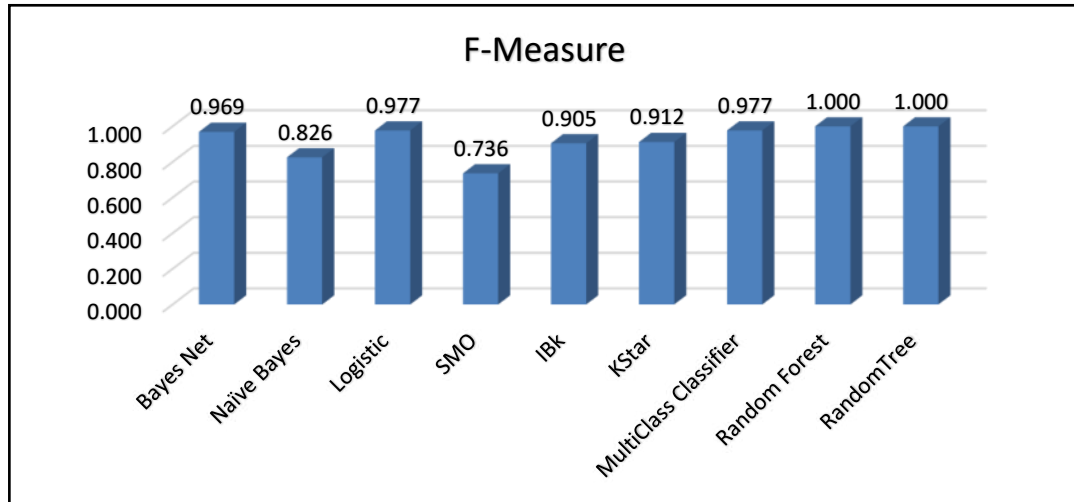
Figure 5.19: Recall (Cross Validation: 10-Folds – Heavy Traffic)



Based on the performance measure recall class label: Heavy Traffic it was found that the highest recall value was of the classifiers Random Forest and Random Tree with

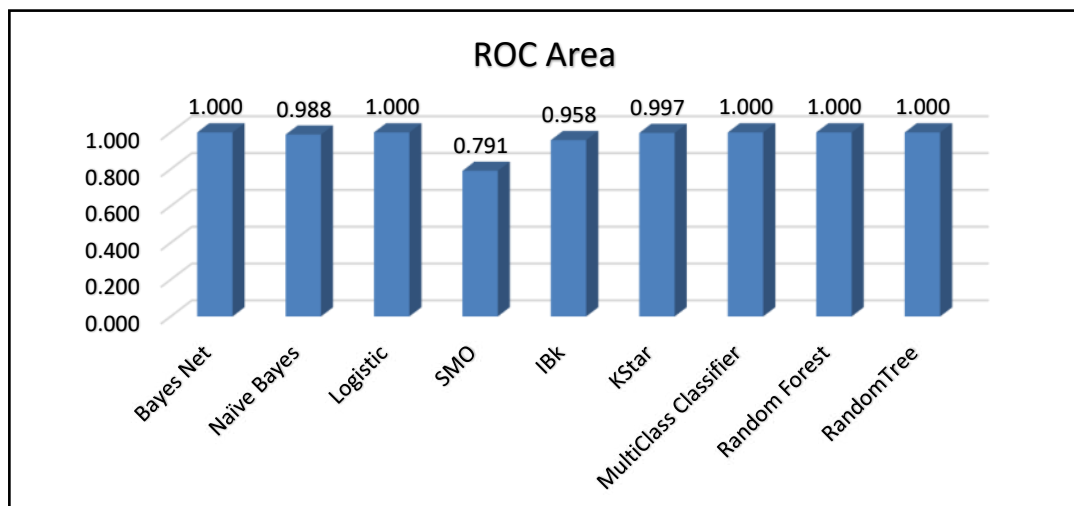
value 1.000, followed by 0.970 of Logistic and Multiclass Classifier respectively whereas the lowest recall value was found to be of the classifiers SMO with value 0.582. Overall, it can be interpreted the most appropriate classifier based on the performance measure recall are found to be Random Forest and Random Tree.

Figure 5.20: F-Measure (Cross Validation: 10-Folds – Heavy Traffic)



According to the performance measure F-Measure class label: Heavy Traffic it was found that the highest F-Measure value was of the classifiers Random Forest and Random Tree with value 1.000, followed by 0.9777 of Logistic and Multiclass Classifier respectively whereas the lowest F-Measure value was found to be of the classifier SMO with value 0.736. Overall, it can be interpreted the most appropriate classifiers based on the performance measure F-Measure is found to be Random Forest and Random Tree.

Figure 5.21: ROC Area (Cross Validation: 10-Folds – Heavy Traffic)



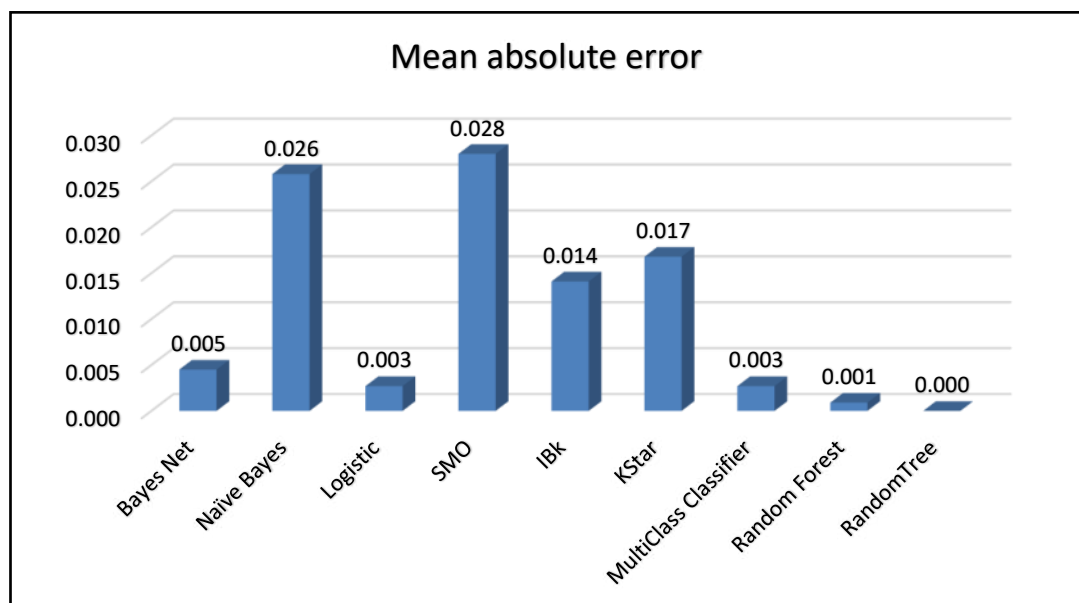
Based on the performance measure ROC class label: Heavy Traffic it was found that the highest ROC Area value was of the classifiers Bayes Net, Logistic, MultiClass Classifier, Random Forest and Random Tree with value 1.00, followed by 0.988 of Naive Bayes respectively whereas the lowest ROC Area value was found to be of the classifier SMO with values 0.791 respectively. Overall, it can be interpreted that the most appropriate classifiers based on the performance measure ROC Area are found to be five Algorithms.

B. Error Measure Results

Table 5.4: Classifiers and Error Measures (Cross Validation: 10-Folds)

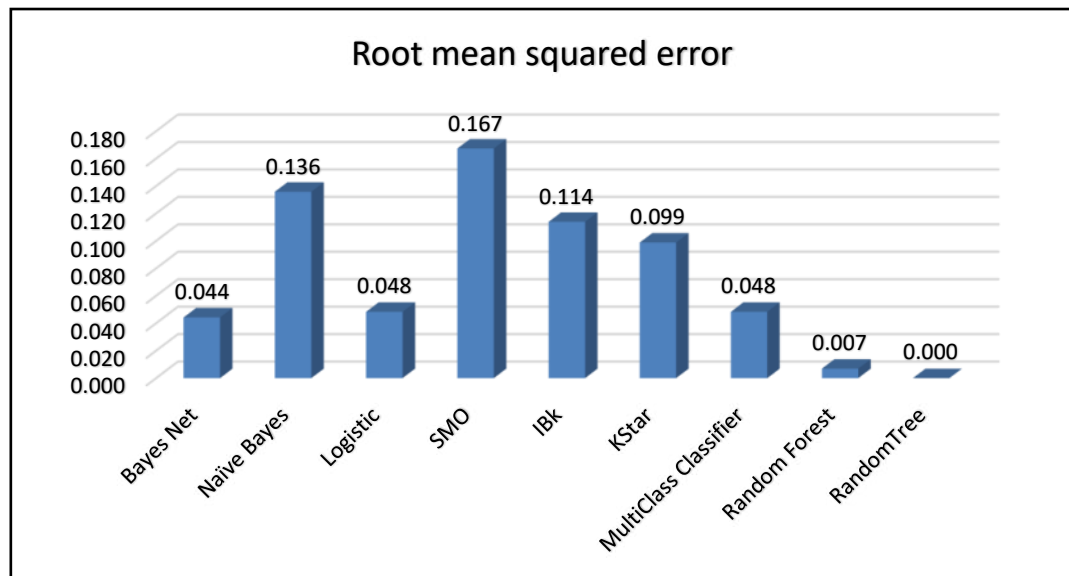
Classifier	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error
Bayes Net	0.005	0.044	3.549%	17.692%
Naïve Bayes	0.026	0.136	20.501%	54.312%
Logistic	0.003	0.048	2.171%	19.311%
SMO	0.028	0.167	22.247%	66.924%
IBk	0.014	0.114	11.180%	45.553%
KStar	0.017	0.099	13.340%	39.526%
MultiClass Classifier	0.003	0.048	2.171%	19.311%
Random Forest	0.001	0.007	0.739%	2.762%
RandomTree	0.000	0.000	0.000%	0.000%

Figure 5.22: Mean Absolute Error (Cross-Validation: 10 Folds)



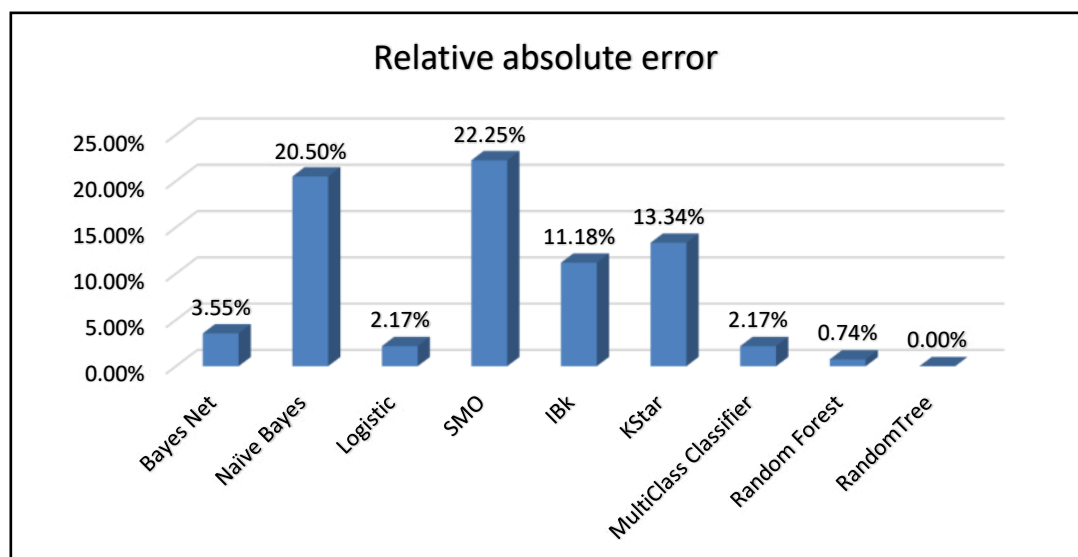
The mean absolute error is found to be lowest in case of Random Tree with the value 0.000 Whereas the mean absolute error value of SMO is found to be highest with value 0.028. So, it can be interpreted that based on the measure Mean absolute Error the most appropriate algorithm is found to be Random Tree at configuration setting – 10-fold cross validation.

Figure 5.23: Root Mean Squared Error (Cross-Validation: 10 Folds)



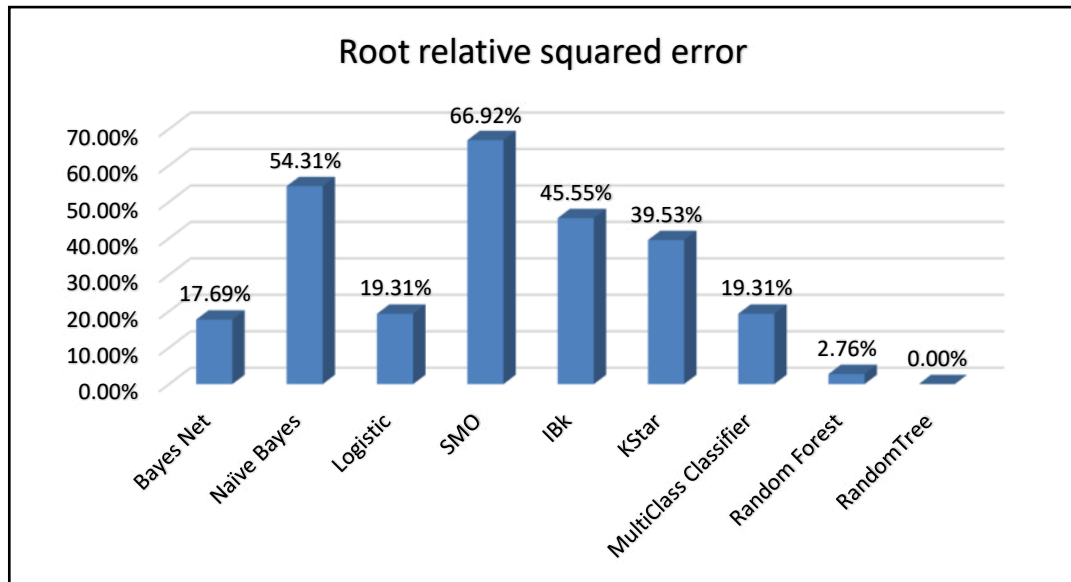
The root mean squared error value is found to be highest in case of SMO with the value of 0.167 whereas the lowest value is found to be of Random Tree with value 0.000. So, it can be interpreted that based on the measure RMSE the most appropriate algorithm is found to be Random Tree at configuration setting – 10-fold cross validation.

Figure 5.24: Relative Absolute Error (Cross Validation: 10-Folds)



Accordingly, the relative absolute error value is found to be lowest in case of Random Tree classifier with 0.00% whereas the highest relative absolute error percentage value is found to be in case of SMO with 22.25%. So, it can be suggested that based on the measure Relative Absolute Error the most appropriate algorithm is found to be Random Tree with lowest value when evaluated at configuration setting – 10-fold cross validation.

Figure 5.25 : Root Relative Squared Error (Cross Validation: 10-Folds)



The root relative squared error value is found to be lowest in case of Random Tree classifier with 0.00% whereas the root relative squared error percentage value is found to be highest in case of SMO with percentage value of 66.92%. So, it can be interpreted that based on the measure RRSE the most appropriate algorithm is found to be Random Tree with lowest percentage value when evaluated at configuration setting – 10-fold cross validation.

In summary, it is important to understand error measures in Machine learning for assessing the performance of model properly and making informed decisions. The choice of specific metrics depends on the nature of the problem, characteristics of the dataset, and goals of the analysis.

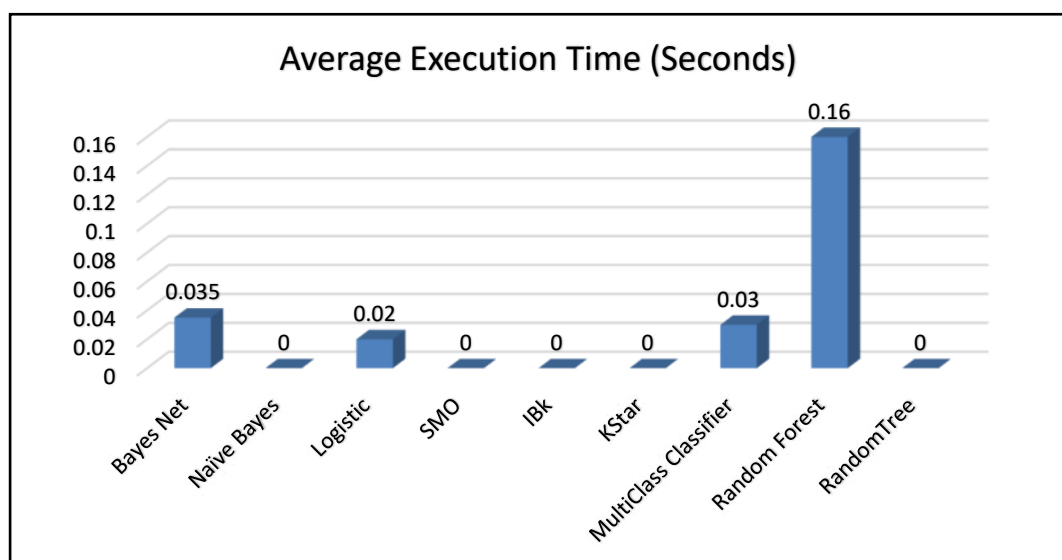
C. Execution Time Results

The execution time of a machine learning algorithm refers to the time it takes for the algorithm to process and analyse input data, train the model (if applicable), and produce predictions or results. Execution time is an important factor when evaluating the efficiency and scalability of machine learning algorithms, especially when dealing with large data sets and real-time applications. The Average Execution Time of Nine Classifier Algorithm is given below.

Table 5.5: Classifiers and Average Execution Time (Cross Validation: 10-Folds)

Classifier	Average Execution Time (Seconds)
Bayes Net	0.035
Naïve Bayes	0
Logistic	0.02
SMO	0
IBk	0
KStar	0
MultiClass Classifier	0.03
Random Forest	0.16
RandomTree	0

Figure 5.26 : Average Execution Time (Cross Validation: 10-Folds)



According to the performance measure average execution time it was found that the lowest average execution time were of the classifiers Naïve Bayes, SMO, IBK, KStar and Random Tree with values nearly 0 Seconds each whereas the highest average execution time was found to be of the classifiers Random Tree classifier with values 0.16 respectively. Overall, it can be interpreted the most appropriate classifiers based on the performance measure average execution time are found to be Naïve Bayes, SMO, IBK, KStar and Random Tree.

There are Several factors like Algorithm complexity, Data Size, Model Training, Hardware resources, Optimizations, Feature Engineering and Software implementation etc. that can affect the execution time of a machine learning algorithm.

5.3.4 Cross-Validation Configuration Setting (25-Folds) Results

Cross-validation is a valuable technique in machine learning for assessing the performance of predictive models. It involves splitting a dataset into multiple subsets or "folds" to train and test the model on different portions of the data. The number of folds, such as the "25-folds" configuration setting you mentioned, determines how many times this process is repeated. The Weka tool was being used for the analysis of various classification machine learning algorithms. K-folds Cross-Validation Approach is used for evaluating the Performance of Machine learning Algorithms, where K value is changed to study difference cases. In 25 folds Cross-Validation K value is 25 where data set is divided into 25 parts, out of which 24 parts are used for training the Machine learning Algorithm and only One part is used for testing the Algorithm. Twenty five iterations are executed for 25-folds, In first iteration part one will be used for testing and remaining Twenty four parts will be used for training. In second iteration part two will be used for testing and remaining twenty four parts will be used for training. This recursion will continue up to the last iteration to complete the Cross-Validation. Following credentials are used for Data Analysis.

Dataset: Udaipur_Traffic

Source: TOMTOM Server

Date: October 2023

Duration: One Month

Number of Instances: 1000

Number of Attributes (After Feature Extraction and Selection): 7

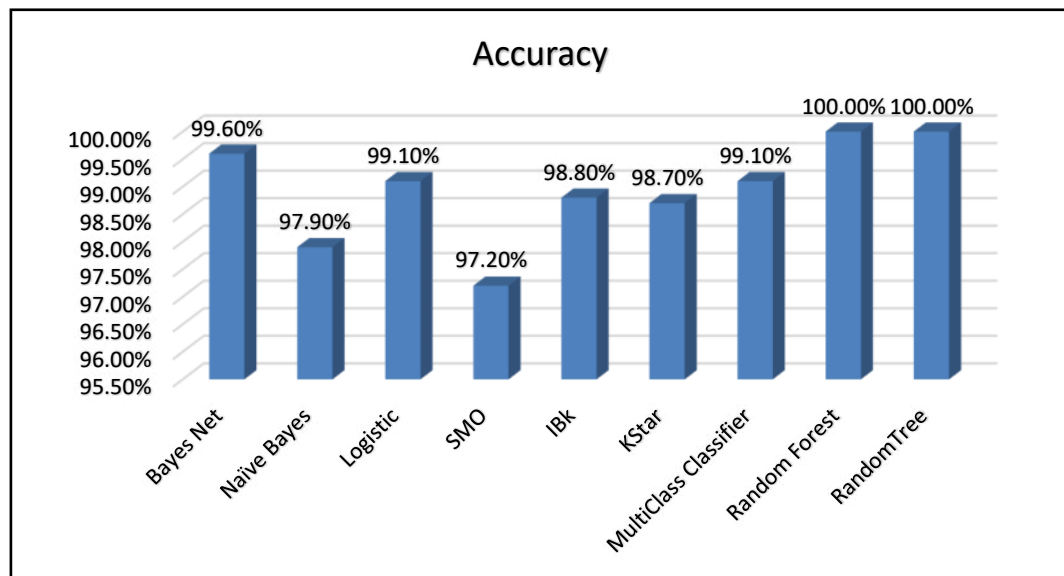
A. Performance Measures

i. Accuracy Measures

Table 5.6: Classifiers and Accuracy Measures (Cross-Validation: 25-Folds)

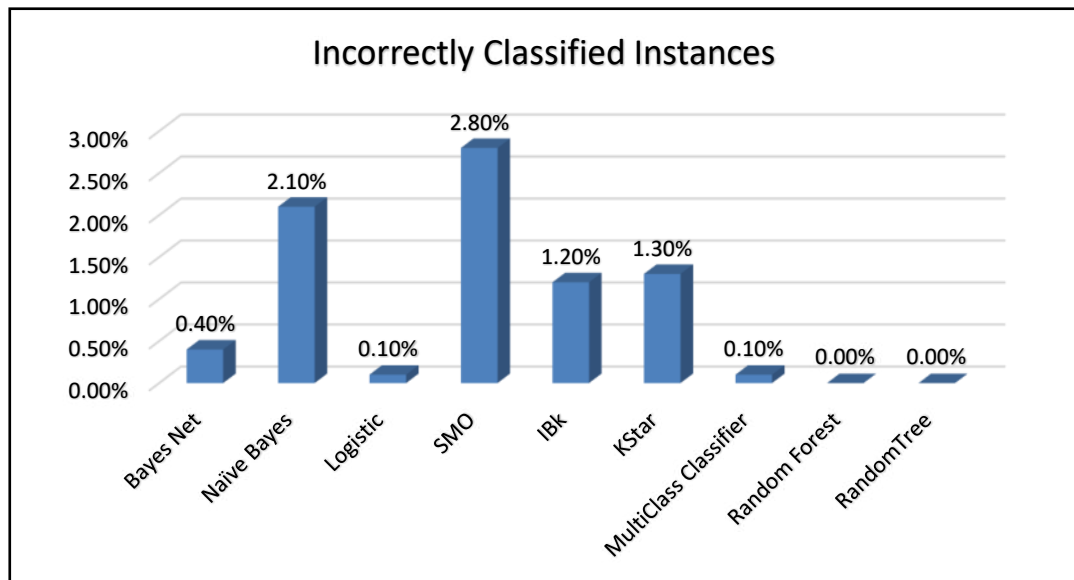
Classifier	Accuracy	Incorrectly Classified Instances	Kappa statistic
Bayes Net	99.60%	0.40%	0.967
Naïve Bayes	97.90%	2.10%	0.818
Logistic	99.10%	0.10%	0.992
SMO	97.20%	2.80%	0.722
IBk	98.80%	1.20%	0.905
KStar	98.70%	1.30%	0.897
MultiClass Classifier	99.10%	0.10%	0.992
Random Forest	100.00%	0.00%	1.000
RandomTree	100.00%	0.00%	1.000

Figure 5.27: Performance Measure Accuracy (Cross-Validation: 25 Folds)



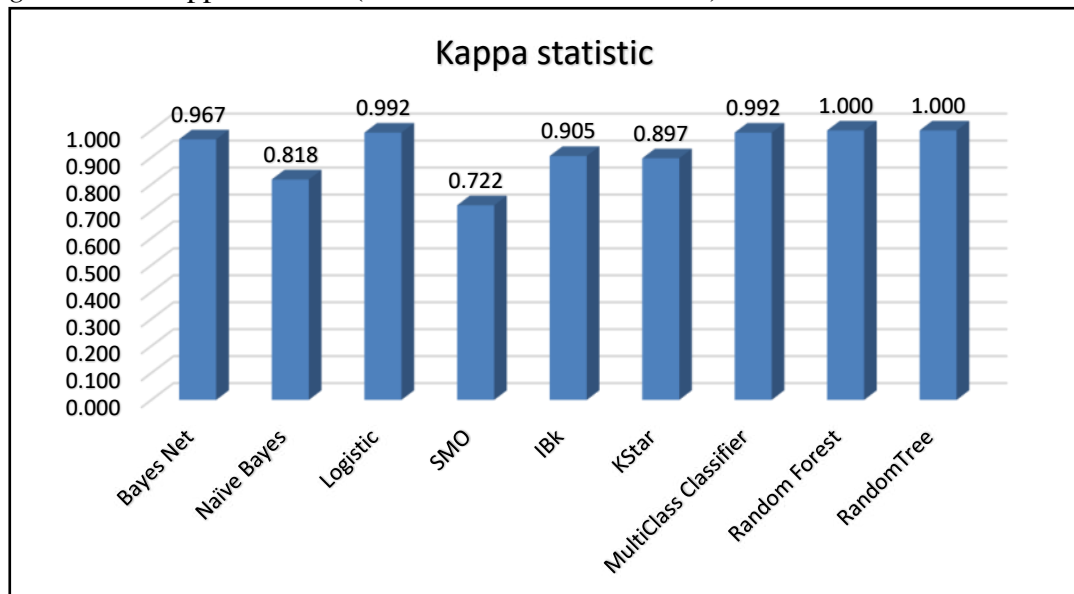
Based on the performance measure accuracy it can be interpreted that Random Forest and Random Tree classifiers were the most appropriate one as they were having the highest accuracy value of 100% whereas classifier Naïve Bayes and SMO were having the lowest value of accuracy 97.90% and 97.20% each.

Figure 5.28: Incorrectly Classified Instances (Cross-Validation: 25 Folds)



According to the performance measure incorrectly classified instances it can be interpreted that Random Forest and Random Tree classifiers were the most appropriate one as these Algorithms were having the lowest number of incorrectly classifies instances, whereas classifiers SMO was having the highest number of incorrectly classified instances.

Figure 5.29: Kappa Statistic (Cross-Validation: 25 Folds)



The provided data consists of a set of Kappa statistic values, which are used to assess the agreement or consistency between classifiers in different situations. These Kappa values range from 0.722 to 1.000, indicating varying levels of agreement. The highest

Kappa value, 1.000, suggests a very good level of agreement between the classifiers in that particular scenario, while the lowest value, 0.722, falls into the good to moderate agreement range. Overall, the data suggests that there is a generally positive level of agreement in the assessed situations, with some instances demonstrating higher agreement than others.

In correctly classified instances and kappa statistics are performance metrics which are important to explore to get more comprehensive insights to develop accurate machine learning model.

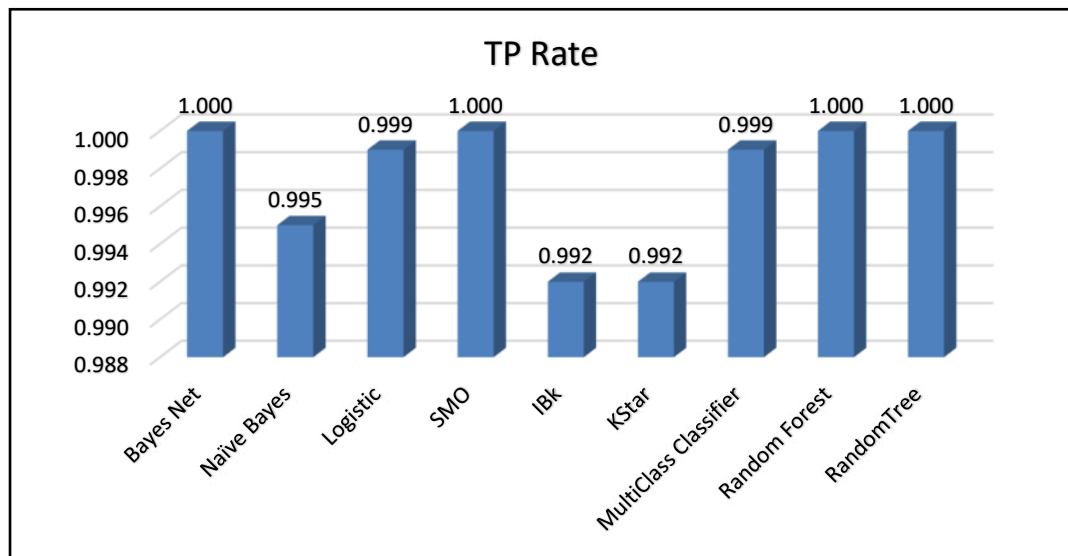
ii. Confusion Matrix Parameters – Low Traffic

Table 5.7 shows confusion matrix parameters TP Rate, FP Rate, Precision, Recall, F-Measure and ROC Area for Low Traffic case. Machine learning often requires a limited amount of data when dealing with low-traffic scenarios. In such cases, the challenge is to create a robust model despite data limitations. Data Augmentation techniques are used to artificially increase the size of data set which can be helpful especially in low traffic scenarios.

*Table 5.7: Classifiers and Performance Measures Class Label: Low Traffic
Cross Validation: 25-Folds*

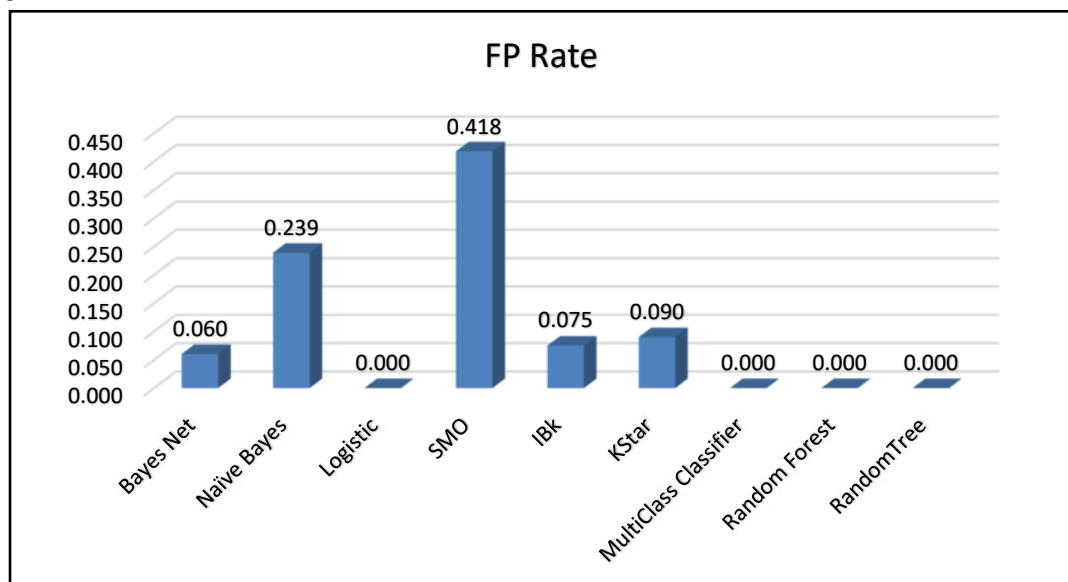
Classifier	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
Bayes Net	1.000	0.060	0.996	1.000	0.998	1.000
Naïve Bayes	0.995	0.239	0.983	0.995	0.989	0.990
Logistic	0.999	0.000	1.000	0.999	0.999	1.000
SMO	1.000	0.418	0.971	1.000	0.985	0.791
IBk	0.992	0.075	0.995	0.992	0.994	0.959
KStar	0.992	0.090	0.994	0.992	0.993	0.997
MultiClass Classifier	0.999	0.000	1.000	0.999	0.999	1.000
Random Forest	1.000	0.000	1.000	1.000	1.000	1.000
RandomTree	1.000	0.000	1.000	1.000	1.000	1.000

Figure 5.30 : TP Rate (Cross Validation: 25-Folds – Low Traffic)



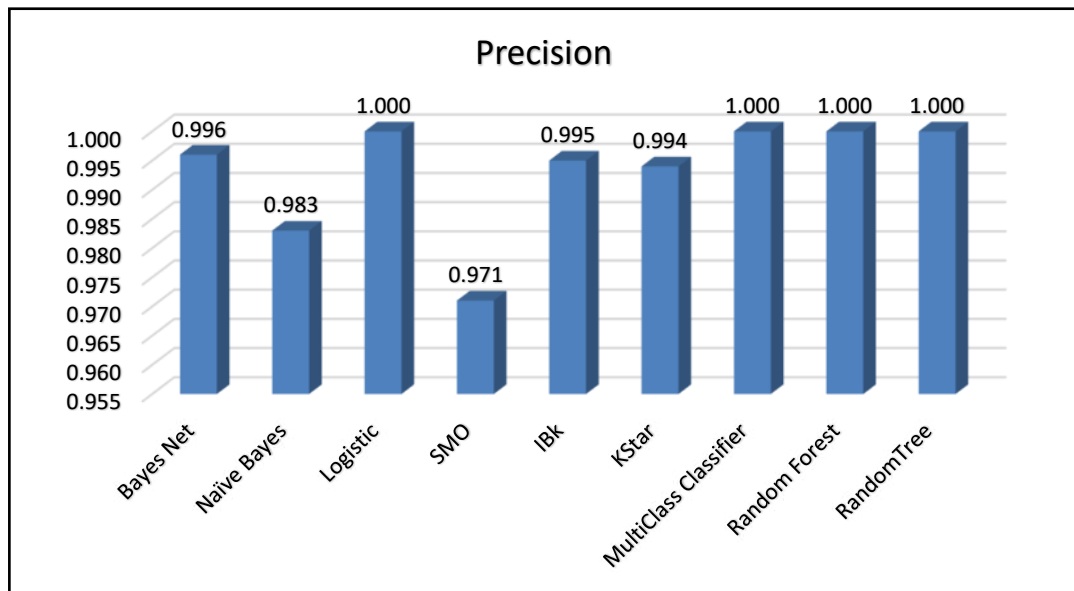
According to the performance measure TP rate for class label: Low Traffic it was found that the highest true positive rate was of the classifiers Bayes Net, SMO, Random Forest and Random Tree with value 1.000, followed by 0.999 of Logistic and MultiClass Classifier, The lowest TP rate was found to be of the classifiers IBK and KStar with value 0.992 respectively.

Figure 5.31 : FP Rate (Cross Validation: 25-Folds – Low Traffic)



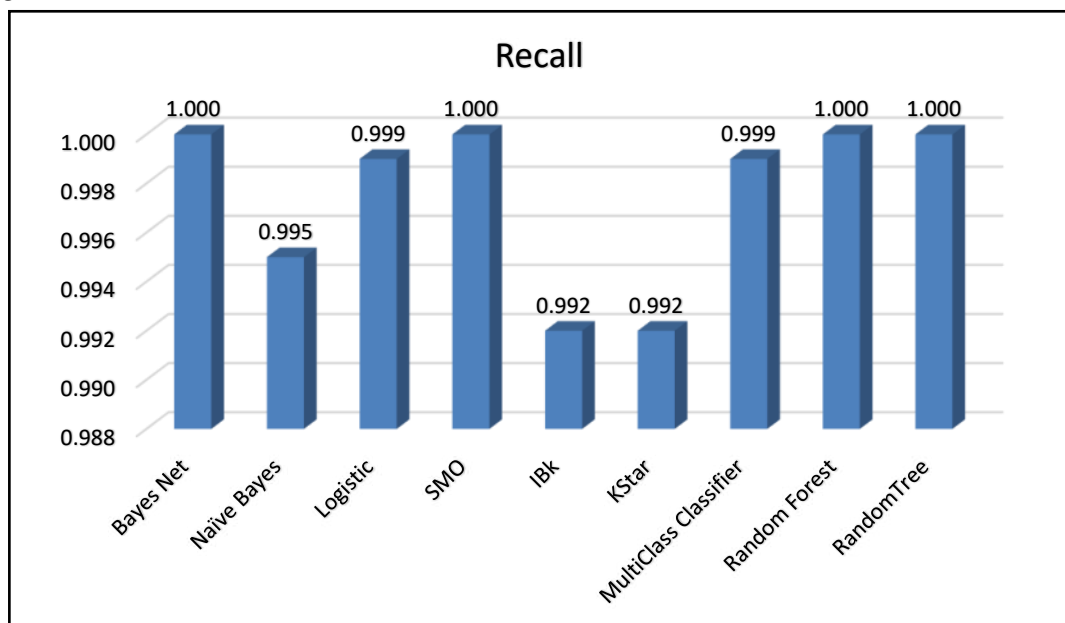
Based on the performance measure FP rate for class label: Low Traffic it was found that the lowest false positive rate was of the classifiers Logistic, MultiClass Classifier, Random Forest and Random Tree with value 0.000, followed by 0.060 of Bayes Net whereas the highest FP rate was found to be of the classifier SMO with value 0.418.

Figure 5.32: Precision (Cross Validation: 25-Folds – Low Traffic)



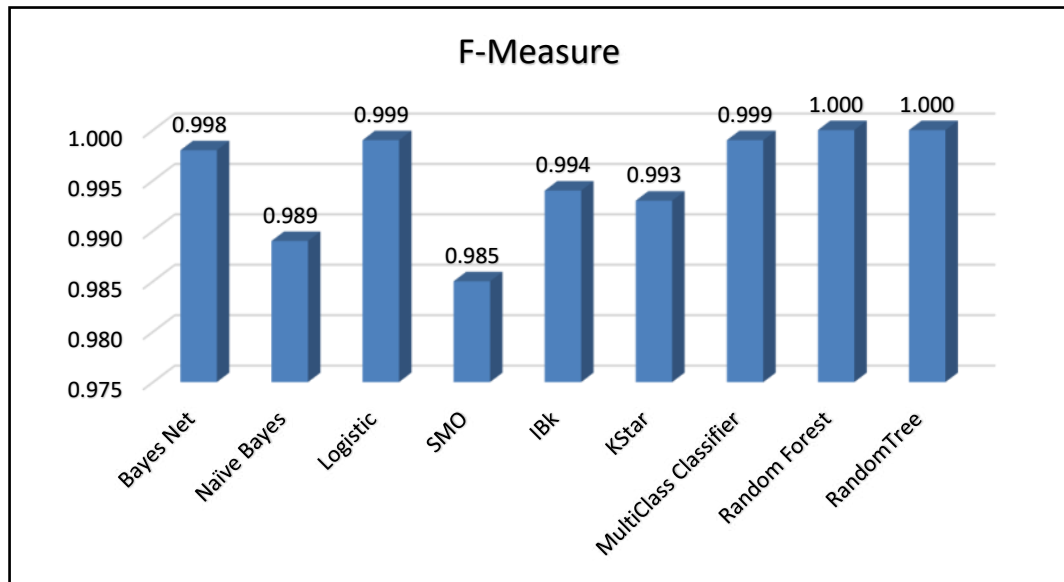
According to the performance measure precision for class label: Low Traffic it was found that the highest precision value was of the classifiers Logistic, MultiClass Classifier, Random Forest and Random Tree with value 1.000, followed by 0.996, 0.995 and 0.994 of Bayes Net, IBK and KStar respectively whereas the lowest precision value was found to be of the classifier SMO with values 0.971 respectively. Overall, it can be interpreted that the most appropriate classifiers based on the performance measure precision are found to be four Algorithms.

Figure 5.33 : Recall (Cross Validation: 25-Folds – Low Traffic)



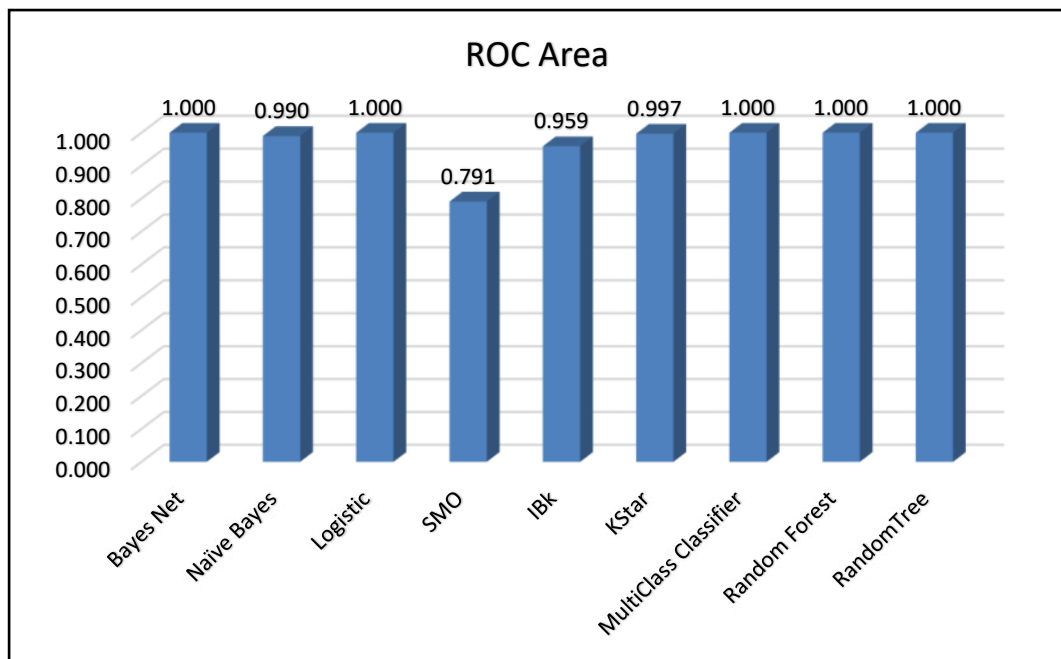
Based on the performance measure recall for class label: Low Traffic it was found that the highest recall value was of the classifiers Bayes Net, SMO, Random Forest and Random Tree with value 1.000, followed by 0.999 of Logistic and MultiClass Classifier respectively whereas the lowest recall value was found to be of the classifiers IBK and KStar with values 0.992 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure recall are found to be Bayes Net, SMO, Random Forest and Random Tree.

Figure 5.34 : F-Measure (Cross Validation: 25-Folds – Low Traffic)



According to the performance measure F-Measure for class label: Low Traffic it was found that the highest F-Measure value was of the classifiers Random Forest and Random Tree with value 1.000, followed by 0.999 and 0.998 of Logistic, MultiClass Classifier and Bayes Net respectively whereas the lowest F-Measure value was found to be of the classifiers SMO with value 0.985. Overall, it can be interpreted that the most appropriate classifier based on the performance measure F-Measure are found to be Random Forest and Random Tree.

Figure 5.35 : ROC Area (Cross Validation: 25-Folds – Low Traffic)



Based on the performance measure ROC for class label: Low Traffic it was found that the highest ROC Area value was of the classifiers Bayes Net, Logistic, MultiClass Classifier, Random Forest and Random Tree with value 1.000, followed by 0.997, 0.990 and 0.959 of KStar, Naïve Bayes and IBK respectively whereas the lowest ROC Area value was found to be of the classifiers SMO with value 0.791 respectively.

In summary, performance measurements play a critical role in evaluating the effectiveness of machine learning algorithms, providing insight into their ability to make accurate predictions and transform appropriately to new, unseen data. Choosing the most appropriate metric depends on the nature of your particular problem, the characteristics of your data, and your analysis goals.

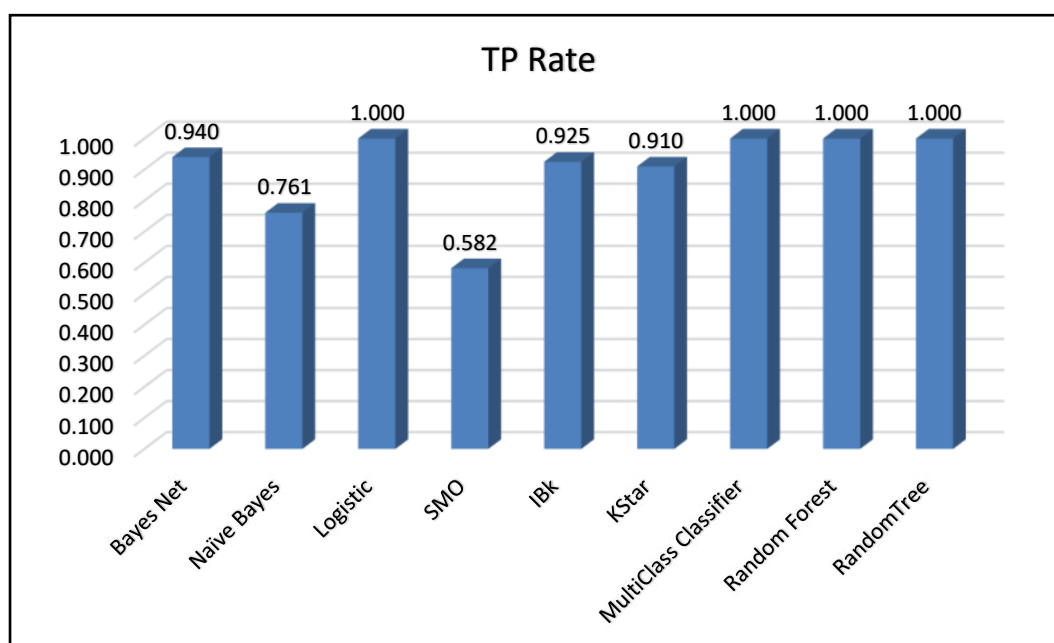
iii. Confusion Matrix Parameters – Heavy Traffic

Heavy Traffic generates huge amounts of data from the various IOT sensors. These data sets can be used by Machine Learning Algorithms to develop prediction models. Table 5.8 shows the Confusion Matrix parameters TP Rate, FP Rate, Precision, Recall, F-Measure and ROC Area obtained for Heavy Traffic conditions.

*Table 5.8: Classifiers Performance Measure Class Label: Heavy Traffic:
Cross Validation: 25-Folds*

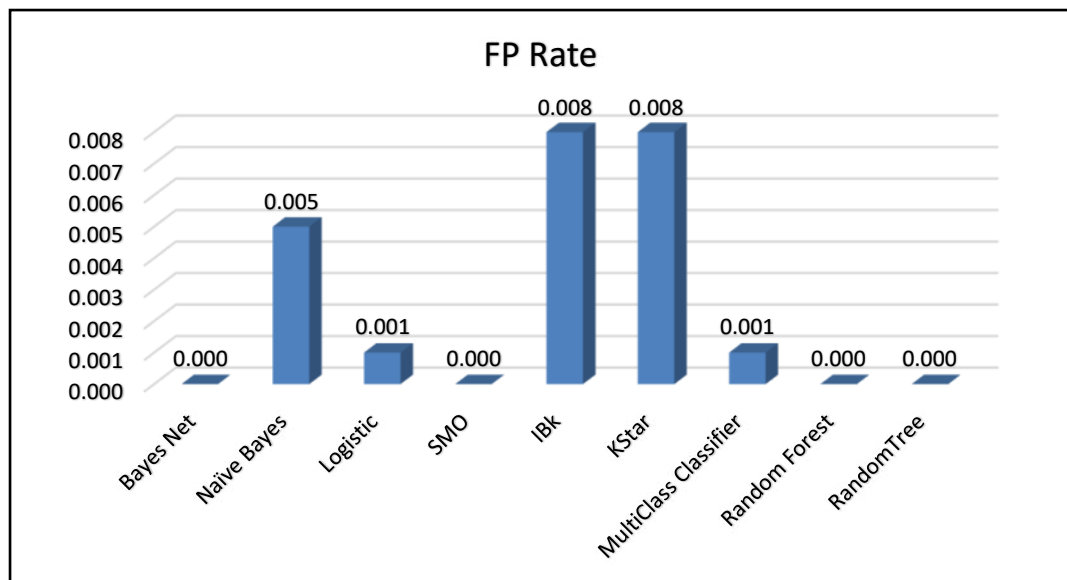
Classifier	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
Bayes Net	0.940	0.000	1.000	0.940	0.969	1.000
Naïve Bayes	0.761	0.005	0.911	0.761	0.829	0.990
Logistic	1.000	0.001	0.985	1.000	0.993	1.000
SMO	0.582	0.000	1.000	0.582	0.736	0.791
IBk	0.925	0.008	0.899	0.925	0.912	0.959
KStar	0.910	0.008	0.897	0.910	0.904	0.997
MultiClass Classifier	1.000	0.001	0.985	1.000	0.993	1.000
Random Forest	1.000	0.000	1.000	1.000	1.000	1.000
RandomTree	1.000	0.000	1.000	1.000	1.000	1.000

Figure 5.36 : TP Rate (Cross Validation: 25-Folds – Heavy Traffic)



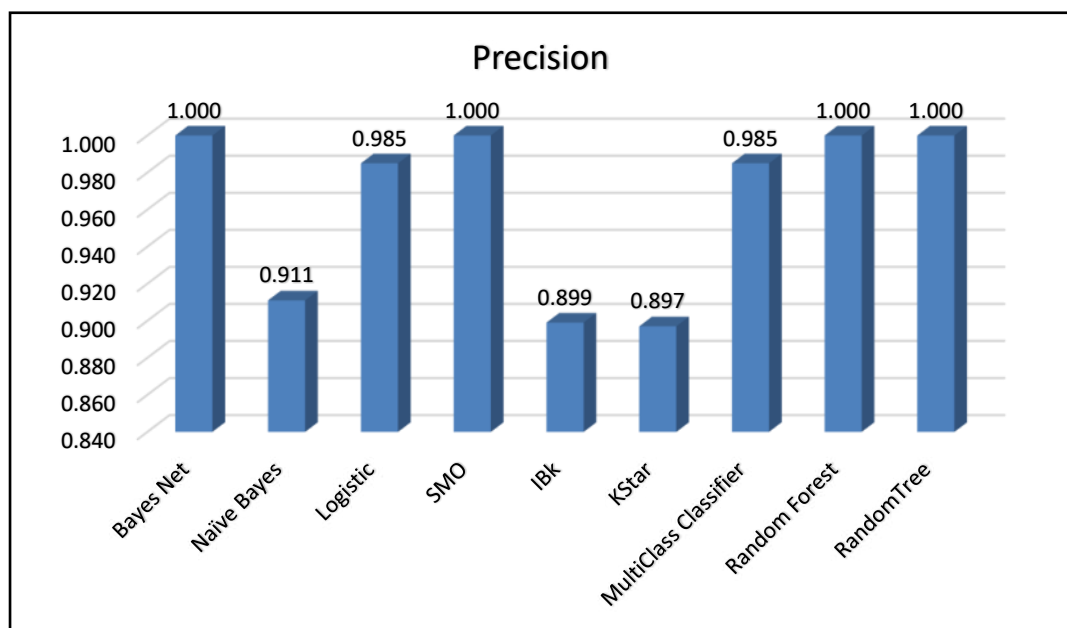
According to the performance measure TP rate for class label: Heavy Traffic it was found that the highest true positive rate was of the classifiers Logistic, MultiClass Classifier, Random Forest and Random Tree with value 1.0, followed by 0.940, 0.925 and 0.910 of Bayes Net, IBK and KStar respectively whereas the lowest TP rate was found to be of the SMO with value 0.582 respectively.

Figure 5.37 : FP Rate (Cross Validation: 25-Folds – Heavy Traffic)



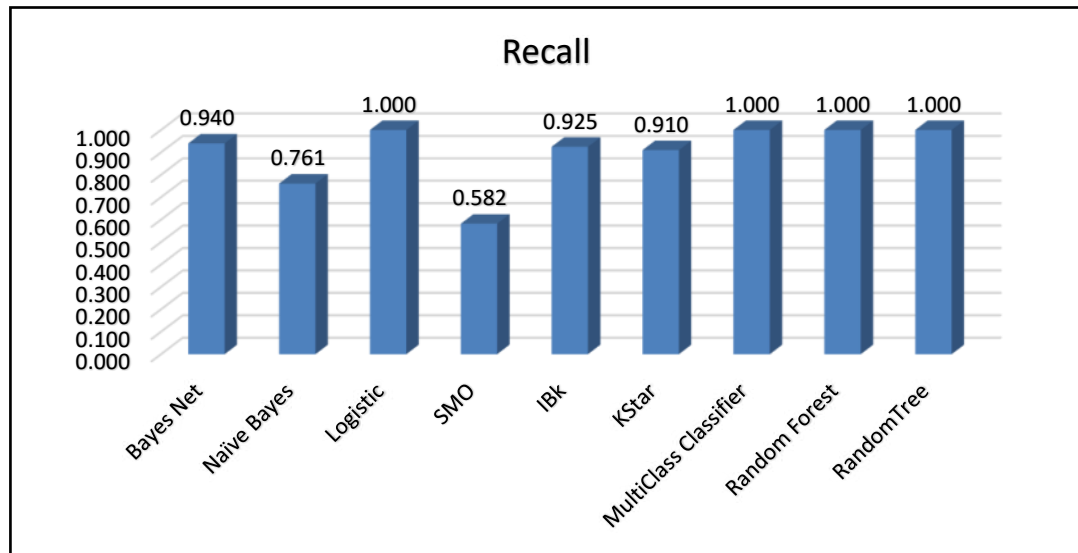
Based on the performance measure FP rate for class label: Heavy Traffic it was found that the lowest false positive rate were of the classifiers Bayes Net, SMO, Random Forest and Random Tree with value 0.00, followed by 0.001 of Logistic and MultiClass Classifier whereas the highest FP rate was found to be of the classifiers IBK and KStar with value 0.008 respectively. Overall, it can be interpreted the most appropriate classifier based on the performance measure FP rate is found to be Bayes Net, SMO, Random Forest and Random Tree with lowest FP rate value.

Figure 5.38: Precision (Cross Validation: 25-Folds – Heavy Traffic)



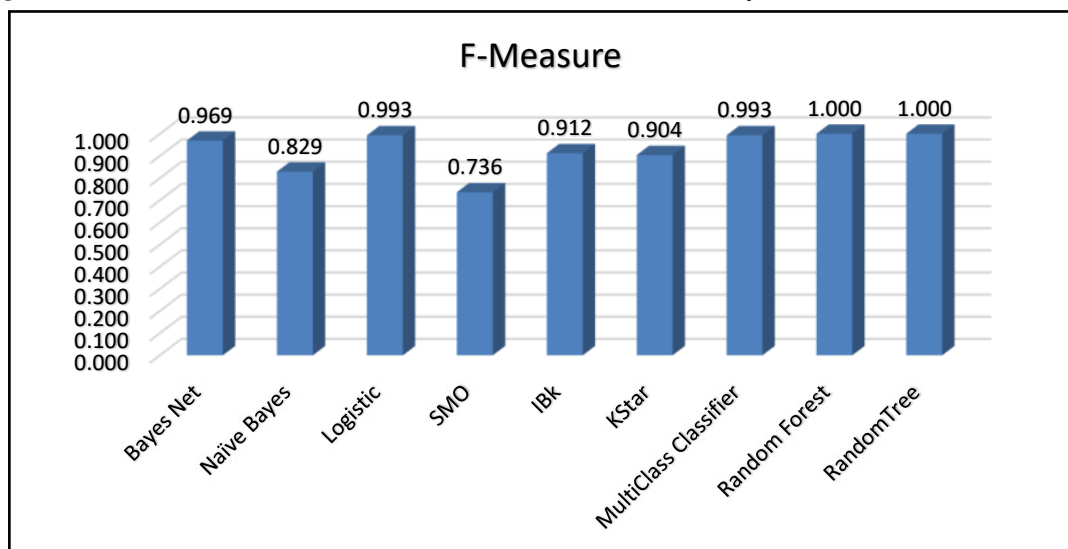
According to the performance measure precision class label: Heavy Traffic it was found that the highest precision value was of the classifiers Bayes Net, SMO, Random Forest and Random Tree with value 1.0, followed by 0.985 of Logistic and MultiClass Classifier respectively whereas the lowest precision values were found to be of the classifiers IBK and KStar with values 0.899 and 0.897 respectively.

Figure 5.39: Recall (Cross Validation: 25-Folds – Heavy Traffic)



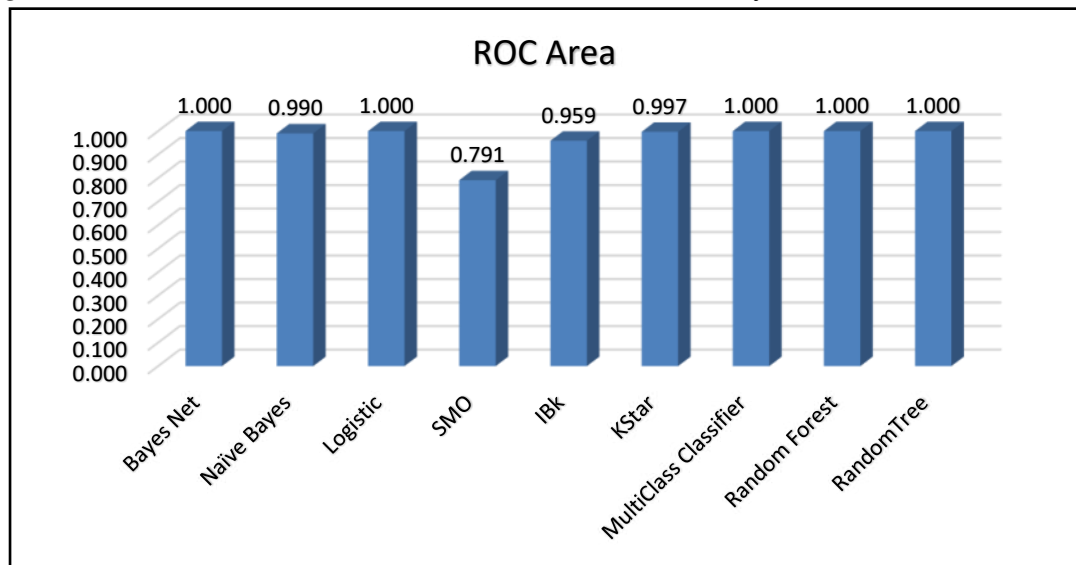
Based on the performance measure recall class label: Heavy Traffic it was found that the highest recall value was of the classifiers Logistic, MultiClass Classifier, Random Forest and Random Tree with value 1.0, followed by 0.940 ,0.925 and 0.910 of Bayes Net, IBK and KStar respectively whereas the lowest recall value was found to be of the classifier SMO with value 0.582 respectively.

Figure 5.40: F-Measure (Cross Validation: 25-Folds – Heavy Traffic)



According to the performance measure F-Measure class label: Heavy Traffic it was found that the highest F-Measure value was of the classifier Random Forest and Random Tree with value 1.0, followed by 0.993 and 0.969 of Logistic, MultiClass Classifier and Bayes Net respectively whereas the lowest F-Measure value was found to be of the classifier SMO with value 0.736. Overall, it can be interpreted that the most appropriate classifier based on the performance measure F-Measure is found to be Random Forest and Random Tree.

Figure 5.41: ROC Area (Cross Validation: 25-Folds – Heavy Traffic)



Based on the performance measure ROC class label: Heavy Traffic it was found that the highest ROC Area value was of the classifiers Bayes Net, Logistic, MultiClass Classifier, Random Forest and Random Tree with value 1.0, followed by 0.997, 0.990 and 0.959 of KStar, Naïve Bayes and IBK respectively whereas the lowest ROC Area value was found to be of the classifier SMO with value 0.791 respectively. Overall, it can be interpreted the most appropriate classifiers based on the performance measure ROC Area are found to be Five Algorithms.

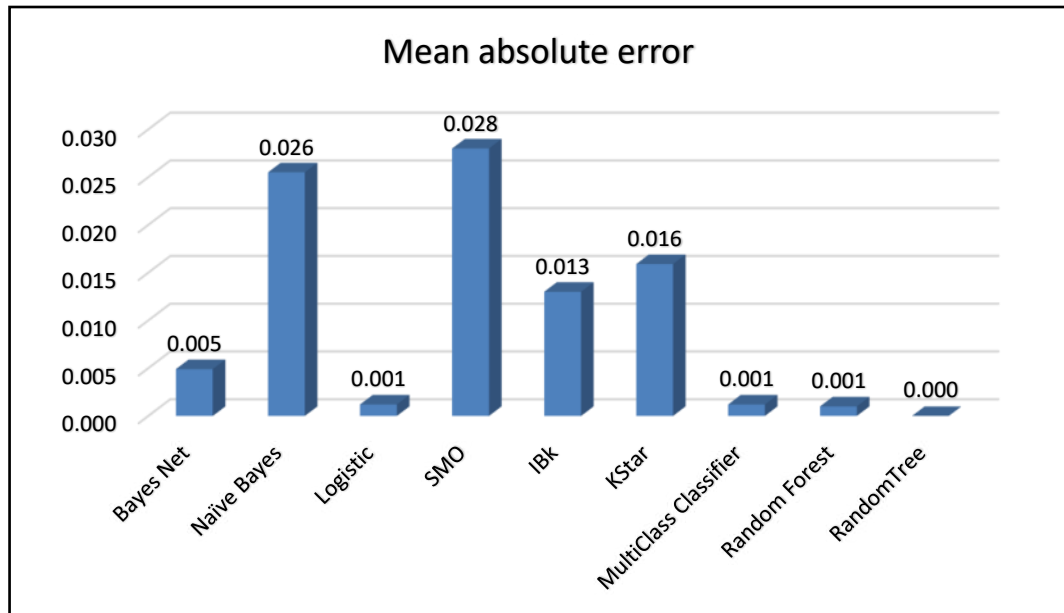
In conclusion the 25 fold Cross validation not only increases the reliability of model but also gives insights of model and explains how model behave under different conditions. More number of folds trains model more accurately to face real life applications and also diminishes the chances of overfitting and underfitting. More number of folds also increases model effectiveness and gives superior model performance, paving the way for more trustworthy and impactful model for real life applications.

B. Error Measure Results

Table 5.9: Classifiers and Error Measures (Cross Validation: 25-Folds)

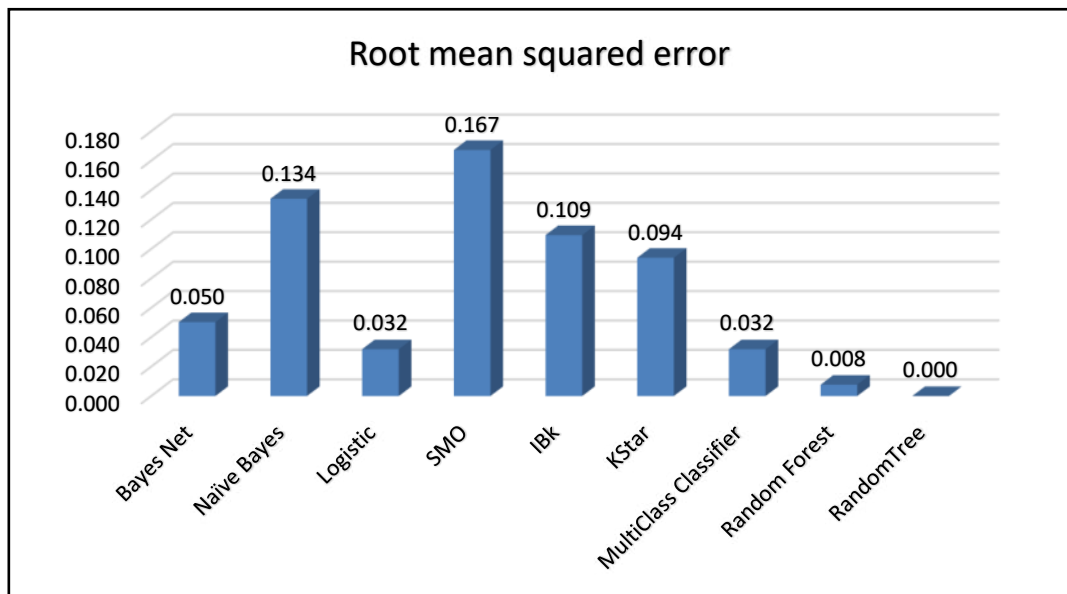
Classifier	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error
Bayes Net	0.005	0.050	3.89%	20.08%
Naïve Bayes	0.026	0.134	20.28%	53.62%
Logistic	0.001	0.032	91.68%	12.75%
SMO	0.028	0.167	22.26%	66.92%
IBk	0.013	0.109	10.34%	43.77%
KStar	0.016	0.094	12.63%	37.59%
MultiClass Classifier	0.001	0.032	0.92%	12.73%
Random Forest	0.001	0.008	0.77%	3.08%
RandomTree	0.000	0.000	0.00%	0.00%

Figure 5.42: Mean Absolute Error (Cross-Validation: 25 Folds)



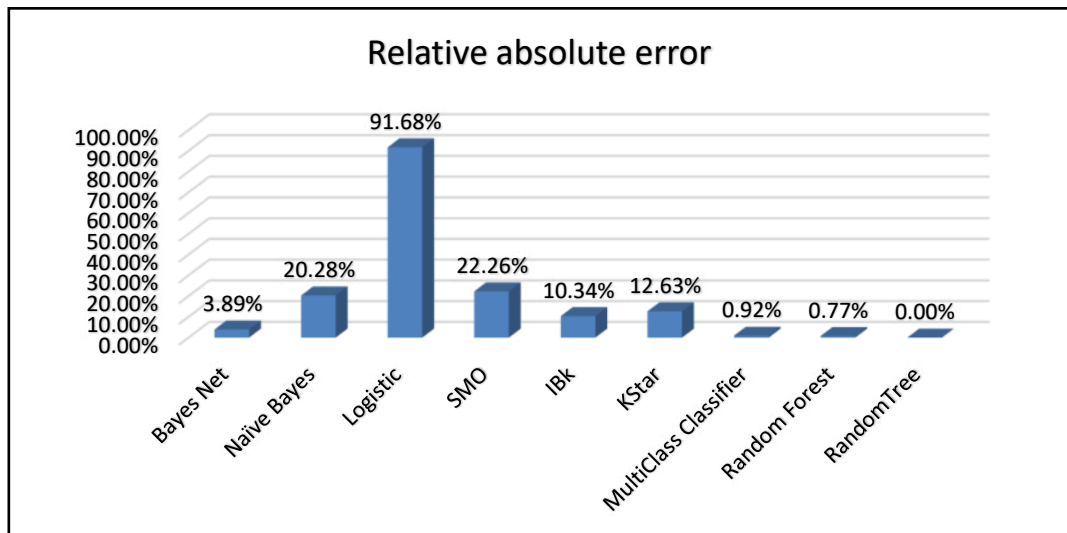
The mean absolute error is found to be lowest in case of Random Tree with the value 0.000. Whereas the mean absolute error value of SMO is found to be highest with value 0.028. So, it can be interpreted that based on the measure Mean absolute Error the most appropriate algorithm is found to be Random Tree at configuration setting – 25-fold cross validation.

Figure 5.43: Root Mean Squared Error (Cross-Validation: 25 Folds)



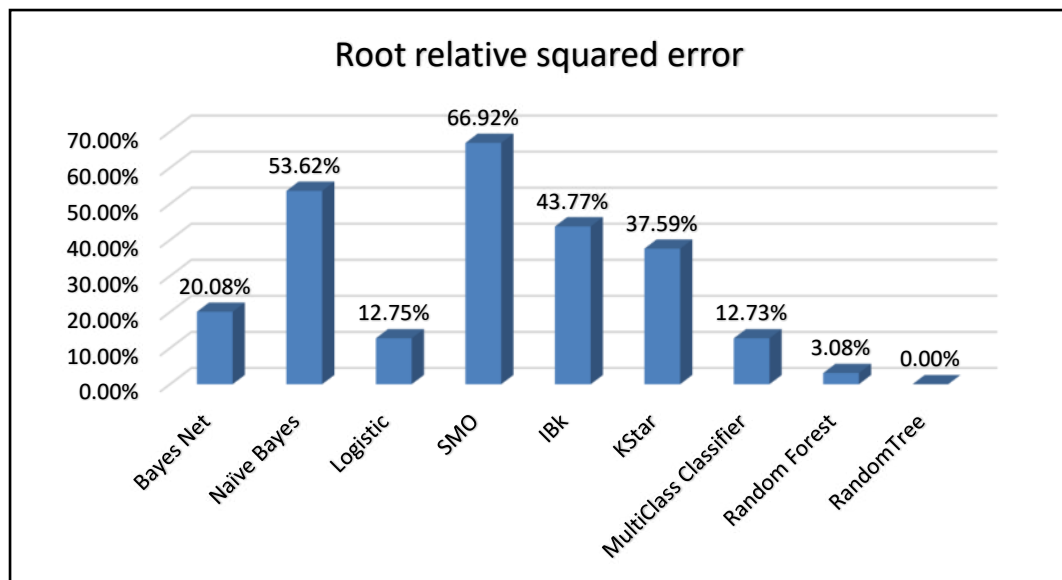
The root mean squared error value is found to be highest in case of SMO with the value of 0.167 whereas the lowest value is found to be of Random Tree with value 0.000. So, it can be interpreted that based on the measure RMSE the most appropriate algorithm is found to be Random Tree at configuration setting – 25-fold cross validation.

Figure 5.44: Relative Absolute Error (Cross Validation: 25-Folds)



Accordingly, the relative absolute error value is found to be lowest in case of Random Tree classifier with 0.00% whereas the highest relative absolute error percentage value is found to be in case of Logistic with 91.68%. So, it can be suggested that based on the measure Relative Absolute Error the most appropriate algorithm is found to be Random Tree with lowest value when evaluated at configuration setting – 25-fold cross validation.

Figure 5.45: Root Relative Squared Error (Cross Validation: 25-Folds)



The root relative squared error value is found to be lowest in case of Random Tree classifier with 0.00% whereas the root relative squared error percentage value is found to be highest in case of SMO with percentage value of 66.92%. So, it can be interpreted that based on the measure RRSE the most appropriate algorithm is found to be Random Tree with lowest percentage value when evaluated at configuration setting – 25-fold cross validation.

In summary, it is important to understand error measures in Machine learning for assessing the performance of model properly and making informed decisions. The choice of specific metrics depends on the nature of the problem, characteristics of the dataset, and goals of the analysis.

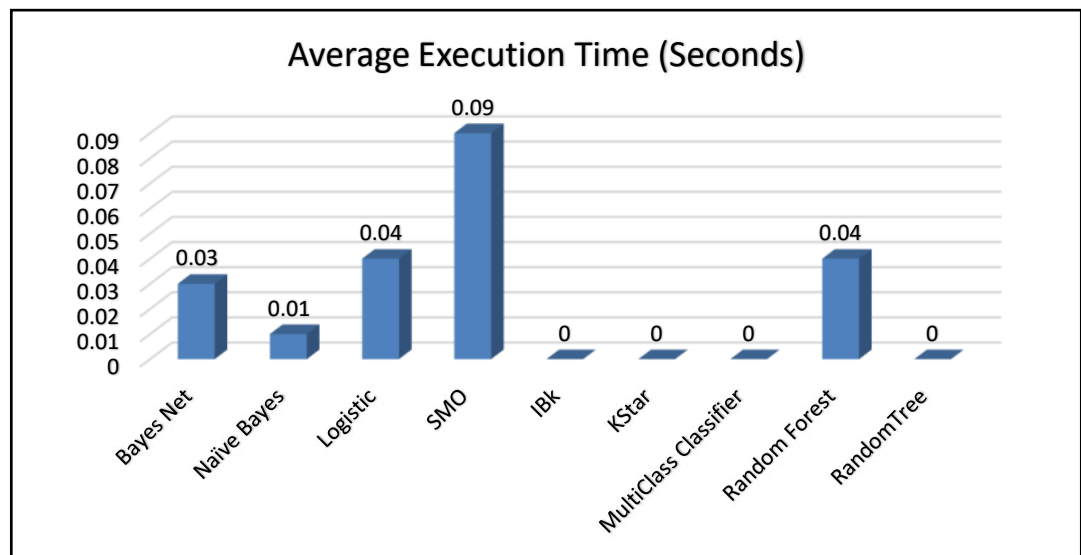
C. Execution Time Results

The execution time of a machine learning algorithm refers to the time it takes for the algorithm to process and analyse input data, train the model (if applicable), and produce predictions or results. Execution time is an important factor when evaluating the efficiency and scalability of machine learning algorithms, especially when dealing with large data sets and real-time applications. The Average Execution Time of Nine Classifier Algorithm for Cross Validation 25 fold is given below.

Table 5.10: Classifiers and Average Execution Time (Cross Validation: 25-Folds)

Classifier	Average Execution Time (Seconds)
Bayes Net	0.03
Naïve Bayes	0.01
Logistic	0.04
SMO	0.09
IBk	0
KStar	0
MultiClass Classifier	0
Random Forest	0.04
RandomTree	0

Figure 5.46 : Average Execution Time (Cross Validation: 25-Folds)

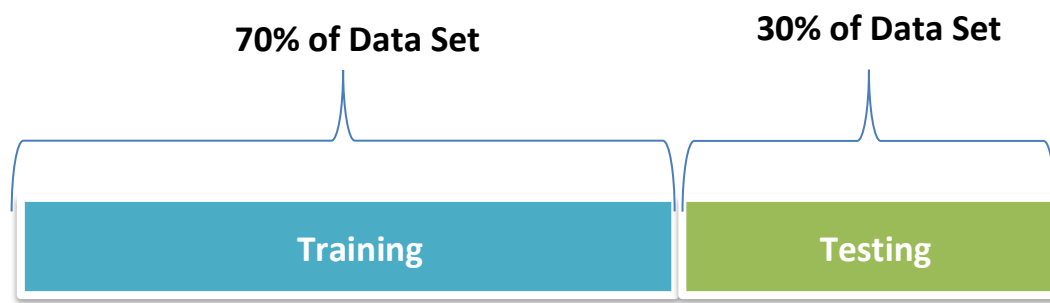


According to the performance measure average execution time it was found that the lowest average execution time were of the classifiers IBK, KStar, MultiClass Classifier and Random Tree with values 0.0 each whereas the highest average execution time was found to be of the classifier SMO with values 0.09 respectively. Overall, it can be interpreted that the most appropriate classifiers based on the performance measure average execution time are found to be IBK, KStar, MultiClass Classifier and Random Tree.

5.3.5 Cross-Validation: Configuration Setting (30% Split) Results

In the context of cross-validation, the "30% Split" configuration setting typically refers to a technique called "holdout validation" or "simple validation." It involves splitting the dataset into two portions: one for training the machine learning model and another for testing its performance. Here 70% of Data set is used for training the machine learning model and 30% is used for testing the Model. Testing is basically used to evaluate the model based on various metrics. Confusion Matrix can be used to evaluate the final performance of the Selected Machine learning models.

Figure 5.47 : Data Set Split



As shown in the Figure 5.47 the entire data set is randomly partitioned into Training set and Testing set. Since The data set is split into only two set, therefore it is constructed very fast on training data and executed for testing very fast. Following credentials are used for Data Analysis.

Dataset: Udaipur_Traffic

Source: TOMTOM Server

Date: October 2023

Duration: One Month

Number of Instances: 1000

Number of Attributes (After Feature Extraction and Selection): 7

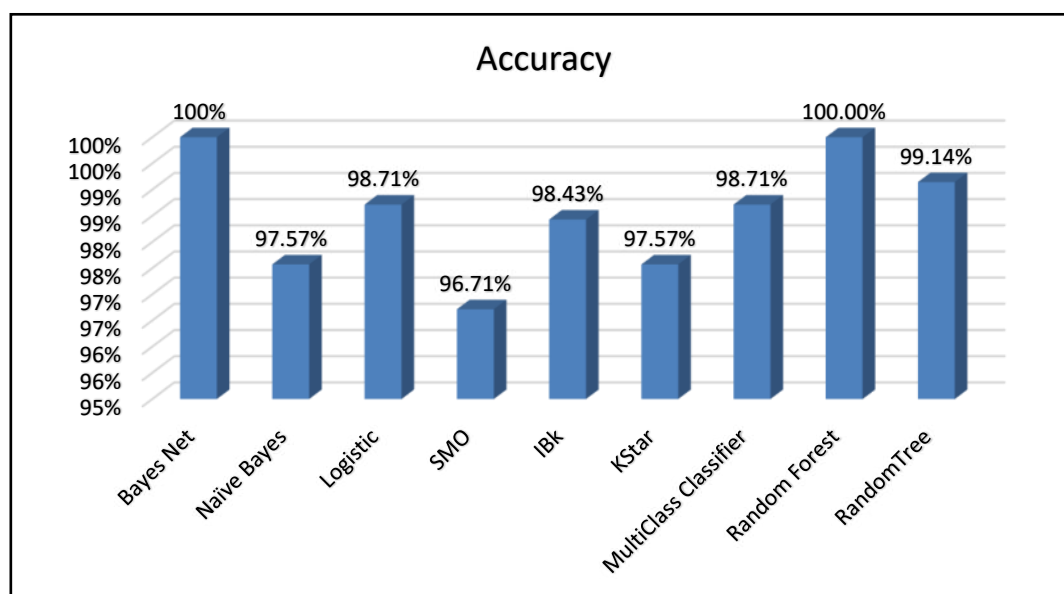
A. Performance Measures

i. Accuracy Measures

Table 5.11: Classifiers and Accuracy Measures (Cross-Validation:30% Split)

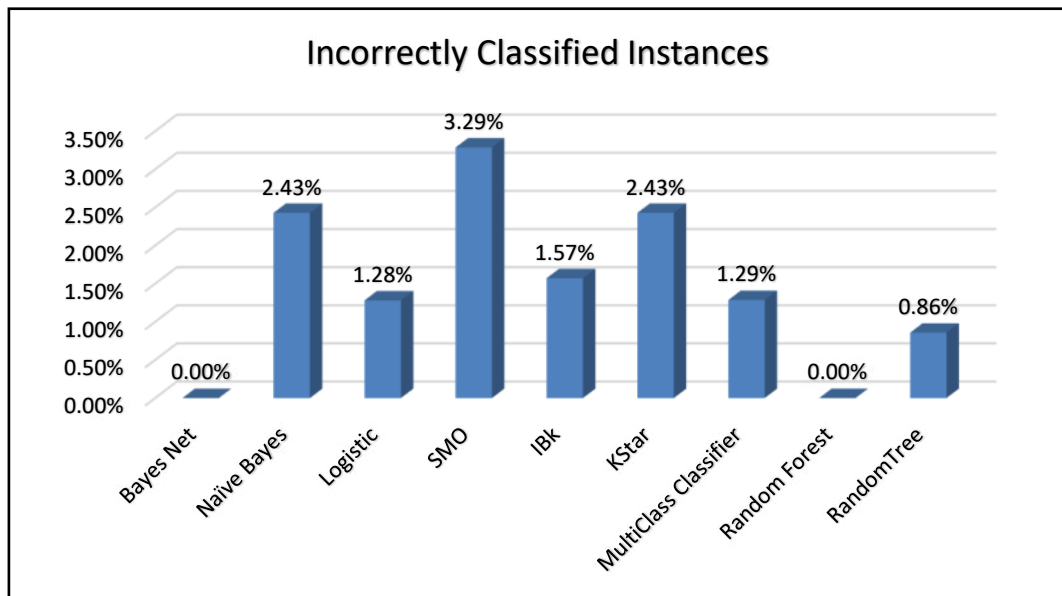
Classifier	Accuracy	Incorrectly Classified Instances	Kappa Statistic
Bayes Net	100%	0.00%	0.010
Naïve Bayes	97.57%	2.43%	0.008
Logistic	98.71%	1.28%	0.009
SMO	96.71%	3.29%	0.707
IBk	98.43%	1.57%	0.881
KStar	97.57%	2.43%	0.797
MultiClass Classifier	98.71%	1.29%	0.900
Random Forest	100.00%	0.00%	1.000
RandomTree	99.14%	0.86%	0.935

Figure 5.48 : Performance Measure Accuracy (Cross-Validation: 30% Split)



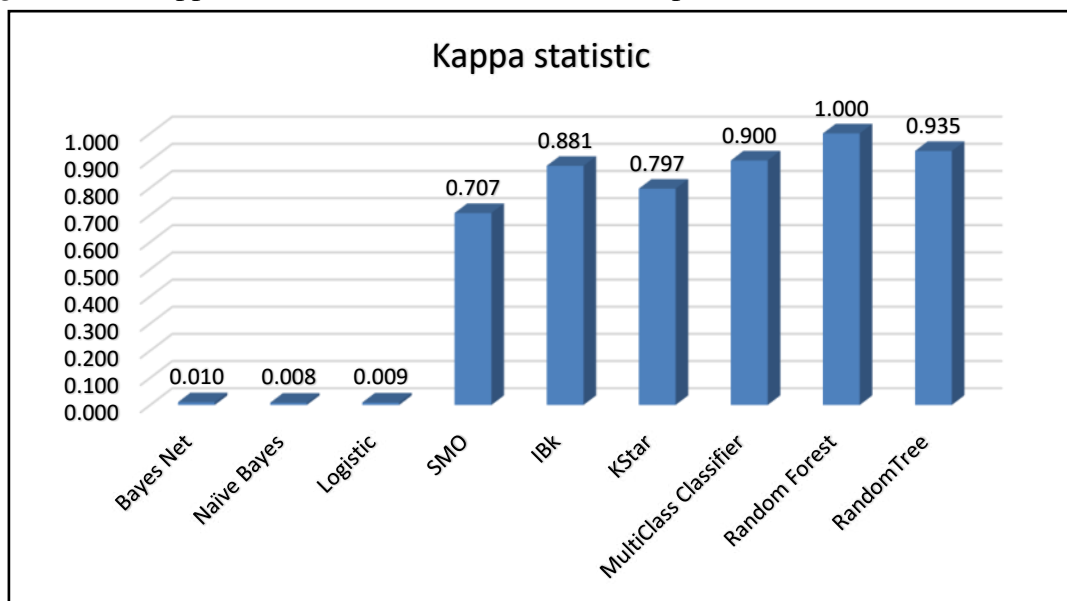
Based on the performance measure accuracy it can be interpreted that Bayes Net and Random Forest classifiers were the most appropriate one as they were having the highest accuracy value of 100% whereas classifier SMO was having the lowest value of accuracy 96.71% respectively.

Figure 5.49: Incorrectly Classified Instances (Cross-Validation: 30% Split)



According to the performance measure incorrectly classified instances it can be interpreted that Bayes Net and Random Forest classifier were the most appropriate one as they were having the lowest number of incorrectly classified instances accounting for 0% whereas classifier SMO classifier was having the highest number of incorrectly classifies instances accounting as 3.29% respectively.

Figure 5.50: Kappa Statistic (Cross-Validation: 30% Split)



The provided data consists of a set of Kappa statistic values, which are used to assess the agreement or consistency between classifiers in different situations. These Kappa values range from 0.008 to 1.000, indicating varying levels of agreement. The highest

Kappa value, 1.000, suggests a very good level of agreement between the classifiers in that particular scenario, while the lowest value, 0.008, falls into the poor agreement range.

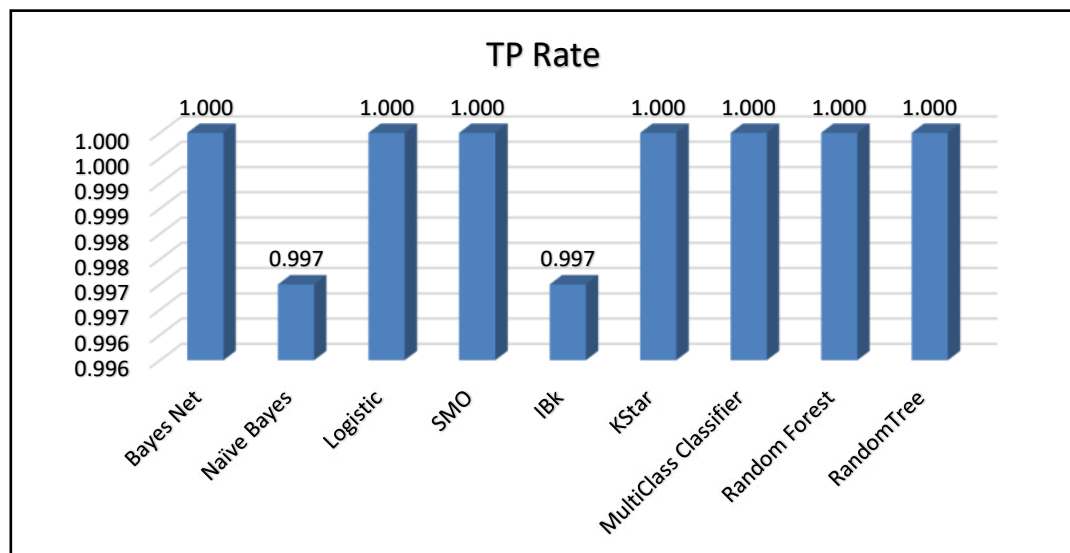
ii. Confusion Matrix Parameters – Low Traffic

Machine learning often requires a limited amount of data when dealing with low-traffic scenarios. In such cases, the challenge is to create a robust model despite data limitations. Data Augmentation techniques are used to artificially increase the size of data set which can be helpful especially in low traffic scenarios.

*Table 5.12: Classifiers Performance Measures Class Label: Low Traffic:
Cross Validation: 30% Split*

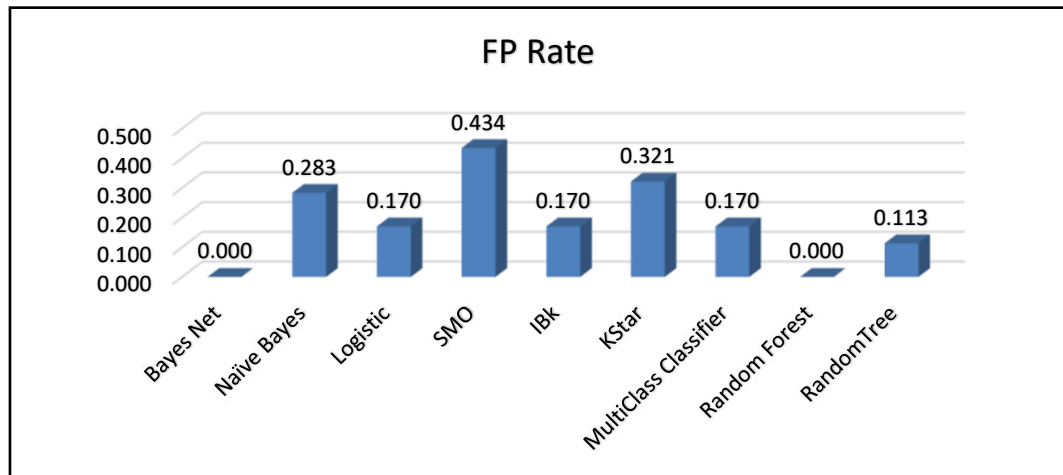
Classifier	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
Bayes Net	1.000	0.000	1.000	1.000	1.000	1.000
Naïve Bayes	0.997	0.283	0.977	0.997	0.987	0.990
Logistic	1.000	0.170	0.986	1.000	0.993	0.990
SMO	1.000	0.434	0.966	1.000	0.983	0.783
IBk	0.997	0.170	0.986	0.997	0.992	0.914
KStar	1.000	0.321	0.974	1.000	0.987	0.997
MultiClass Classifier	1.000	0.170	0.986	1.000	0.993	0.990
Random Forest	1.000	0.000	1.000	1.000	1.000	1.000
RandomTree	1.000	0.113	0.991	1.000	0.995	0.943

Figure 5.51: TP Rate (Cross-Validation: 30% Split)



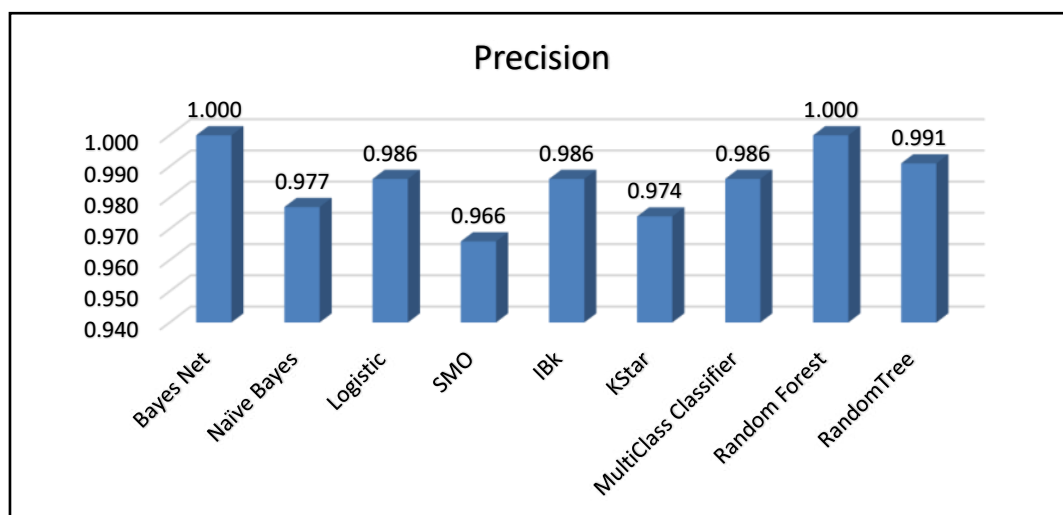
According to the performance measure TP rate it was found that the highest true positive rate was of the classifiers Bayes Net, Logistic, SMO, KStar, MultiClass Classifier, Random Forest and Random Tree with value 1.0, whereas the lowest TP rate was found to be of the classifiers Naïve Bayes and IBK with values 0.997 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure TP rate are seven Algorithms.

Figure 5.52: FP Rate (Cross-Validation: 30% Split)



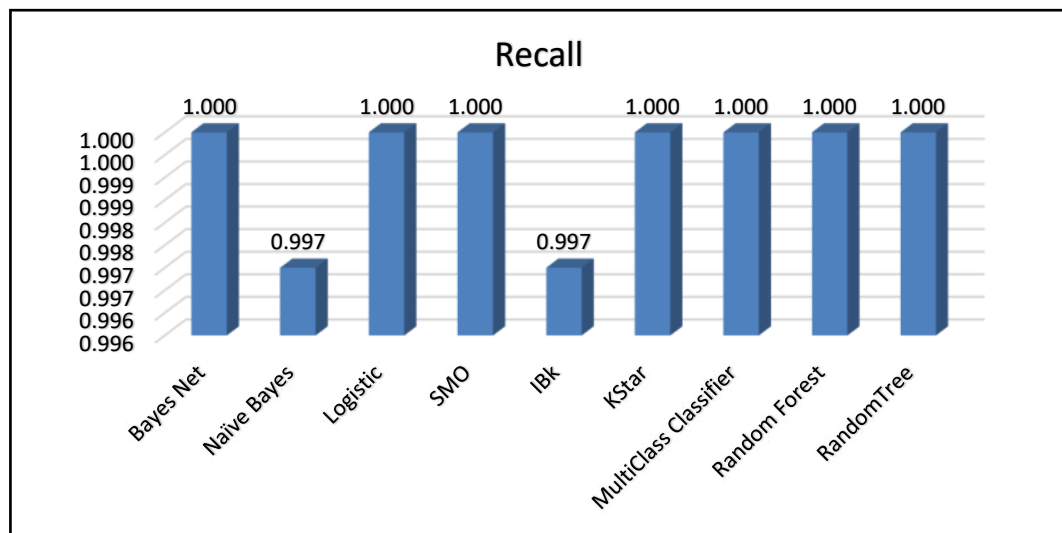
Based on the performance measure FP rate it was found that the lowest false positive rate was of the classifiers Bayes Net and Random Forest with value 0.000, whereas the highest FP rate was found to be of the classifier SMO with value 0.434 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure FP rate is found to be Bayes Net and Random Forest with lowest FP rate value.

Figure 5.53: Precision (Cross-Validation: 30% Split)



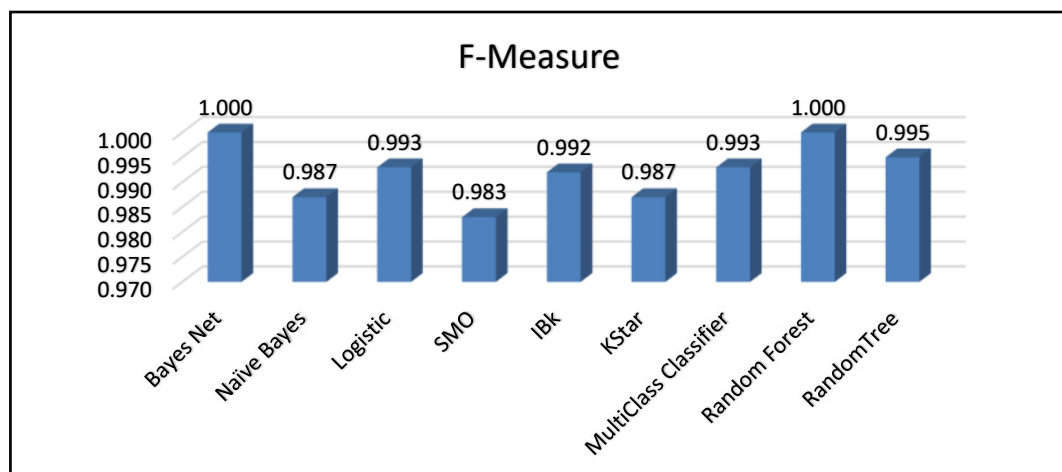
According to the performance measure precision it was found that the highest precision value was of the classifier Bayes Net and Random Forest with value 1.0, followed by 0.991 of Random Tree respectively whereas the lowest precision value was found to be of the classifiers SMO with values 0.966 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure precision is found to be the Bayes Net and Random Forest.

Figure 5.54: Recall (Cross-Validation: 30% Split)



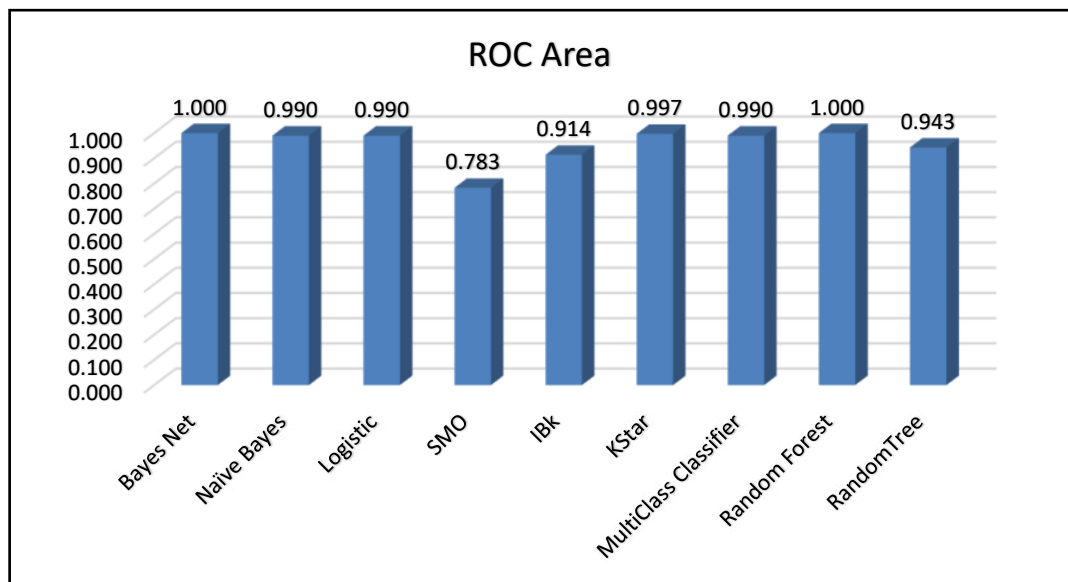
Based on the performance measure recall it was found that the highest recall value was of the classifiers Bayes Net, Logistic, SMO, KStar, MultiClass Classifier, Random Forest and Random Tree with value 1.0, whereas the lowest recall value was found to be of the classifiers Naïve Bayes and IBK with values 0.997 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure recall is found to be Seven Algorithms.

Figure 5.55: F-Measure (Cross-Validation: 30% Split)



According to the performance measure F-Measure it was found that the highest F-Measure value was of the classifiers Bayes Net and Random Forest with value 1.00, followed by 0.995 of Random Tree respectively whereas the lowest F-Measure value was found to be of the classifiers SMO classifier with values 0.983 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure F-Measure is found to be Bayes Net and Random Forest.

Figure 5.56 : ROC Area (Cross-Validation: 30% Split)



Based on the performance measure ROC it was found that the highest ROC Area value was of the classifiers Bayes Net and Random Forest with value 1.0, followed by 0.997 of KStar respectively whereas the lowest ROC Area value was found to be of the classifiers SMO with values 0.783 respectively. Overall, it can be interpreted the most appropriate classifier based on the performance measure ROC Area is found to be Bayes Net and Random Forest.

In summary, performance measurements play a critical role in evaluating the effectiveness of machine learning algorithms, providing insight into their ability to make accurate predictions and transform appropriately to new, unseen data. Choosing the most appropriate metric depends on the nature of your problem, the characteristics of your data, and your analysis goals.

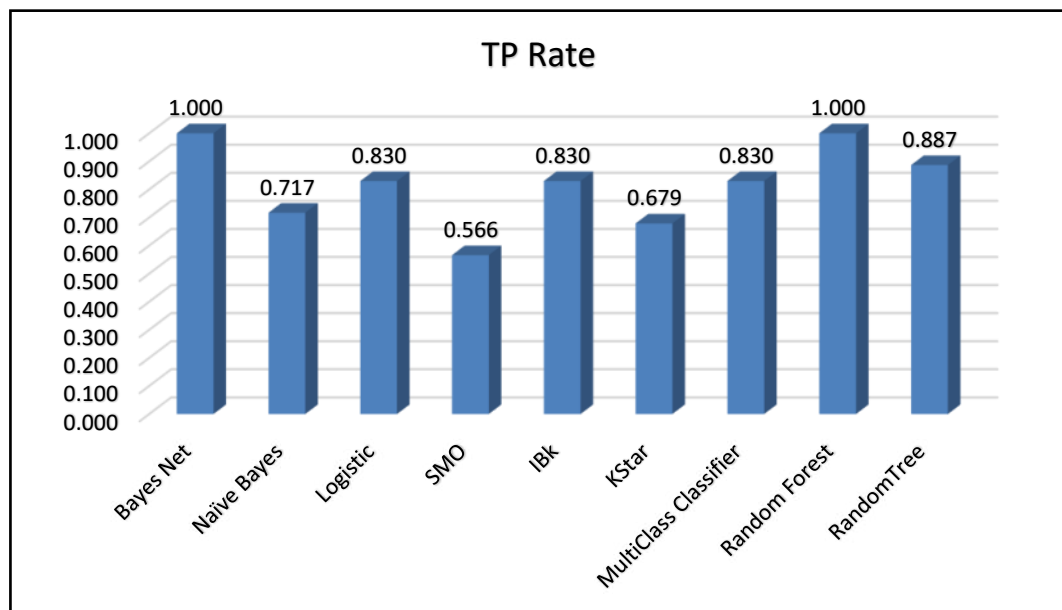
iii. Confusion Matrix Parameters – Heavy Traffic

Heavy Traffic generates huge amounts of data from the various IOT sensors. These data sets can be used by Machine Learning Algorithms to develop prediction models. Table 5.13 shows the Confusion Matrix parameters TP Rate, FP Rate, Precision, Recall, F-Measure and ROC Area obtained for Heavy Traffic conditions.

*Table 5.13: Classifiers Performance Measure Class Label: Heavy Traffic:
Cross Validation: 30% Split*

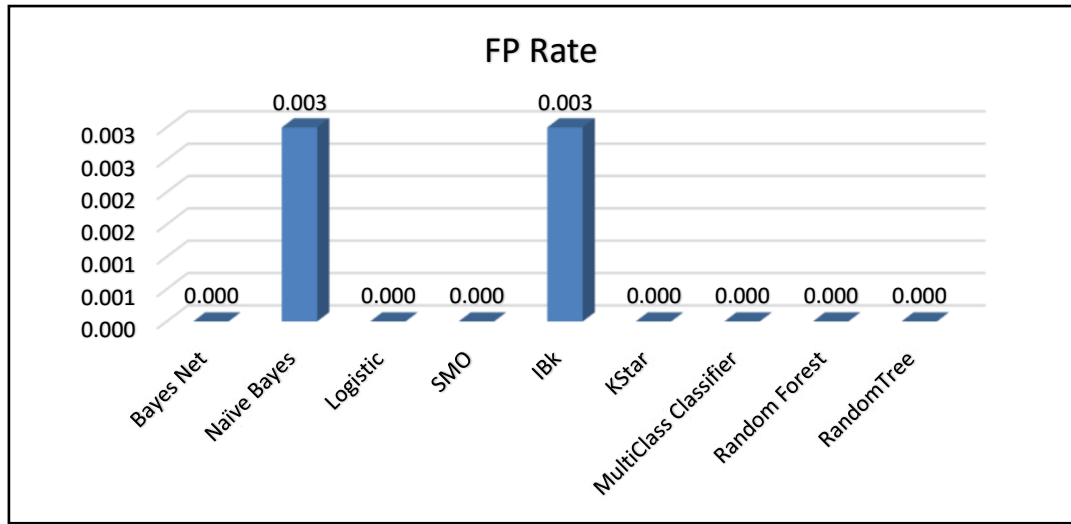
Classifier	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
Bayes Net	1.000	0.000	1.000	1.000	1.000	1.000
Naïve Bayes	0.717	0.003	0.950	0.717	0.817	0.990
Logistic	0.830	0.000	1.000	0.830	0.907	0.999
SMO	0.566	0.000	1.000	0.566	0.723	0.783
IBk	0.830	0.003	0.957	0.830	0.889	0.914
KStar	0.679	0.000	1.000	0.679	0.809	0.097
MultiClass Classifier	0.830	0.000	1.000	0.830	0.907	0.999
Random Forest	1.000	0.000	1.000	1.000	1.000	1.000
RandomTree	0.887	0.000	1.000	0.887	0.940	0.943

Figure 5.57: TP Rate (Cross-Validation: 30% Split)



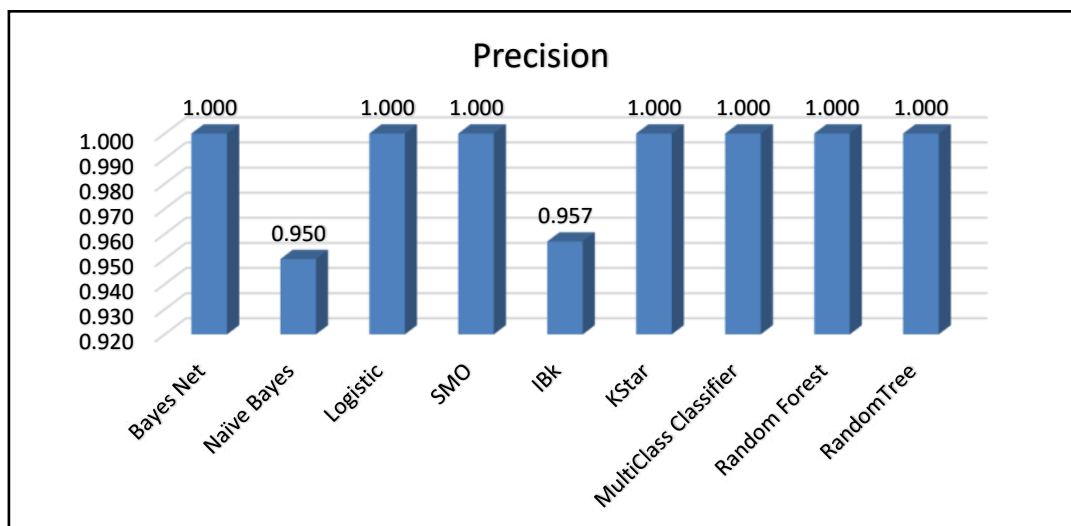
According to the performance measure TP rate for class label: Heavy Traffic it was found that the highest true positive rates were of the classifier Bayes Net and Random Forest with value 1.0, followed by 0.887 of Random Tree respectively whereas the lowest TP rate was found to be of the classifiers SMO with value 0.566 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure TP rate is Bayes Net and Random Forest.

Figure 5.58: FP Rate (Cross-Validation: 30% Split)



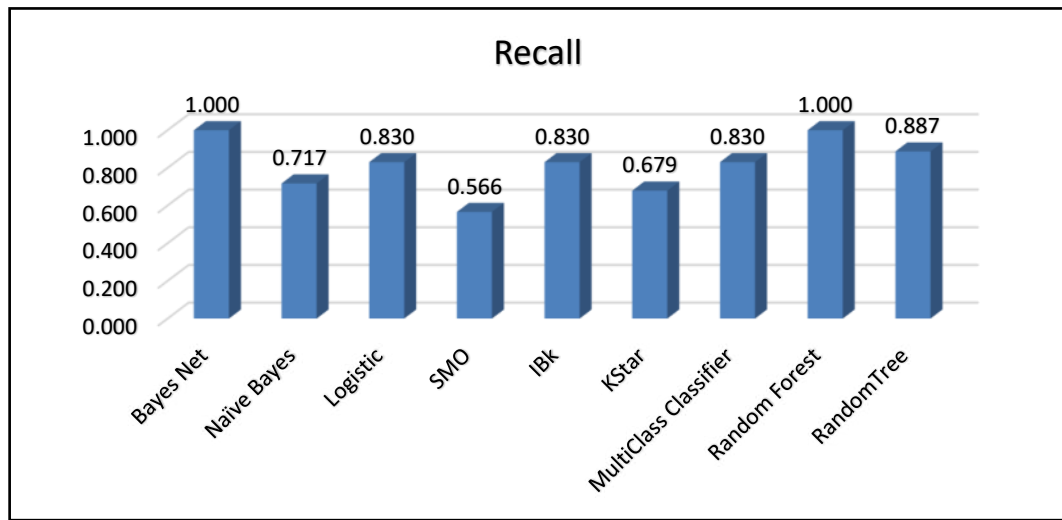
Based on the performance measure FP rate for class label: Heavy Traffic it was found that the lowest false positive rates were of the classifier Bayes Net, Logistic, SMO, KStar, MultiClass Classifier, Random Forest, and Random Tree with value 0.00 each, whereas the highest FP rate was found to be of the classifiers Naïve Bayes and IBK with values 0.003.

Figure 5.59 : Precision (Cross-Validation: 30% Split)



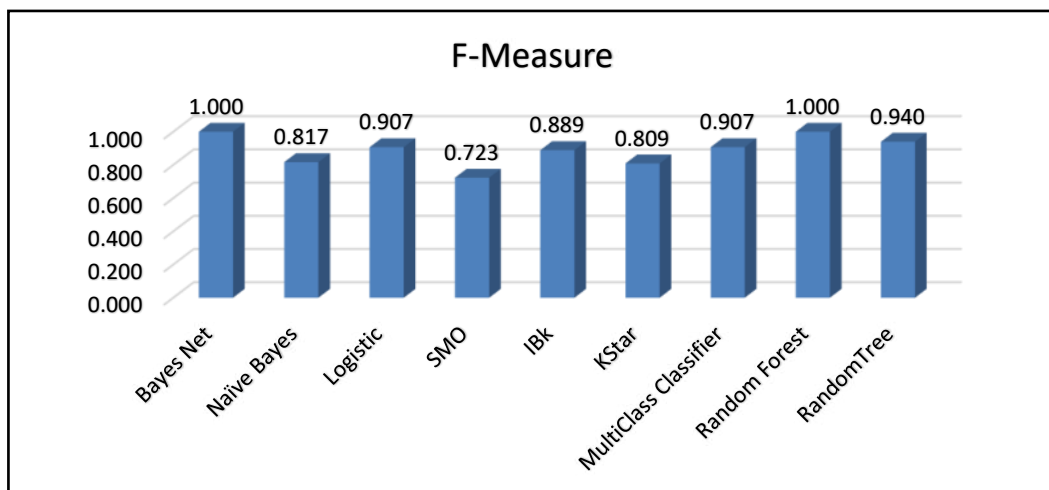
According to the performance measure precision class label: Heavy Traffic it was found that the highest precision value was of the classifiers Bayes Net, Logistic, SMO, KStar, MultiClass Classifier, Random Forest, and Random Tree with value 1.0, whereas the lowest precision value was found to be of the classifiers Naïve Bayes with value 0.950 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure precision is found to be Seven Algorithms.

Figure 5.60: Recall (Cross-Validation: 30% Split)



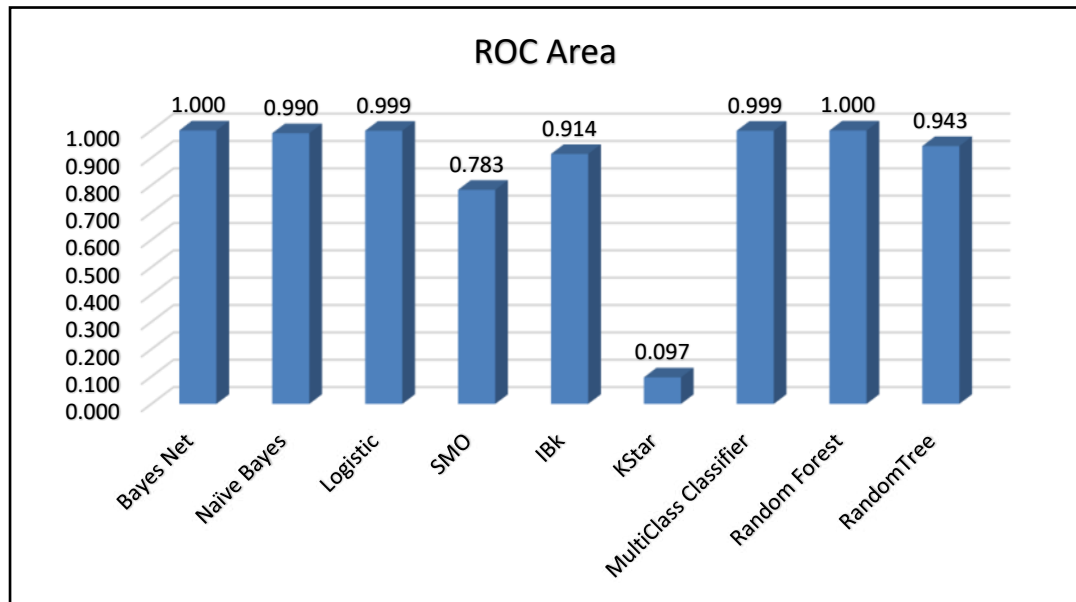
Based on the performance measure recall class label: Heavy Traffic it was found that the highest recall value was of the classifiers Bayes Net and Random Forest with value 1.0, followed by 0.887 of Random Tree respectively whereas the lowest recall value was found to be of the classifiers SMO with value 0.566 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure recall is found to be Bayes Net and Random Forest.

Figure 5.61: F-Measure (Cross-Validation: 30% Split)



According to the performance measure F-Measure class label: Heavy Traffic it was found that the highest F-Measure value was of the classifiers Bayes Net and Random Forest with value 1.0, followed by 0.940 of Random Tree respectively whereas the lowest F-Measure value was found to be of the classifier SMO classifier with value 0.723 respectively. Overall, it can be interpreted that the most appropriate classifier based on the performance measure F-Measure is found to be Bayes Net and Random Forest.

Figure 5.62: ROC Area (Cross-Validation: 30% Split)



Based on the performance measure ROC class label: Heavy Traffic it was found that the highest ROC Area values were of the classifiers Bayes Net and Random Forest with value 1.00, followed by 0.999 of Logistic and MultiClass Classifier respectively whereas the lowest ROC Area value was found to be of the classifier KStar with value 0.097 respectively. Overall, it can be interpreted the most appropriate classifier based on the performance measure ROC Area are found to be Bayes Net and Random Forest.

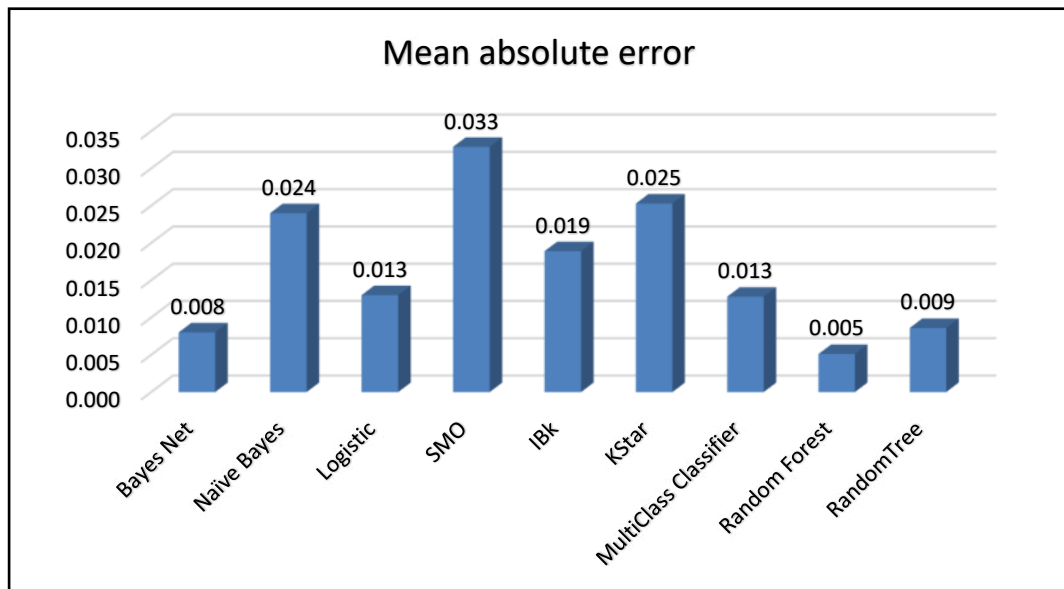
In conclusion the 30% split Cross validation not only increases the reliability of model but also gives insights of model and explains how model behave under different conditions. Sometimes 80:20 model is also used to face real life applications where 80% data is used for Training the model and 20% data is used for testing the model. More percentage data for training also increases model effectiveness and gives superior model performance, paving the way for more trustworthy and impactful model for real life applications.

B. Error Measure Results

Table 5.14: Classifiers and Error Measures (Cross Validation: 30% Split)

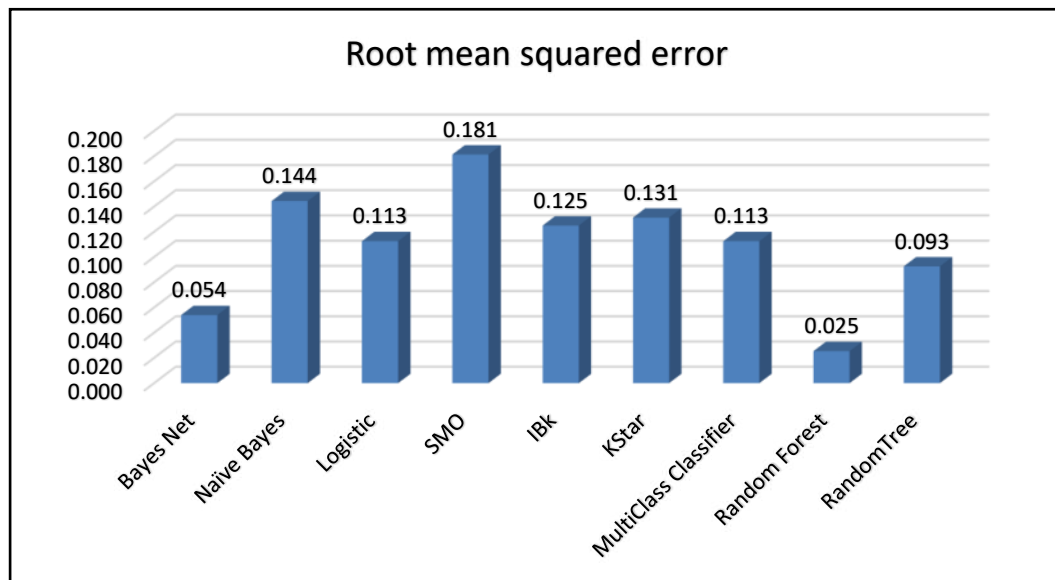
Classifier	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error
Bayes Net	0.008	0.054	7.50%	20.50%
Naïve Bayes	0.024	0.144	20.34%	54.31%
Logistic	0.013	0.113	10.86%	42.35%
SMO	0.033	0.181	27.88%	68.19%
IBk	0.019	0.125	16.05%	47.02%
KStar	0.025	0.131	21.48%	49.39%
MultiClass Classifier	0.013	0.113	10.86%	42.35%
Random Forest	0.005	0.025	4.30%	9.54%
RandomTree	0.009	0.093	7.27%	34.83%

Figure 5.63: Mean Absolute Error (Cross-Validation: 30% Split)



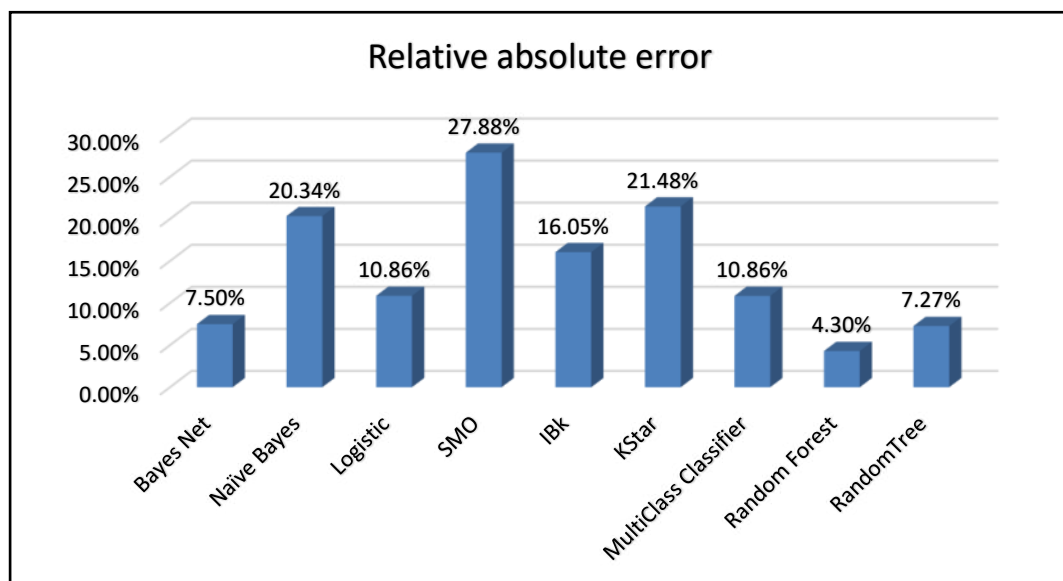
The mean absolute error is found to be lowest in the case of Random Forest with the value 0.005 Whereas the mean absolute error value of SMO is found to be highest with value 0.033.

Figure 5.64: Root Mean Squared Error (Cross-Validation: 30% Split)



The root mean squared error value is found to be highest in case of SMO classifier with the value of 0.181 respectively whereas the lowest value is found to be of Random Forest with value 0.025. So, it can be interpreted that based on the measure RMSE the most appropriate algorithm is found to be Random Forest at configuration setting – 30% Split Method.

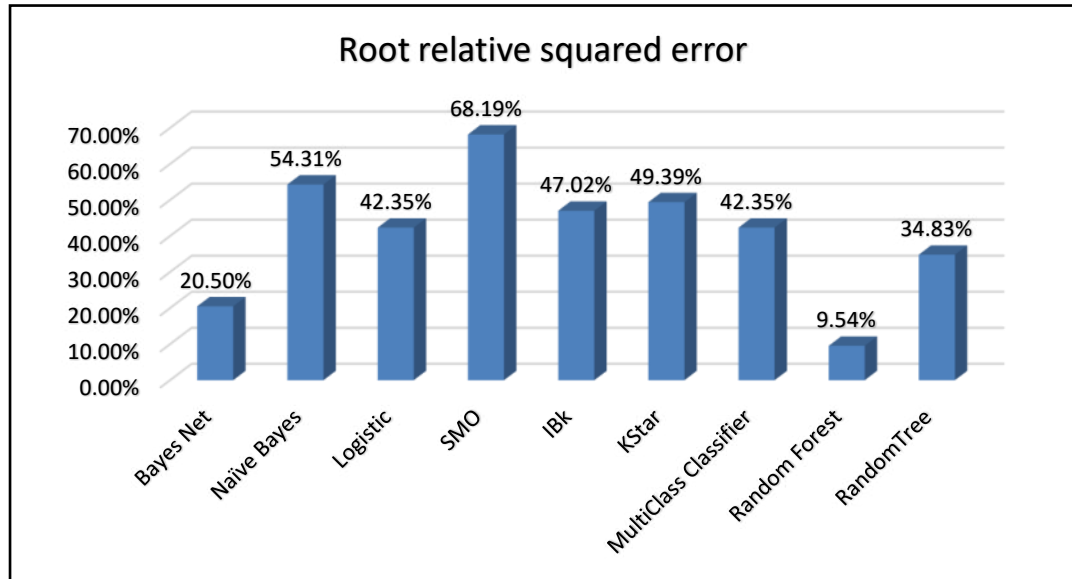
Figure 5.65: Relative Absolute Error (Cross-Validation: 30% Split)



Accordingly, the relative absolute error value is found to be lowest in case of Random Forest classifier with 4.30% whereas the highest relative absolute error percentage

value is found to be in case of SMO with percentage value 27.88% respectively. Based on the measure RAE the most appropriate algorithm is found to be Random Forest.

Figure 5.66: Root Relative Square Error (Cross-Validation: 30% Split)



The root relative squared error value is found to be lowest in case of Random Forest classifier with 9.54% whereas the root relative squared error percentage value is found to be highest in case of SMO with percentage value of 68.19%. So, it can be interpreted that based on the measure RRSE the most appropriate algorithm is found to be Random Forest with lowest percentage value when evaluated at configuration setting – 30% Split cross validation.

In summary, it is important to understand error measures in Machine learning for assessing the performance of model properly and making informed decisions. The choice of specific metrics depends on the nature of the problem, characteristics of the dataset, and goals of the analysis.

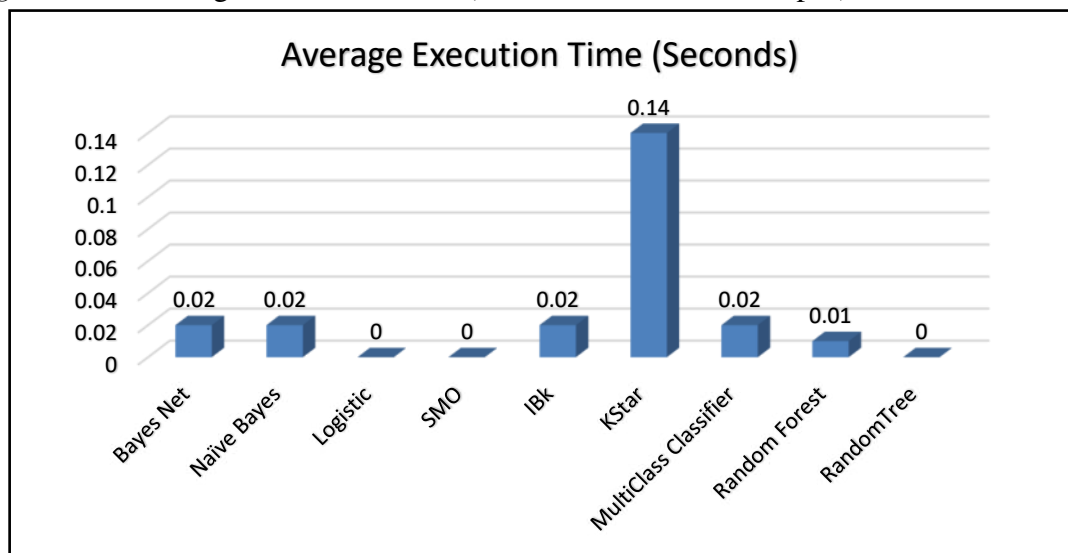
C. Execution Time Results

The execution time of a machine learning algorithm refers to the time it takes for the algorithm to process and analyse input data, train the model (if applicable), and produce predictions or results. Execution time is an important factor when evaluating the efficiency and scalability of machine learning algorithms, especially when dealing with large data sets and real-time applications. The Average Execution Time of Nine Classifier Algorithm for Cross Validation 30% split is given below.

Table 5.15: Classifiers and Average Execution Time (Cross Validation: 30% Split)

Classifier	Average Execution Time (Seconds)
Bayes Net	0.02
Naïve Bayes	0.02
Logistic	0
SMO	0
IBk	0.02
KStar	0.14
MultiClass Classifier	0.02
Random Forest	0.01
Random Tree	0

Figure 5.67: Average Execution Time (Cross-Validation: 30% Split)



According to the performance measure average execution time it was found that the lowest average execution time were of the classifiers Logistic, SMO and Random Tree with values 0.00 each whereas the highest average execution time was found to be of the classifier KStar classifier with value 0.14 respectively. Overall, it can be interpreted that the most appropriate classifiers based on the performance measure average execution time are found to be Logistic, SMO and Random Tree.

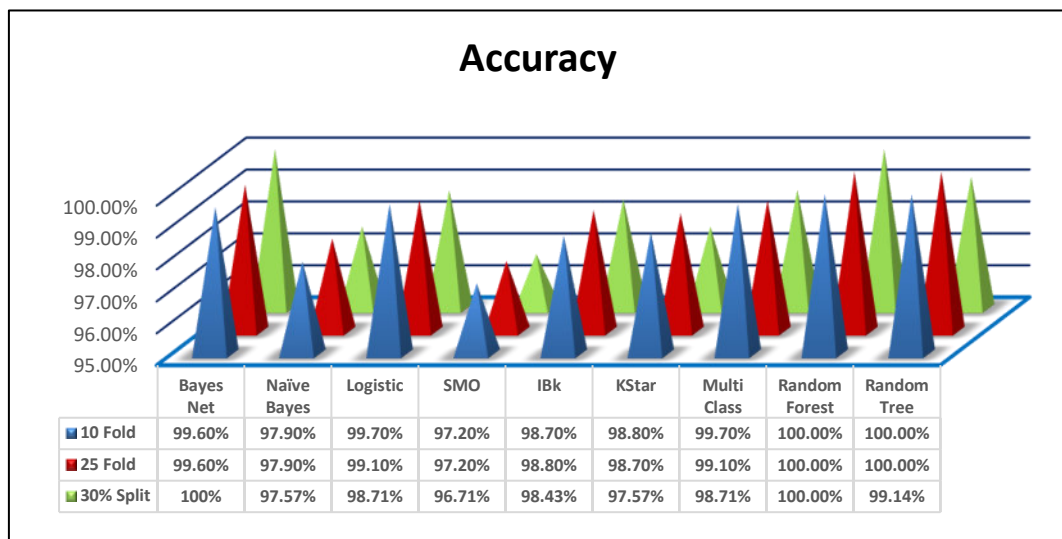
5.3.6 Consolidated Result

Nine Machine Learning Algorithms are Analyzed using Cross – Validation of 10-Fold, 25-Fold and 30% split on various Performance Accuracy Measures, Confusion Matrix Parameters and Error Measures in two different conditions “Low Traffic” and “Heavy Traffic”. Summary of 10 – Fold, 25 – Fold and 30% split is shown below in 3D plots.

A. Performance Measures

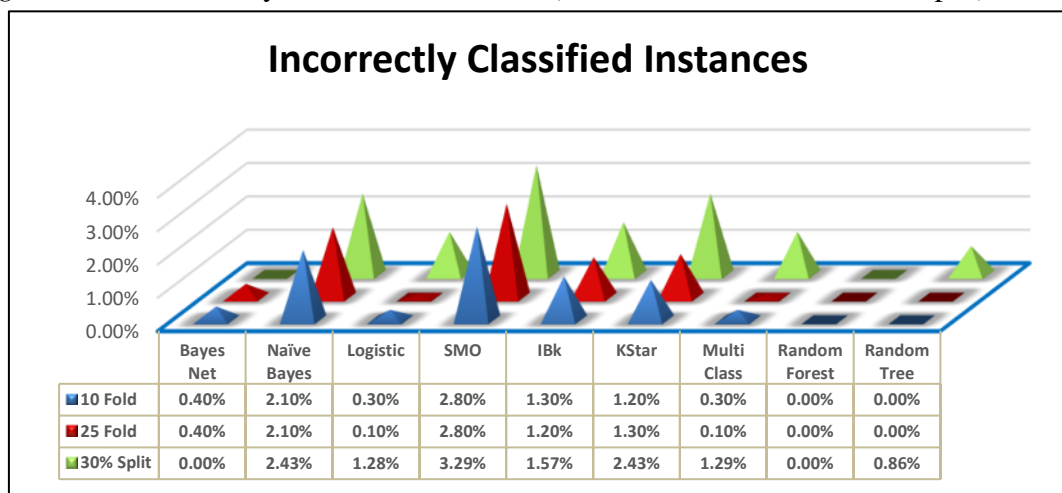
i. Accuracy Measures

Figure 5.68: Accuracy (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



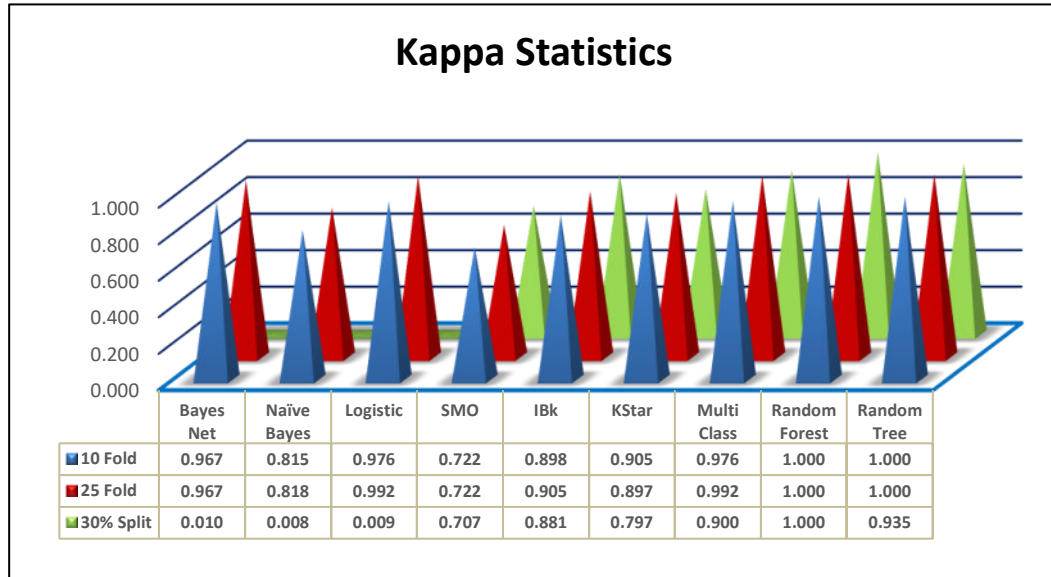
From above three-Dimensional plot it is clear that the maximum average Accuracy score for Random Forest is 100% therefore it is concluded that Random Forest is best algorithms for getting best Accuracy using Cross – Validation of 10-Fold, 25-Fold and 30% Split.

Figure 5.69: Incorrectly Classified Instances (10-Fold, 25-Fold and 30% Split)



From above three-Dimensional plot it is clear that the minimum average incorrectly Classified Instance score for Random Forest is 0% therefore it is concluded that Random Forest is best algorithms for getting best Incorrectly Classified Instances using Cross – Validation of 10-Fold, 25-Fold and 30% Split.

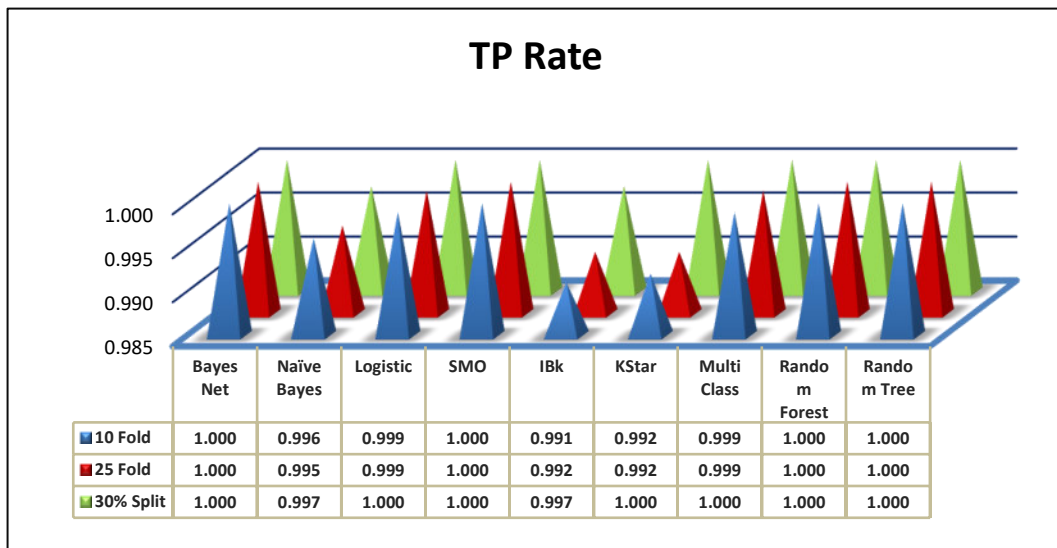
Figure 5.70: Kappa Statistics (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



From above three-Dimensional plot it is clear that the maximum average Kappa Statistics score for Random Forest is 1.0% therefore it is concluded that Random Forest is best algorithms for getting best Kappa Statistics using Cross – Validation of 10-Fold, 25-Fold and 30% Split.

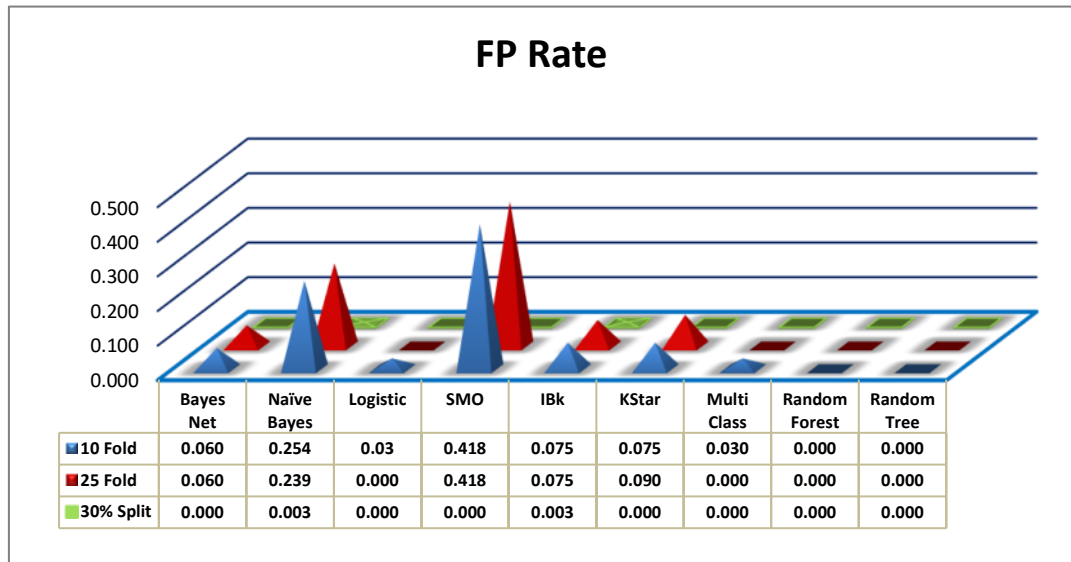
ii. Confusion Matrix Parameters – Low Traffic

Figure 5.71: TP Rate (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



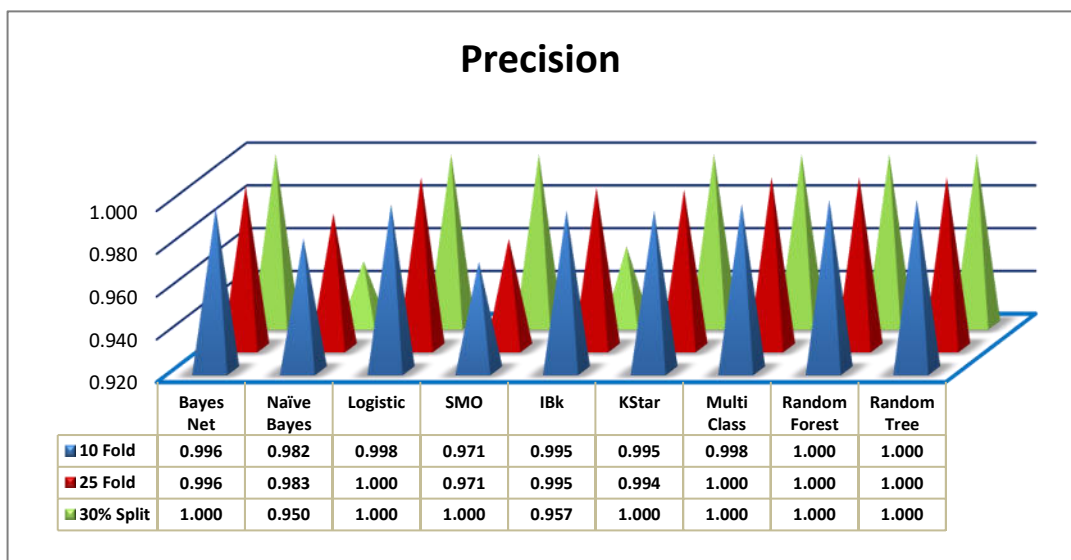
From above three-Dimensional plot it is clear that the maximum average score for Bayes Net, SMO, Random Forest and Random Tree is 1.0 therefore it is concluded that Bayes Net, SMO, Random Forest and Random Tree are best algorithms for getting best True Positive Rate using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Low Traffic Conditions.

Figure 5.72: FP Rate (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



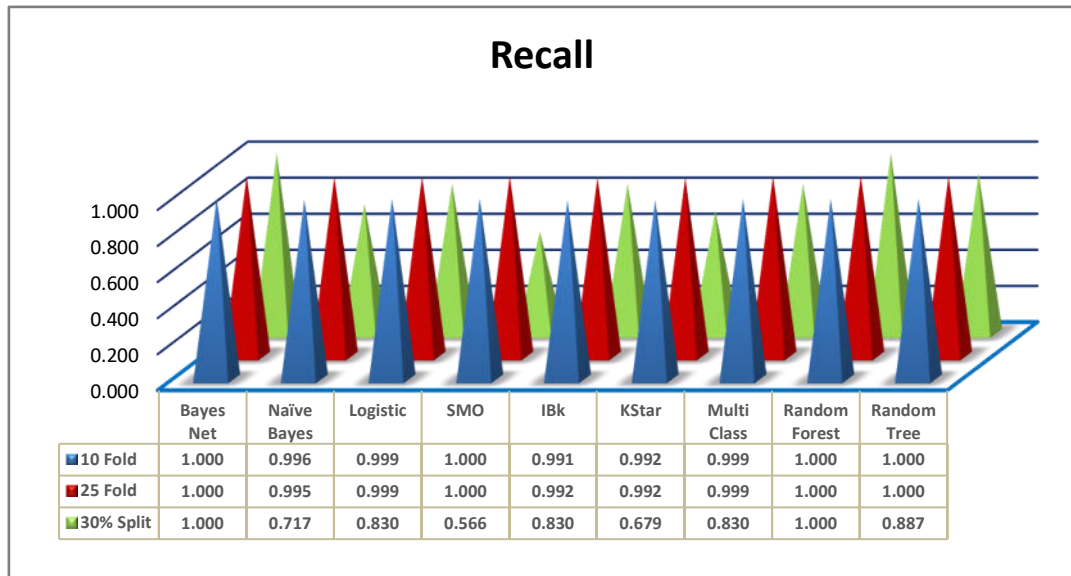
From above three-Dimensional plot it is clear that the minimum average score for Random Forest and Random Tree is 0.0, therefore it is concluded that Random Forest and Random Tree are the best algorithms for getting best False Positive Rate using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Low Traffic Conditions.

Figure 5.73: Precision (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



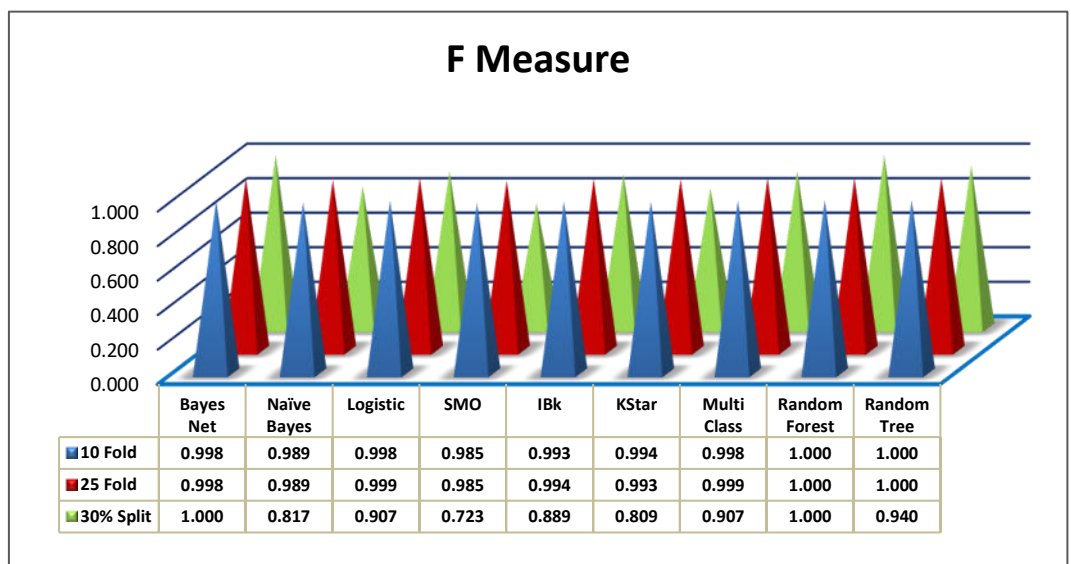
From above three Dimensional plot it is clear that the maximum average score for Random Forest and Random Tree is 1.0, therefore it is concluded that Random Forest and Random Tree are the best algorithms for getting best Precision using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Low Traffic Conditions.

Figure 5.74: Recall (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



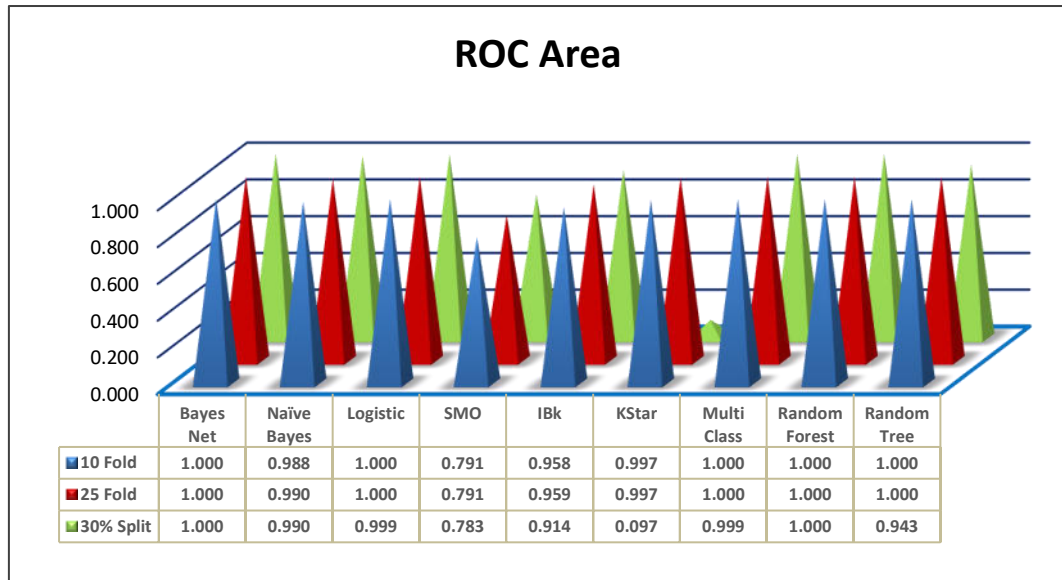
From above three-Dimensional plot it is clear that the maximum average score for Bayes Net and Random Forest is 1.0 therefore it is concluded that Bayes Net and Random Forest are the best algorithms for getting best Recall using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Low Traffic Conditions.

Figure 5.75: F Measure (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



From above three Dimensional plot it is clear that the maximum average score for Random Forest is 1.0, therefore it is concluded that Random Forest is the best algorithms for getting best F Measure using Cross – Validation of 10-Fold, 25-Fold and 30% Split

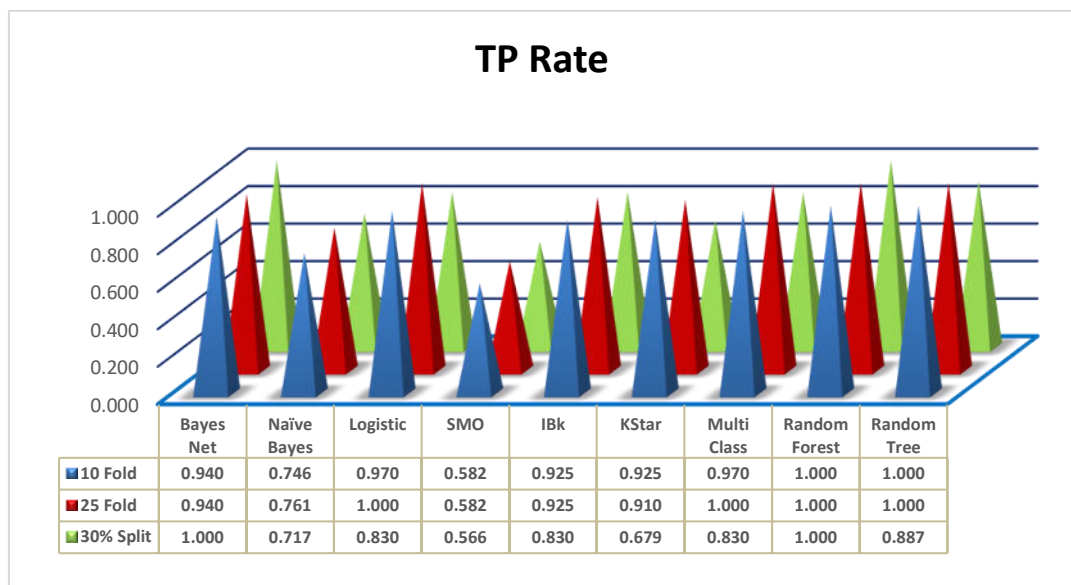
Figure 5.76: ROC Area (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



From above three-Dimensional plot it is clear that the maximum average score for Bayes Net and Random Forest is 1.0 therefore it is concluded that Bayes Net and Random Forest are the best algorithms for getting best ROC Area using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Low Traffic Conditions.

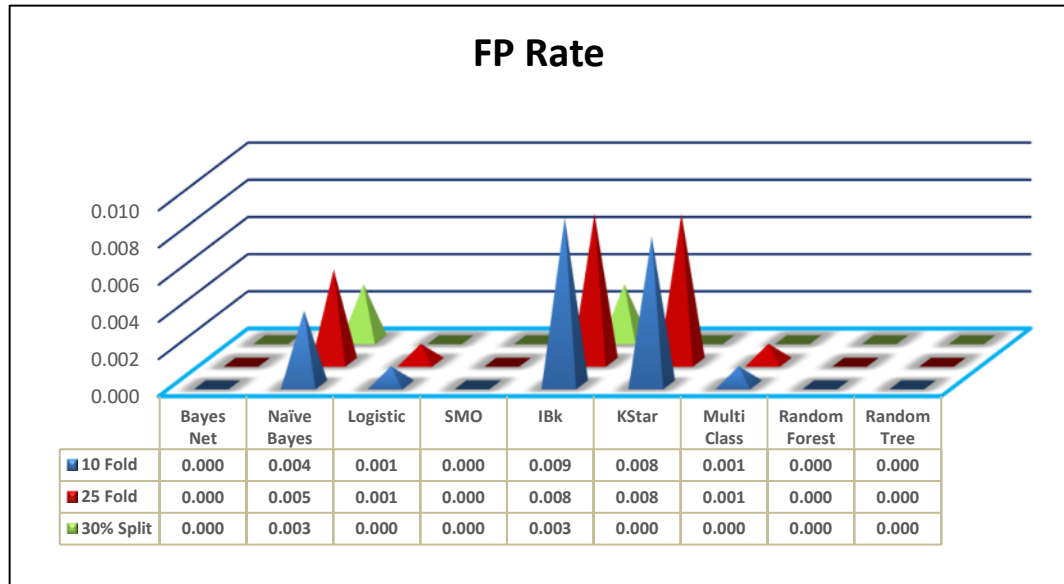
iii. Confusion Matrix Parameters – Heavy Traffic

Figure 5.77: TP Rate (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



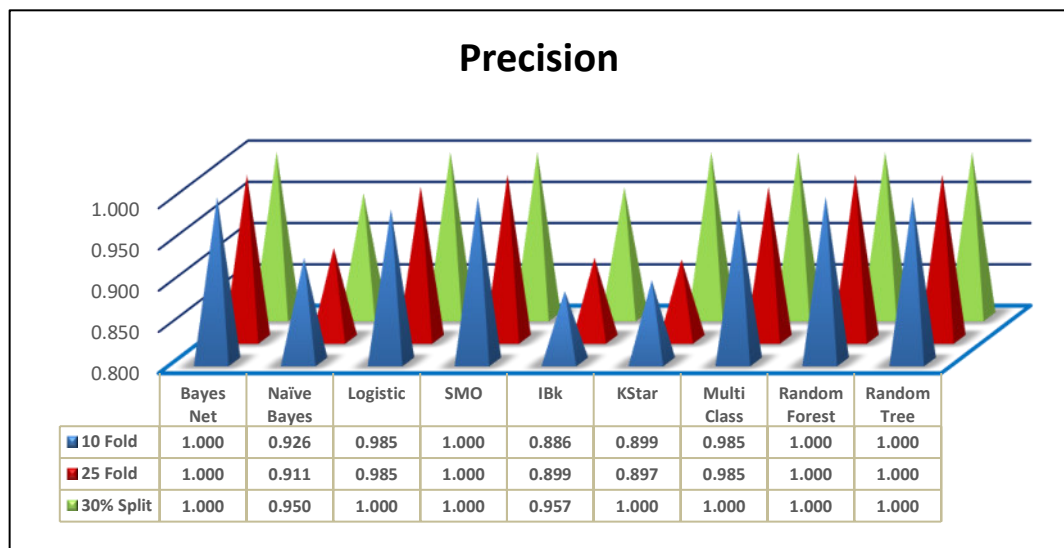
From above three Dimensional plot it is clear that the maximum average TP Rate score for Random Forest is 1.0, therefore it is concluded that Random Forest is the best algorithms for getting best TP Rate using Cross – Validation of 10-Fold, 25-Fold and 30% Split in heavy traffic conditions.

Figure 5.78: FP Rate (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



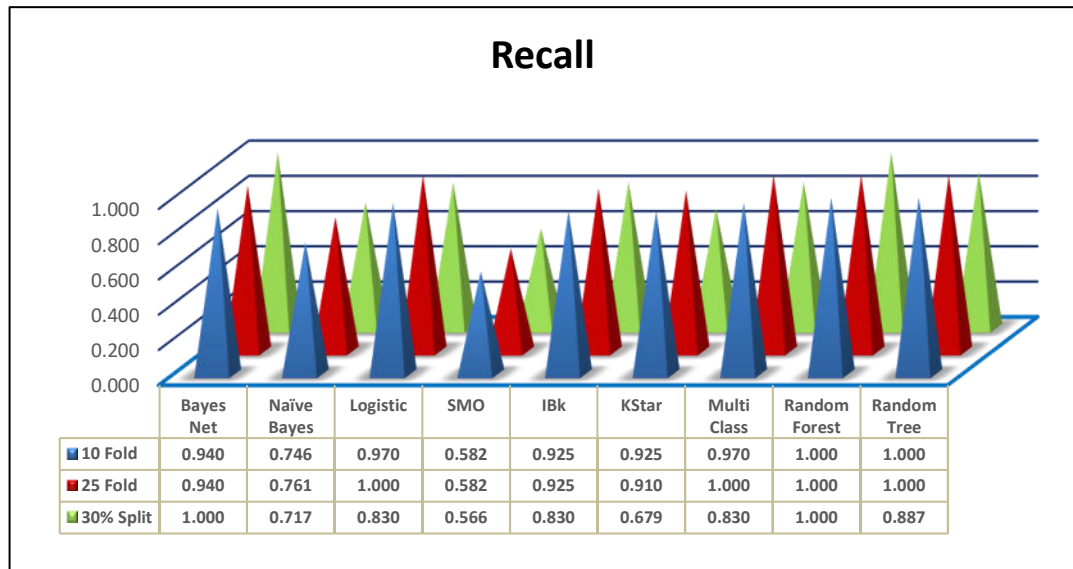
From above three-Dimensional plot it is clear that the minimum average FP Rate score for Bayes Net, SMO, Random Forest and Random Tree is 0.0, therefore it is concluded that Bayes Net, SMO, Random Forest and Random Tree are the best algorithms for getting best False Positive Rate using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Heavy Traffic Conditions.

Figure 5.79: Precision (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



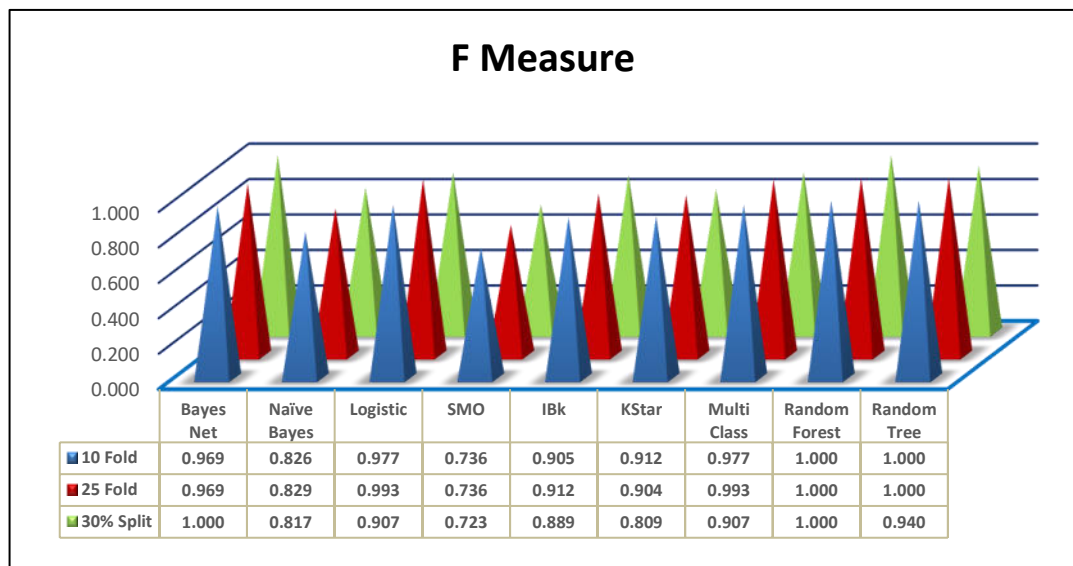
From above three-Dimensional plot it is clear that the maximum average Precision score for Bayes Net, SMO, Random Forest and Random Tree is 1.0, therefore it is concluded that Bayes Net, SMO, Random Forest and Random Tree are the best algorithms for getting best Precision using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Heavy Traffic Conditions.

Figure 5.80: Recall (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



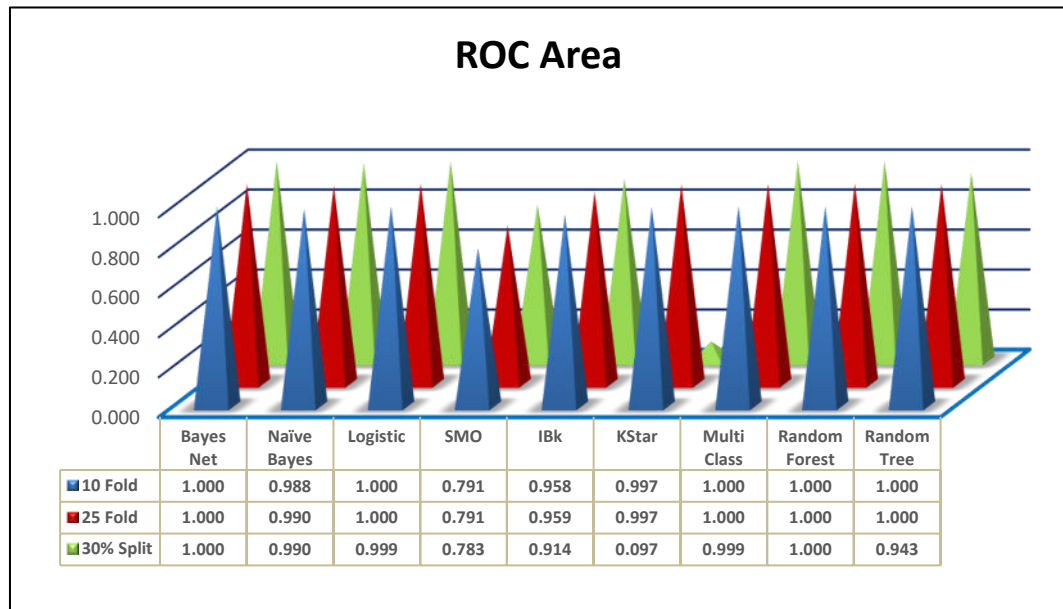
From above three-Dimensional plot the maximum average Recall score for Random Forest is 1.0, therefore it is concluded that Random Forest is the best algorithm for getting best Recall using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Heavy Traffic Conditions.

Figure 5.81: F Measure (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



From above three Dimensional plot it is clear that the maximum average F Measure score for Random Forest is 1.0, therefore it is concluded that Random Forest is the best algorithms for getting best F Measure using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Heavy Traffic Conditions.

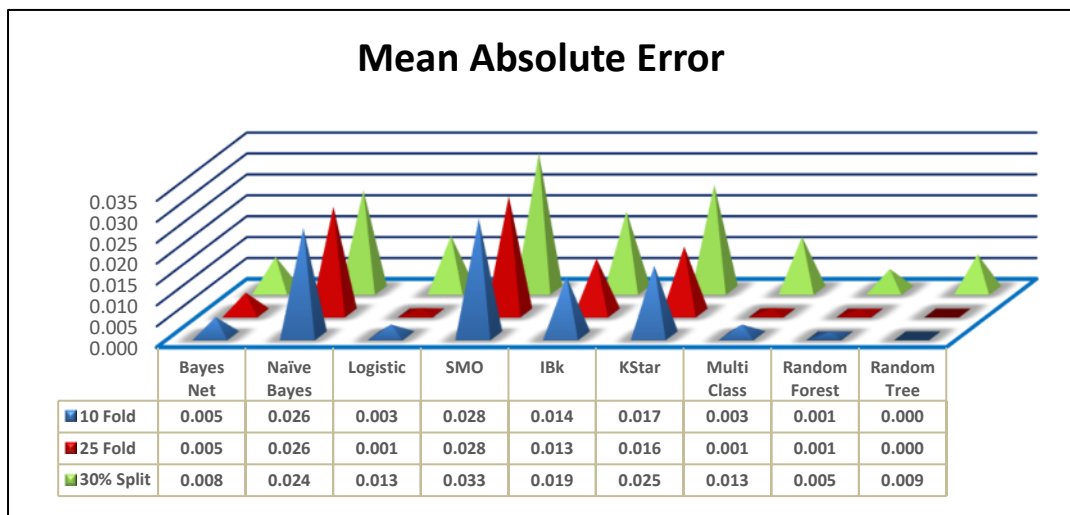
Figure 5.82: ROC Area (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



From above three Dimensional plot it is clear that the maximum average ROC Area score for Bayes Net and Random Forest is 1.0 therefore it is concluded that Bayes Net and Random Forest are the best algorithms for getting best ROC Area using Cross – Validation of 10-Fold, 25-Fold and 30% Split in Heavy Traffic Conditions.

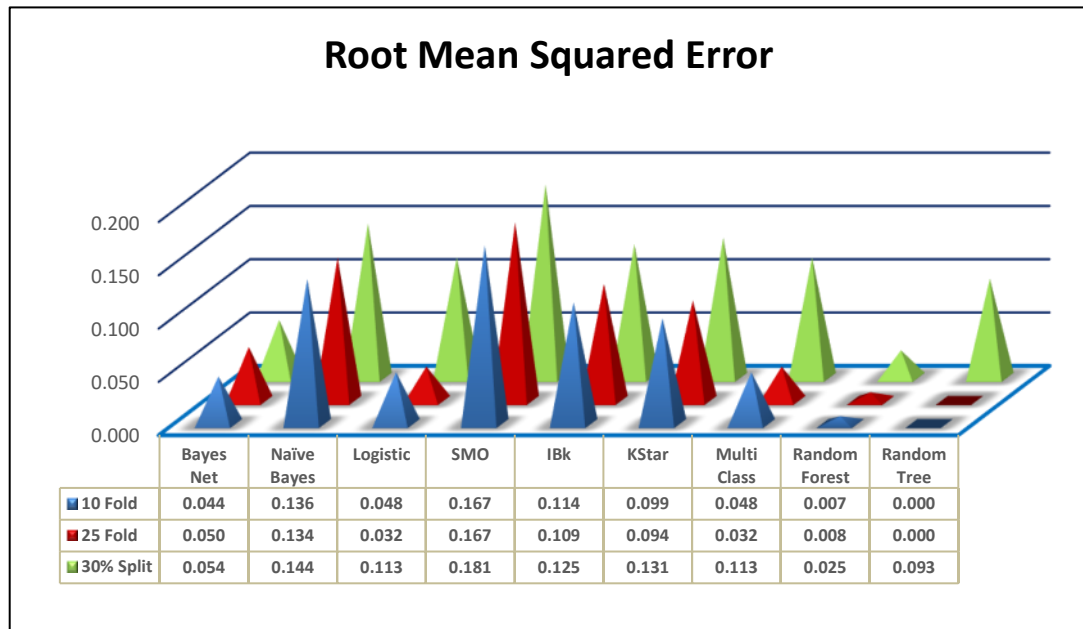
B. Error Measure

Figure 5.83: Mean Absolute Error (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



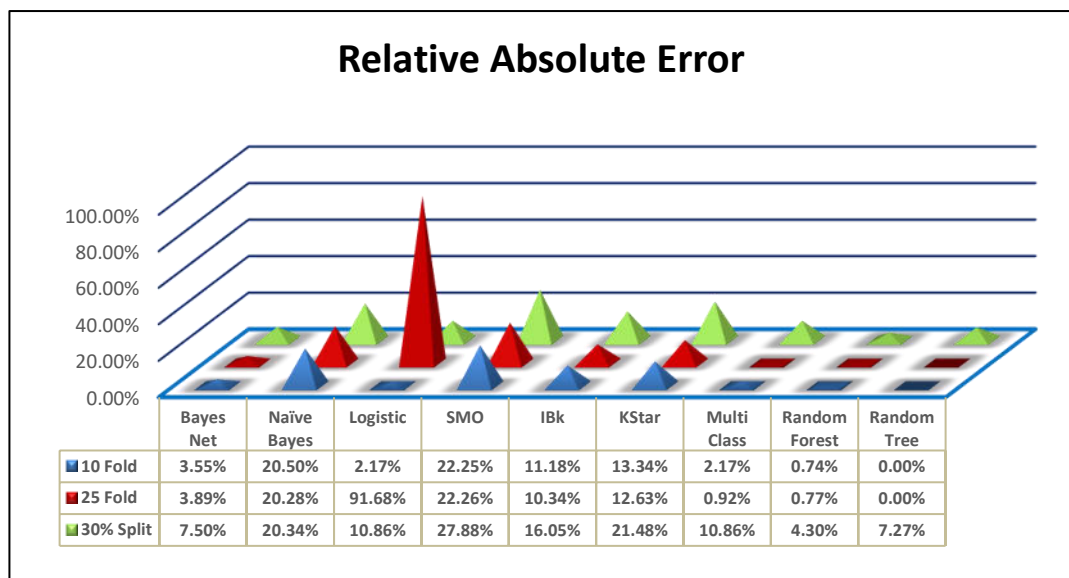
From above three Dimensional plot it is clear that the average Mean Absolute Error score for Random Forest is minimum and it's value is 0.002 therefore it is concluded that Random Forest is the best algorithms for getting Minimum Mean Absolute error using Cross – Validation of 10-Fold, 25-Fold and 30% Split.

Figure 5.84: Root Mean Squared Error (10-Fold, 25-Fold and 30% Split)



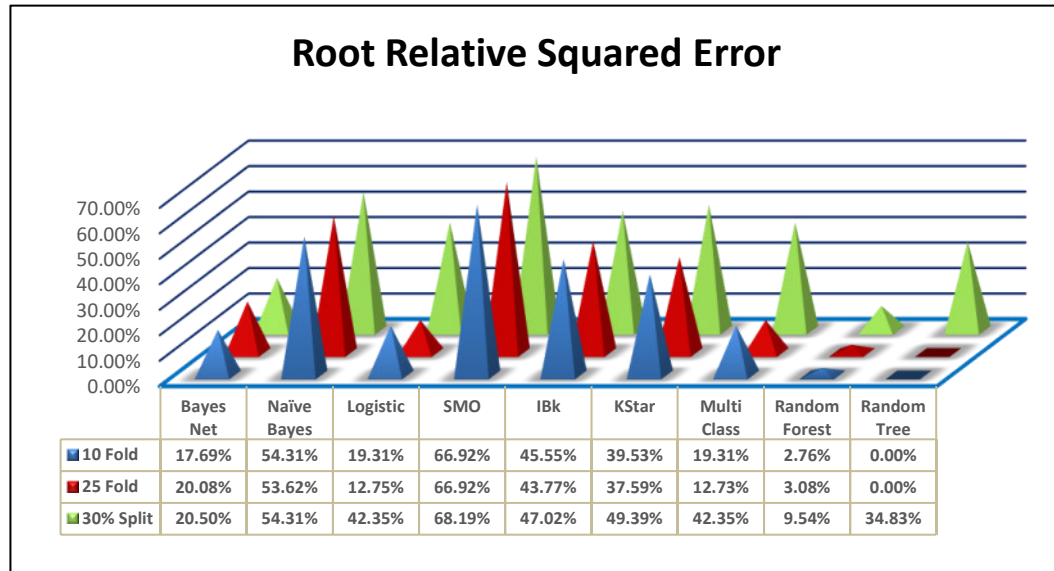
From above three Dimensional plot it is clear that the average Root Mean Squared Error score for Random Forest is minimum and it's value is 0.013, therefore it is concluded that Random Forest is the best algorithms for getting Minimum Root Mean Squared Error using Cross – Validation of 10-Fold, 25-Fold and 30% Split.

Figure 5.85: Relative Absolute Error(Cross-Validation:10-Fold,25-Fold & 30% Split)



From above three Dimensional plot it is clear that the average Relative Absolute Error score for Random Forest is minimum and it's value is 1.94%, therefore it is concluded that Random Forest is the best algorithms for getting Minimum Relative Absolute Error using Cross – Validation of 10-Fold, 25-Fold and 30% Split.

Figure 5.86: Root Relative Squared Error (10-Fold, 25-Fold and 30% Split)

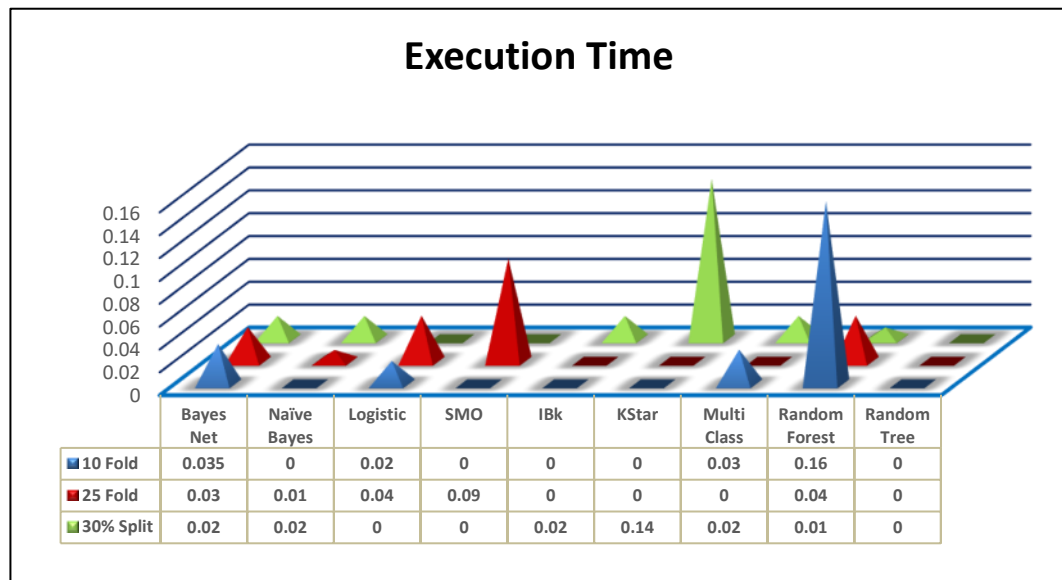


From above three Dimensional plot it is clear that the average Root Relative Squared Error score for Random Forest is minimum and it's value is 5.13%, therefore it is concluded that Random Forest is the best algorithms for getting Minimum Root Relative Squared Error using Cross – Validation of 10-Fold, 25-Fold and 30% Split.

C. Execution Time

Execution Time is one of the important parameter which decides the speed at which algorithm can evaluate and test data sets. It is very important parameter specially when data sets are too large, which is the prime requirement of any machine learning algorithm. If the machine learning algorithm is slow and efficient in all other aspects then cloud computing is the remedy for that. Multitasking in Cloud computing is the solution for slow Machine learning algorithms. The figure 5.87 shown below compares the execution time in seconds of Nine machine learning algorithms.

Figure 5.87: Execution Time (Cross-Validation: 10-Fold, 25-Fold and 30% Split)



The above figure indicates that Random forest Algorithm Average execution time is maximum with average value of 0.07seconds. Random forest algorithm excels in all other parameters except execution time. This drawbacks can be nullified using cloud computing and parallel computing. The Random Tree is the most efficient and fast algorithm with average execution time in nano seconds or nearly zero.

5.3.7 Dominance Chart

Table 5.16 shows the Machine learning Algorithm Dominance chart For 10-fold, 25-fold and 30% split Performance measures. This chart indicates the highest average marks obtained by the machine learning algorithms. One mark is allotted to the machine learning algorithm which got highest score in three categories 10-fold, 25-fold and 30% split. Dominance chart includes twenty parameters from Performance measure, Confusion matrix (Low Traffic and Heavy Traffic), Error measures and Execution time in seconds for finding out Total score. The selected algorithm should have highest marks out of twenty to be the best Machine Learning Algorithm.

Table 5.16 : Machine Learning Algorithms Dominance Chart

Dominance Chart (Cross -Validation 10 Fold, 25 Fold and 30% Split)																					
Classifiers	Performance Measure			Confusion Matrix Parameters Low Traffic						Confusion Matrix Parameters Heavy Traffic						Error Measures				Execution Time	Total Score
	Accuracy	Incorrectly Classified Instances	Kappa statistic	TP Rate	FP Rate	Precision	Recall	F- Measure	ROC Area	TP Rate	FP Rate	Precision	Recall	F- Measure	ROC Area	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error		
Bayes Net				1			1		1		1	1			1						6
Naïve Bayes																					
Logistic																					
SMO				1							1	1									3
IBK																					
K Star																					
Multi Class																					
Random Forest	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		19
Random Tree				1	1	1					1	1								1	6

It is clear from the Table 5.16 that the highest score is obtained by Random Forest algorithm with total score of 19, followed by Random Tree and Bayes Net with six marks each. Thus considering Performance measures, Confusion matrix parameters, Error Measure and Execution Time, it is concluded that the Random Forest is the best Algorithm for forecasting the traffic flow conditions in smart city.

5.3.8 Weighted Sum Model Analysis Using Python

The Weighted Sum Model is a decision-making approach used to evaluate and rank a set of alternatives based on multiple criteria. It involves assigning weights to each criterion and then calculating a weighted sum of the normalized values of each criterion for each alternative. The alternative with the highest weighted sum is considered the best choice. The procedure for Weighted Sum Model is listed below.

1. **Classifier and Metric Definitions:** Begin by defining the list of classifiers or alternatives you want to evaluate. Each classifier is associated with a set of metrics (criteria) that measure its performance.
2. **Metrics and Weights:** Assign weights to each metric (criterion) based on its importance. These weights reflect the relative significance of each criterion in the decision-making process. The sum of weights should add up to 1 or 100% to ensure a meaningful comparison.
3. **Normalization:** Normalize the metric values for each classifier to a common scale, often within the range of 0 to 1. This step ensures that metrics with different units and scales can be compared effectively.
4. **Weighted Sum Calculation:** For each classifier, calculate the weighted sum of its normalized metric values. This is done by multiplying each normalized metric value by its corresponding weight, and then summing up these weighted values.
5. **Best Alternative:** Identify the alternative with the highest calculated weighted sum. This alternative is considered the best choice according to the chosen criteria and their assigned weights.
6. **Decision:** The alternative with the highest weighted sum is selected as the best choice based on the defined criteria and their weights.

The Weighted Sum Model is widely used in decision analysis when there are multiple factors to consider, and each factor carries a different level of importance. It's important to note that the success of the model heavily depends on the accuracy of the assigned weights and the relevance of the chosen metrics.

Python Pseudocode:

```
// Define classifier names
classifiers = ["Classifier 1", "Classifier 2", ...]

// Provided metrics (Replace with your actual metrics)
metrics = [
    [metric_value_1_classifier_1, metric_value_2_classifier_1, ...],
    [metric_value_1_classifier_2, metric_value_2_classifier_2, ...],
    ...
]

// Define weights for each metric (customize these weights)
metric_weights = [weight_metric_1, weight_metric_2, ...]

// Function to normalize metrics to [0, 1] range
function NormalizeMetrics(metrics):
    normalized_metrics = EmptyMatrix()
    for each classifier_metrics in metrics:
        normalized_classifier_metrics = Normalize(classifier_metrics)
        AddToMatrix(normalized_metrics, normalized_classifier_metrics)
    return normalized_metrics

// Function to calculate the weighted sum for each classifier
function CalculateWeightedSums(normalized_metrics, metric_weights):
    weighted_sums = EmptyList()
    for each classifier_metrics in normalized_metrics:
        weighted_sum = CalculateDotProduct(classifier_metrics, metric_weights)
        AppendToList(weighted_sums, weighted_sum)
    return weighted_sums
```

```

// Function to find the best alternative (index)
function FindBestAlternative(weighted_sums):
    best_alternative_index = IndexOfMax(weighted_sums)
    return best_alternative_index

// Function to print weighted sums
function PrintWeightedSums(classifiers, weighted_sums):
    for i from 0 to length(classifiers) - 1:
        Print(classifiers[i], " - Weighted Sum:", weighted_sums[i])

// Function to print the best alternative
function PrintBestAlternative(best_alternative):
    Print("The best alternative is:", best_alternative)

// Main program
function Main():
    normalized_metrics = NormalizeMetrics(metrics)
    weighted_sums = CalculateWeightedSums(normalized_metrics, metric_weights)
    best_alternative_index = FindBestAlternative(weighted_sums)
    best_alternative = classifiers[best_alternative_index]
    PrintWeightedSums(classifiers, weighted_sums)
    PrintBestAlternative(best_alternative)

// Call the main program
Main()

```

5.3.8.1 Multi-Criteria Decision Making - Weighted Sum Method

The Weighted Sum Method, a fundamental technique in Multi-Criteria Decision Making, facilitates decision-makers in evaluating and ranking alternatives by considering multiple criteria. This approach involves identifying relevant decision criteria, assigning weights to signify their importance, evaluating each alternative's performance on these criteria, normalizing scores to ensure comparability, and calculating a weighted sum for each alternative. The resulting scores enable a systematic ranking of alternatives, aiding decision-makers in selecting the most suitable

option that aligns with their preferences and objectives. This method serves as a valuable decision support tool across various domains.

i. Evaluation with Cross Validation-10 Folds

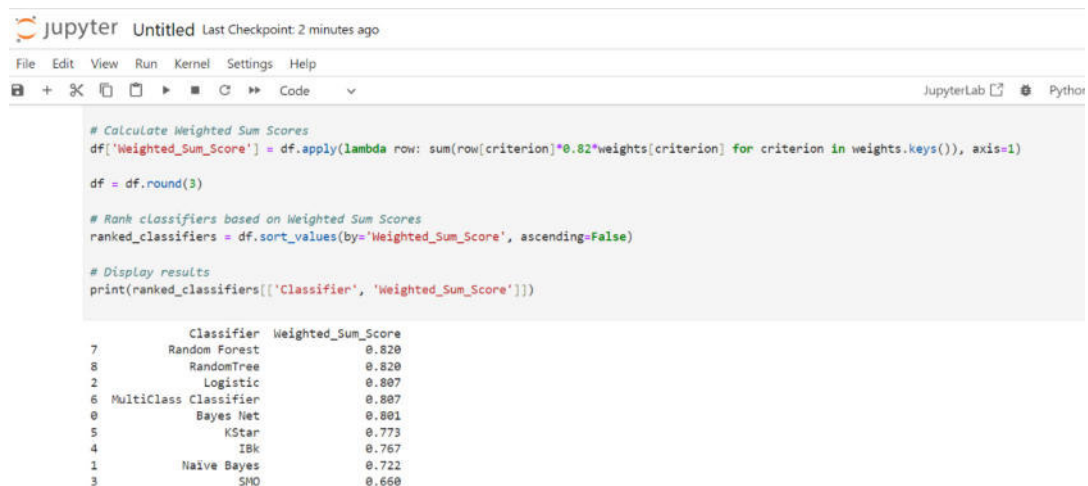
Table 5.17 presents a range of classification models assessed by their "Score (Weighted Sum)," with higher scores indicating superior performance.

Table 5.17: Weighted Sum Score (Cross Validation-10 Folds)

S.No.	Classification Models	Score (Weighted Sum)
1	Bayes Net	0.801
2	Naïve Bayes	0.722
3	Logistic	0.807
4	SMO	0.660
5	IBk	0.767
6	KStar	0.773
7	MultiClass Classifier	0.807
8	Random Forest	0.820
9	Random Tree	0.820

The weighed sum-based score suggests that Random Forest and Random Tree has the highest score of 0.820 followed by Logistic and Multiclass classifier with core 0.807 and Bayes Net with score 0.801. The lowest scores being obtained by the classifiers KStar, IBK, Naïve Bayes and SMO with values 0.773,0.767,0.722 and 0.660 respectively. Further Rank and Percentile analysis would be executed to obtain the final rankings.

Figure 5.88: Running MCDM Method in Python Environment



```

# Calculate Weighted Sum Scores
df['Weighted_Sum_Score'] = df.apply(lambda row: sum(row[criterion]*0.82*weights[criterion] for criterion in weights.keys()), axis=1)

df = df.round(3)

# Rank classifiers based on Weighted Sum Scores
ranked_classifiers = df.sort_values(by='Weighted_Sum_Score', ascending=False)

# Display results
print(ranked_classifiers[['Classifier', 'Weighted_Sum_Score']])

```

Classifier	Weighted_Sum_Score
Random Forest	0.820
RandomTree	0.820
Logistic	0.807
MultiClass Classifier	0.807
Bayes Net	0.801
KStar	0.773
IBk	0.767
Naïve Bayes	0.722
SMO	0.660

5.3.9 Rank and Percentile Method:

Ranking and percentile methods can be applied to arrange classification algorithms in machine learning based on their performance metrics. Here's how you can use these methods:

Ranking Method Algorithm:

1. **Collect Performance Metrics:** Gather performance metrics (e.g., accuracy, precision, recall, F1 score, ROC AUC) for each classification algorithm. These metrics are typically computed using cross-validation or other evaluation techniques.
2. **Calculate Ranks:** Calculate ranks for each algorithm based on each performance metric. Assign a rank of 1 to the algorithm with the highest score for a metric, 2 to the second highest, and so on. In the case of ties, you can use methods like averaging ranks.
3. **Calculate Average Rank:** After ranking algorithms for each metric, calculate the average rank for each algorithm across all the metrics. This average rank represents the overall ranking for each algorithm.
4. **Sort and Present Results:** Sort the algorithms based on their average ranks in ascending order. The algorithm with the lowest average rank is considered the top performer, while the one with the highest average rank is considered the lowest performer. You can present these rankings in a table or report.

Percentile Method Algorithm:

1. **Collect Performance Metrics:** Similar to the ranking method, gather performance metrics for each classification algorithm.
2. **Calculate Percentiles:** For each performance metric, calculate the percentile rank of each algorithm. Percentile rank indicates the percentage of algorithms that performed worse than a particular algorithm for a given metric.
3. **Calculate Average Percentile Rank:** After calculating percentiles for each metric, compute the average percentile rank for each algorithm across all the metrics. This average percentile rank represents the overall ranking for each algorithm.

4. **Sort and Present Results:** Sort the algorithms based on their average percentile ranks in ascending order. Lower average percentile rank indicates better performance across multiple metrics.

Python Pseudocode : Rank_Classification_Algorithm

Input:

- List of classification algorithms (Algorithms)
- List of performance metrics (Metrics)
- Dictionary of algorithm performance data (AlgorithmMetrics)

Output:

- Sorted list of algorithms based on average rank (RankedAlgorithms)

Begin:

```
// Initialize an empty dictionary to store ranks for each algorithm
```

```
Initialize an empty dictionary AlgorithmRanks
```

```
// Step 1: Calculate ranks for each algorithm and metric
```

```
For each Algorithm in Algorithms:
```

```
    // Initialize a list to store ranks for each metric
```

```
    Initialize an empty list MetricRanks
```

```
    For each Metric in Metrics:
```

```
        // Calculate the rank for the Algorithm based on the Metric
```

```
        Rank = CalculateRank(AlgorithmMetrics[Algorithm][Metric])
```

```
        // Append the rank to the MetricRanks list
```

```
        Append Rank to MetricRanks
```

```
    End For
```

```
    // Calculate the average rank for the Algorithm
```

```
    AverageRank = CalculateAverageRank(MetricRanks)
```

```
    // Store the average rank in the AlgorithmRanks dictionary
```

```
    AlgorithmRanks[Algorithm] = AverageRank
```

```
End For
```

```
// Step 2: Sort algorithms based on average rank
```

```
SortedAlgorithms = SortAlgorithmsByRank(AlgorithmRanks)
```

```
// Step 3: Output the sorted list of algorithms
```

```
Return SortedAlgorithms
```

End

Function: CalculateRank

Input:

- List of performance scores (Scores)

Output:

- Rank for the algorithm based on the scores (Rank)

Begin:

// Sort the scores in descending order

SortedScores = SortScoresDescending(Scores)

// Initialize the rank as 1

Rank = 1

For each Score in SortedScores:

// Assign the current rank to the Score

Set Rank for Score = Rank

// Increment the rank for the next Score if it has the same value

If NextScoreExists() AND NextScore() = Score Then

Increment Rank

End If

End For

Return Rank

End

Function: CalculateAverageRank

Input:

- List of ranks (Ranks)

Output:

- Average rank (AverageRank)

Begin:

// Calculate the mean (average) of the ranks

AverageRank = Mean(Ranks)

Return AverageRank

End

Function: SortAlgorithmsByRank

Input:

- Dictionary of algorithm ranks (AlgorithmRanks)

Output:

- Sorted list of algorithms based on rank (SortedAlgorithms)

```

Begin:
    // Sort the algorithms based on their average ranks
    SortedAlgorithms = Sort(Algorithms, AlgorithmRanks[Algorithm])

    Return SortedAlgorithms
End

```

The outcomes shows the ranking given to various classification algorithm (applied at cross validation- 10 folds) using the rank and percentile method. The results in the table below shows the classification model (evaluated at cross validation 10-folds), point (classifier ID), Rank and Percentage.

Table 5.18: Classification Models and Ranks (Cross Validation-10 Folds)

Classification Models	Point (Classifier ID)	Score (Weighted Sum)	Rank	Percentile
Random Forest	8	0.820	1	100.00%
Random Tree	9	0.820	1	100.00%
Logistic	3	0.807	3	62.50%
MultiClass Classifier	7	0.807	3	62.50%
Bayes Net	1	0.801	5	50.00%
KStar	6	0.773	6	37.50%
IBK	5	0.767	7	25.00%
Naïve Bayes	2	0.722	8	12.50%
SMO	4	0.660	9	0%

It was found that based on configuration setting: cross validation – 10 folds Random Forest and Random Tree are the best and most appropriate classifier for traffic congestion control and traffic flow as both are having the highest score of 0.820 with percentile 100%. Logistic and MultiClass Classifier are the second most appropriate algorithms having total score of 0.807, rank 3 and percentile of 62.50%. The third best classifier being identified is Bayes net with a total score of 0.801 and percentile of 50.00%. For predicting the Udaipur traffic flow Random Forest and Random Tree are the most appropriate algorithms.

ii. Evaluation with Cross Validation-25 Folds

Table 5.19 shows the ranking given to various classification algorithms (applied at cross validation- 25 folds) using the rank and percentile method.

Table 5.19: Weighted Sum Score (Cross Validation-25 Folds)

S. No.	Classification Models	Score (Weighted Sum)
1	Bayes Net	0.801
2	Naïve Bayes	0.724
3	Logistic	0.815
4	SMO	0.660
5	IBk	0.770
6	KStar	0.769
7	MultiClass Classifier	0.815
8	Random Forest	0.820
9	RandomTree	0.820

The weighed sum-based score suggests that Random Forest and Random Tree has the highest score of 0.820 followed by Logistic and MultiClass Classifier with score 0.815. The results in Table 5.20 show the classification model (evaluated at cross validation 25-folds), point (classifier ID), Weighted Sum, Rank and Percentage.

Table 5.20: Classification Models and Ranks (Cross Validation-25 Folds)

Classification Models	Point (Classifier ID)	Score (Weighted Sum)	Rank	Percent
Random Forest	8	0.820	1	100.00%
Random Tree	9	0.820	1	100.00%
Logistic	3	0.815	3	62.50%
MultiClass Classifier	7	0.815	3	62.50%
Bayes Net	1	0.801	5	50.00%
IBK	5	0.770	6	37.50%
KStar	6	0.769	7	25.00%
Naïve Bayes	2	0.724	8	12.50%
SMO	4	0.660	9	0.00%

It was found that based on configuration setting: cross validation – 25 folds Random Forest and Random Tree are the best and most appropriate classifier for traffic

congestion control and traffic flow as it has the highest score of 0.820 with percentile 100%. Logistic and MultiClass Classifier are the second most appropriate algorithm having a total score of 0.815, rank 3 and percentile of 62.50%. The third best classifier being identified is Bayes Net with a total score of 0.801 and percentile of 50.00%. For predicting the Udaipur traffic flow Random Forest is the most appropriate algorithm.

iii. Evaluation with Cross Validation-Split: 30%

Table 5.21 shows the ranking given to various classification algorithms (applied at cross validation- 30% Split) using the rank and percentile method. The weighed sum-based score suggests that Random forest has the highest score of 0.820 followed by Random Tree with score 0.779 and MultiClass Classifier with score 0.765.

Table 5.21: Weighted Sum Score (Cross Validation-30% Split)

S. No.	Classification Models	Score (Weighted Sum)
1	Bayes Net	0.658
2	Naïve Bayes	0.585
3	Logistic	0.619
4	SMO	0.652
5	IBk	0.749
6	KStar	0.638
7	MultiClass Classifier	0.765
8	Random Forest	0.820
9	RandomTree	0.779

The result in the table below shows the classification model ,point (classifier ID), Rank and Percentage evaluated at cross validation 30%-Split.

Table 5.22: Classification Models and Ranks (Cross Validation-30% Split)

Classification Models	Point (Classifier ID)	Score (Weighted Sum)	Rank	Percent
Random Forest	8	0.820	1	100.00%
Random Tree	9	0.779	2	87.50%
Multi Class Classifier	7	0.765	3	75.00%
IBK	5	0.749	4	62.50%
Bayes Net	1	0.658	5	50.00%
SMO	4	0.652	6	37.50%
KStar	6	0.638	7	25.00%
Logistic	3	0.619	8	12.50%
Naïve Bayes	2	0.585	9	0.00%

It was found that based on configuration setting: cross validation – 30% Split Random forest is the best and most appropriate classifier for traffic congestion control and traffic flow as it has the highest score of 0.820 with percentile 100%. Random Tree is the second most appropriate algorithm having a total score of 0.779, rank 2 and percentile of 87.50%. The third best classifier being identified is Multi Class Classifier with a total score of 0.765 and percentile of 75%. For predicting the Udaipur traffic flow Random forest is the most appropriate algorithm while considering configuration setting cross validation – 30%.

Finally, a Random Forest-based predictive model's high ranking in a cross-validation setting, whether it's 10-fold, 25-fold or 30% split indicates its robustness and effectiveness in handling the complexities of traffic management. Its ability to capture non-linear patterns, handle real-time data, and provide insights into important features makes it a valuable tool for improving traffic flow, reducing congestion, and enhancing overall transportation efficiency.

5.4 Hypothesis Testing Results

Hypothesis is nothing but a tentative statement or proposed explanation made based on limited evidence as a starting point for further investigation. The following two hypothesis are being tested for proposed research work.

Hypothesis 1:

The main objective of my research is to solve commuting problems in smart cities using Artificial Intelligence, IoT and Machine Learning technologies. To reach meaningful conclusion of my research I am interested in finding whether there is significant difference between the type of smart technologies used in smart cities. To examine the difference between different categorical variables Chi-Square test is applied after doing survey from different age group participants from different cities. The Hypothesis statements are:

H₀1: There is no significant difference between technologies used for enhancing the transportation system in smart cities.

The related alternative hypothesis is as follows.

H_a1: There is a significant difference between technologies used for enhancing the transportation system in smart cities.

Test Applied: To test the hypothesis H₀1 the Chi-Square Test was being used. The outcomes of the Chi-Square test are shown below in the table.

Table 5.23: Type of Technology and Level of Enhancement

Type of Technology and Level of Enhancement in Smart Transportation System: Crosstabulation				
Count				
		Enhancement in Smart Transportation System		Total
		High	Low	
Type of Technology	AI Based	12	0	12
	Fog Computing	7	0	7
	IoT-Based Traffic Prediction Models	7	6	13
	Machine Learning-Based Traffic Prediction Models	14	4	18
Total		40	10	50

Table 5.24: Chi-Square Test Results

Chi-Square Test			
	Value	df	Asymptotic Significance (2-sided)
Pearson Chi-Square	10.363 ^a	3	.016
Likelihood Ratio	13.026	3	.005
N of Valid Cases	50		
a. 4 cells (50.0%) have expected count less than 5. The minimum expected count is 1.40.			

Table 5.25: Calculation of Expected Frequency

Calculation of Expected Frequency			
Total of Technology	Total of Enhancement in Smart Transportation System	Expected Frequency	EF
12	40	12*40/50	9.6
7	40	7*40 /50	5.6
13	40	13*40 /50	10.4
18	40	18*40 /50	14.4
12	10	12*10 /50	2.4
7	10	7*10 /50	1.4
13	10	13*10 /50	2.6
18	10	18*10 /50	3.6

Table 5.26: Observed and Expected Frequency calculations.

Observed and Expected Frequency for the calculation of X²			
Observed Frequency (OF)	Expected Frequency (EF)	(OF - EF)²	(OF - EF)² / EF
12	9.6	5.76	0.6
7	5.6	1.96	0.35
7	10.4	11.56	1.11
14	14.4	0.16	0.01
0	2.4	5.76	2.4
0	1.4	1.96	1.4
6	2.6	11.56	4.45
4	3.6	0.16	0.04
		Total (Σ)	10.36

Degree of Freedom =(r-1) (c-1)

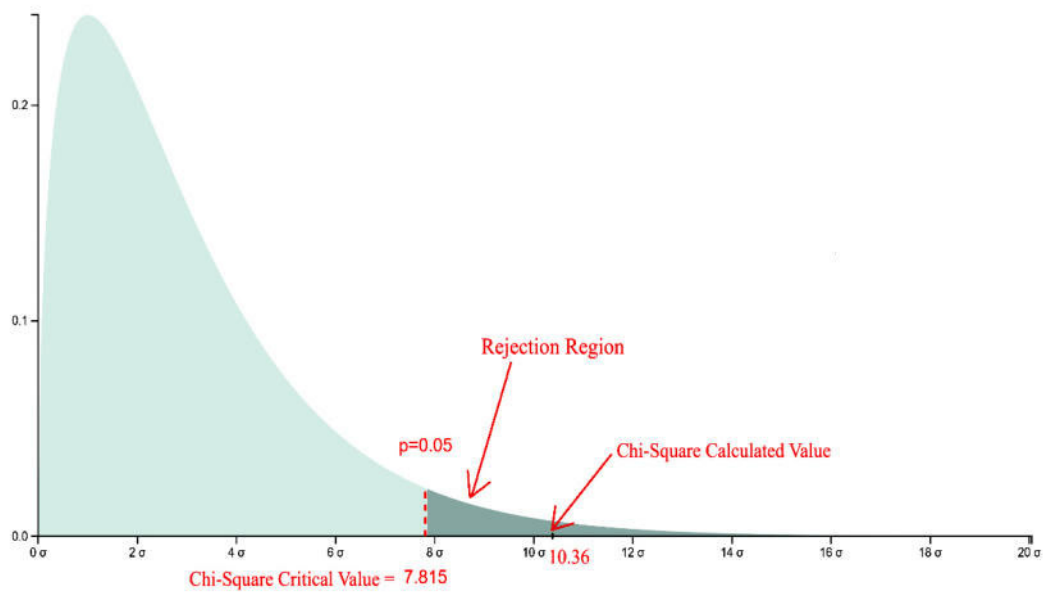
= (4-1) *(2-1) =3

Table value @5% level of significance = 7.815

Calculated Value of Chi-Square = 10.36

Result: The Chi-Square test results confirm that as the Pearson Chi-Square value was found to be 10.36 at degree of freedom 3 and the corresponding p-value is found to be 0.016 which is lesser than the standard alpha value of 0.05 this interpret that the null hypothesis H_0 is rejected and alternate hypothesis H_a is being accepted and it can be concluded that there is significant difference between technologies used for enhancing the transportation system for smart cities.

Figure 5.89: Right Tailed Chi-Square curve



Source: Chi-Square Distribution Calculator With Graph Generator [28]

From above figure Right Tailed Chi-Square curve it is clear that Calculated Chi-Square value lies in the rejection region therefore H_0 is rejected and H_a is accepted. This indicates that there is a significant difference between the listed technologies like AI based, Fog Computing based, IoT based and Machine learning based traffic prediction models in solving traffic congestion problems in smart cities.

Hypothesis 2:

The weighted sum method algorithm for traffic prediction model generates different performance scores for 10-fold, 25-fold and 30% split case with different weights to each criterion. We are interested to know whether all machine learning algorithms average performance score is more than 75%. Following are the Hypothesis statements.

H₀2: The Machine learning-based traffic prediction models have average performance scores of greater than or equal to 75%.

The related alternative hypothesis is as follows.

H_a2: The Machine learning-based traffic prediction models have average performance scores of less than 75%.

$$\mathbf{H_0: \mu \geq 0.75}$$

$$\mathbf{H_a: \mu < 0.75}$$

Test Applied: As Sample mean is known and number of samples are less than 30, therefore One-Sample lower tail t-test is applied to test the hypothesis H₀2. Calculations of t-test are shown in the table shown below.

Table 5.27: T-test calculation

Classifier	WSM average Performance Score Values
Bayes Net	0.75
Naïve Bayes	0.68
Logistic	0.75
SMO	0.66
IBk	0.76
KStar	0.73
MultiClass Classifier	0.80
Random Forest	0.82
RandomTree	0.81
Sample Mean (\bar{X})	0.751
Standard Deviation(S)	0.055
Number of Samples(n)	9
Claim (μ)	0.75
\sqrt{n}	3
S/\sqrt{n}	0.018

One Sample T-Test: $T = (\bar{X} - \mu) / S/\sqrt{n} = 0.060$

Degree of Freedom: 8

Figure 5.90: T-Test Python Program Output

```
[1]: import pandas as pd
    from scipy.stats import ttest_1samp

    # Create a sample dataset (Average of 10-fold, 25-fold and 30%split performance score)
    data = pd.Series([0.75,0.68,0.75,0.66,0.76,0.73,0.80,0.82,0.81])

    # Define the null hypothesis mean
    null_hypothesis_mean = 0.75

    # Perform one-sample t-test
    t_statistic, p_value = ttest_1samp(data, null_hypothesis_mean)

    # For a right-tailed test, just halve the p-value since scipy returns a two-tailed p-value
    one_tail_p_value = p_value / 2

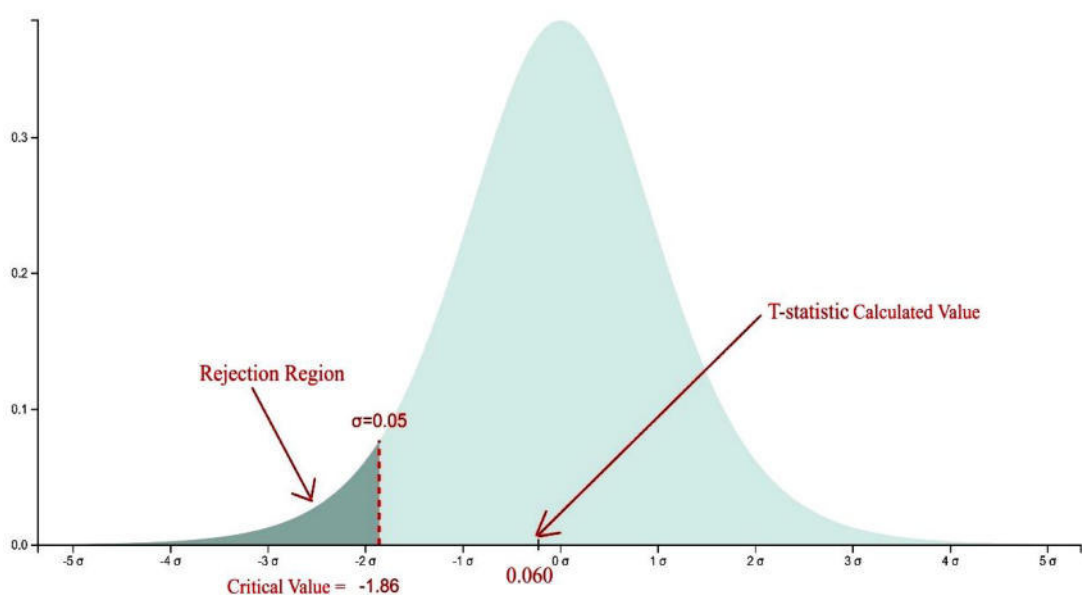
    # Display the results
    print("T-statistic:", t_statistic)
    print("One-tailed P-value:", one_tail_p_value)

    # Interpret the results
    alpha = 0.05
    if one_tail_p_value < alpha:
        print("Reject the null hypothesis - The sample mean is significantly greater than the null hypothesis mean.")
    else:
        print("Fail to reject the null hypothesis - The sample mean is not significantly greater than the null hypothesis mean.")

T-statistic: 0.06024752331288722
One-tailed P-value: 0.47671818579554764
Fail to reject the null hypothesis - The sample mean is not significantly greater than the null hypothesis mean.
```

Result: Table value @ 5% level of significance is -1.860 and the calculated T-statistics value is 0.060 which is larger than the critical T-value of -1.860, Also the p-value found to be 0.476 which is greater than the standard alpha value of 0.05 therefore null hypothesis is failed to be rejected. Therefore, we can say that the Machine learning-based traffic prediction models have average performance score more than or equal to 75%.

Figure 5.91: Left Tailed T-test curve



Source: T-test Distribution Calculator With Graph Generator [29]

From above figure Left Tailed T-test curve it is clear that calculated T-statistic value lies in the acceptance region therefore H_0 is failed to reject and H_a is rejected. This indicates machine learning algorithms are useful in solving traffic congestion problems and generates average performance score of more than 75%.

5.5 Summary

In a comparative analysis of nine machine learning algorithms conducted utilizing the Weka tool, Random Forest and Random Tree emerged as the foremost viable classifiers for anticipating traffic congestion. Bayes Net moreover illustrated solid performance, ranking second respectively.

The investigation included assessing different performance parameters and error measures to evaluate the adequacy of each calculation. Among these measurements, Random Forest and Random Tree reliably beat other classifiers over multiple criteria. Furthermore, two hypotheses were tried during the analysis. The first hypothesis looked for to investigate the differences between Artificial Intelligence, Fog-based, IoT-based technologies, and Machine Learning approaches in traffic forecast models. This hypothesis was rejected and alternate hypothesis was accepted, recommending that there is significant differences between these technologies in terms of their effectiveness for traffic forecast. The second hypothesis aimed to determine if the average performance score of the classifiers surpassed 75%. This hypothesis failed to be rejected, demonstrating that the classifiers achieved satisfactory performance levels more than 75%.

Overall, the discoveries recommend that Random Forest and Random Tree classifiers are well-suited for traffic congestion forecast, with Bayes Net also offering strong performance. Furthermore, the analysis demonstrates that the choice between AI, Fog-based, IoT-based advances, and Machine Learning approaches may essentially affect the effectiveness of traffic prediction models.

Chapter – 6

Conclusion and Future Scope

- 6.1 Summary of Findings
 - 6.1.1 Configuration setting: cross validation – 10 folds
 - 6.1.2 Configuration setting: cross validation – 25 folds
 - 6.1.3 Configuration setting: cross validation – 30% Split
- 6.2 Hypotheses Based Findings
- 6.3 Challenges in Smart Transportation
- 6.4 Future Directions in Smart Transportation

Smart Transportation Systems are increasingly integrating Machine Learning and Artificial Intelligence AI to revolutionize transportation networks. Machine Learning algorithms optimize traffic management by predicting congestion and adapting traffic signals in real time. Predictive maintenance powered by Machine Learning prevents infrastructure failures, saving costs and enhancing safety. AI-driven public transportation planning improves routes and schedules based on dynamic factors. Ride-sharing and mobility services use ML to match riders, optimize routes, and adjust prices, reducing congestion and enhancing user experiences. Autonomous vehicles rely on AI for navigation and obstacle avoidance. Traffic predictions and alerts from ML models aid drivers in choosing optimal routes. AI-driven parking management systems guide drivers to available spaces, reducing congestion. These applications collectively enhance transportation efficiency, safety, and sustainability in smart cities.

6.1 Summary of Findings

Nine machine learning algorithms were analyzed on various performance measures and error measures to predict traffic congestion in smart cities. following seven features/attributes were extracted and selected from twenty one features/attributes for analysis.

1. SPEED
2. NUM_READS
3. HOUR
4. ZIP CODES
5. REGION
6. BUS_COUNT
7. CLASS LABEL

Performance measure parameters like Accuracy, incorrectly classified instances and kappa statistics were used to set up benchmark to compare various machine learning algorithms and to select best algorithm. To gain deeper insights and weaknesses of classification models under consideration, confusion matrix parameters like TP rate, FP rate, precision, recall, F-measure and ROC Area were used for analysis. All these performance measures helped us in selecting the most suitable and accurate prediction model for predicting the traffic congestion in smart cities.

To optimize and tune machine learning models various error measures like mean absolute error, root mean squared error, relative absolute error, root relative absolute error were used for analysis. Error measures gave us quantitative assessment of how well our machine learning model has performed in predicting traffic congestion in smart cities. K-folds and split method Cross-Validation approach is used for evaluating the performance of nine Machine learning Algorithms, where K value is changed to study different cases.

After Cross-Validation approach Machine Learning Algorithms are analysed by vector of features. These features could be measurable characteristics of data. For each feature, a weight is assigned according to the relevance and importance, to understand the significance of that feature and the behaviour of overall outcome on prediction. The Multi-Criteria Decision Making - Weighted Sum Method is used to generate performance score for each machine learning algorithm. The weights assigned to different parameters is shown below.

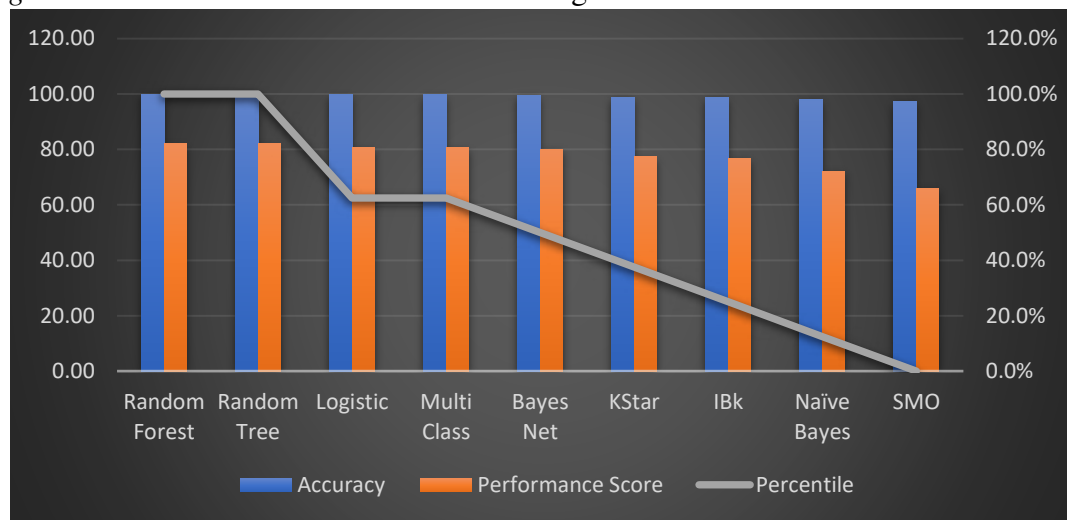
1. Accuracy = 0.3
2. Kappa = 0.2
3. TP Rate = 0.1
4. Precision = 0.1
5. Recall = 0.1
6. F Measure = 0.1
7. ROC Area = 0.1

Ranking and percentile methods was applied to arrange classification algorithms in machine learning based on their performance metrics. The major findings related to the comparative analysis of nine machine learning predictive algorithm models is discussed below with different configuration settings.

6.1.1 Configuration Setting: Cross Validation – 10 folds

- Random Forest and Random Tree are the best and most appropriate classifier for traffic congestion control and traffic flow as both are having the highest accuracy of 100% and performance score of 0.820 with percentile 100%.
- Logistic Regression and MultiClass are the second most appropriate algorithms having accuracy of 99.7% and performance score of 0.807 with percentile of 62.50%.
- The third best classifier being identified is Bayes Net with accuracy of 99.6% and performance score of 0.801 with percentile of 50%.

Figure 6.1: Cross Validation 10 – Fold Findings

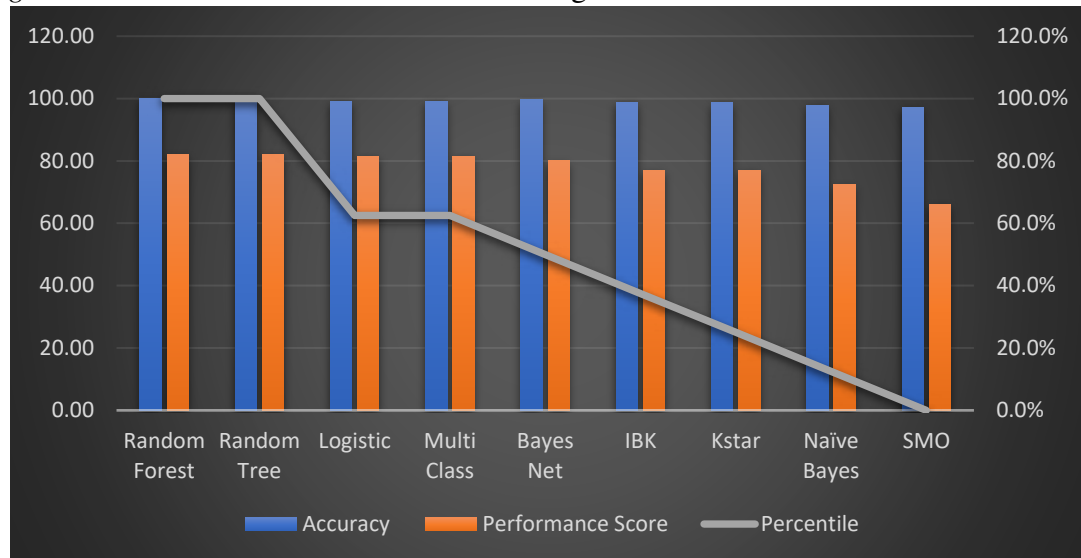


Considering all the factors in Cross Validation – 25 folds, for predicting the Udaipur traffic flow Random Forest is the most appropriate algorithm.

6.1.2 Configuration Setting: Cross Validation – 25 folds

- Random Forest and Random Tree are the best and most appropriate classifier for traffic congestion control and traffic flow as both are having the highest accuracy of 100% and performance score of 0.820 with percentile 100%.
- Logistic Regression and MultiClass are the second most appropriate algorithms having accuracy of 99.1% and performance score of 0.815 with percentile of 62.50%.
- The third best classifier being identified is Bayes Net with accuracy of 99.6% and performance score of 0.801 with percentile of 50%.

Figure 6.2: Cross Validation 25 – Fold Findings

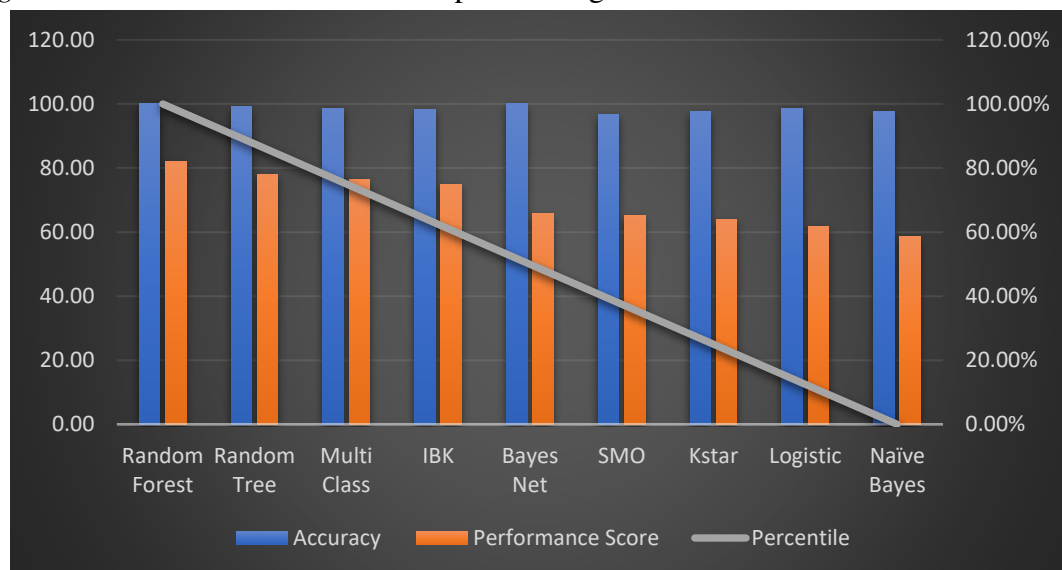


Considering all the factors in Cross Validation – 25 folds, for predicting the Udaipur traffic flow Random Forest is the most appropriate algorithm.

6.1.3 Configuration setting: Cross validation – 30% Split

- Random Forest is the best and most appropriate classifier for traffic congestion control and traffic flow as it is having the highest score of 0.820 with percentile 100%.
- Random Tree is the second most appropriate algorithm having total performance score of 0.779, rank 2 and percentile of 87.50%.
- The third best classifier being identified is Multi Class with total score of 0.765 and percentile of 75%.

Figure 6.3: Cross Validation 30% – Split Findings



Considering all the factors in Cross Validation – 30% Split, for predicting the Udaipur traffic flow Random Forest is the most appropriate algorithm.

By conducting comparative analysis of Cross Validation – 10 folds, Cross Validation – 25 folds and 30% split, it can be concluded that the Random Forest is the best and most appropriate classifier for traffic congestion control and traffic flow and second most appropriate classifier is Random Tree.

6.2 Hypotheses Based Findings

6.2.1 Hypotheses 1

The main objective of Hypothesis 1 is to find out whether technologies like AI, ML, IoT and fog computing can increase the efficiency of transportation system in smart cities. The acceptance of the alternate hypothesis (H_{a1}) indicates a significant difference between technologies used for enhancing transportation systems in smart cities. Following points can be concluded:

- The choice of technology has a discernible impact on the overall performance of transportation systems in smart cities.
- Different technologies like AI, IoT, Fog Computing and ML contribute to diverse approaches in addressing transportation challenges within smart city frameworks.
- The accepted alternate hypothesis suggests that certain technologies are more effective than others in enhancing the efficiency of transportation systems in smart cities.
- Policymakers and urban planners should make strategic decisions regarding technology selection to optimize the functionality and effectiveness of smart city transportation.
- Efficient allocation of resources should consider the technologies that have demonstrated significant differences in enhancing transportation within the smart city context.
- Ongoing evaluation of emerging technologies is crucial to adapt and integrate the most effective solutions for smart city transportation.
- The technology chosen significantly influences the experience of end-users, emphasizing the importance of user-centric design in smart city transportation systems.

- Given the varied impact of technologies, collaboration between technology developers, urban planners, and policymakers is vital to holistically address transportation challenges in smart cities.
- The type of technology used likely has implications for the environmental sustainability of smart city transportation systems, affecting factors such as emissions, energy consumption, and ecological impact.

The acceptance of the alternate hypothesis highlights the need for long-term planning that considers the evolving landscape of transportation technologies within the context of smart cities. The fundamental importance of IoT, Artificial Intelligence, and Machine Learning in addressing commuting challenges within smart cities is underscored. Through the utilization of AI and ML algorithms, these technologies play a crucial role in optimizing resource allocation and effectively managing congestion. This highlights their essential contribution to enhancing the overall efficiency of transportation infrastructure in smart cities. In summary, the transformative impact of IoT, AI, and ML on smart city commuting systems is emphasized, emphasizing their crucial role in creating adaptive, user-friendly, and seamless transportation networks that significantly contribute to overcoming the challenges associated with urban commuting.

6.2.2 Hypotheses 2

The main objective of Hypothesis 2 is to know whether all machine learning algorithms average performance score is more than 75%. The acceptance of hypothesis (H_02) implies that there is significant evidence to suggest that average performance score of machine learning algorithms is very good in predicting traffic congestion in smart cities with fair amount of agreement between them. Following points can be concluded:

- The existing Machine learning based traffic prediction models and traffic control systems exhibit significant differences in terms of their effectiveness. Certain models may outperform others in accurately predicting and managing traffic conditions.
- There is a notable divergence in the technological approaches employed by traffic prediction models and traffic control systems within the smart city infrastructure. This implies that varied technologies are being utilized to address traffic-related challenges.

- The acceptance of the null hypothesis implies an opportunity for using Machine learning based prediction models for optimized performance of traffic flow. Policymakers and technology developers may need to consider improvements or adjustments to enhance the overall efficiency of traffic prediction and control in smart cities.
- Highlights potential challenges in seamlessly integrating Machine learning based traffic prediction models with traffic control systems. It emphasizes the need for careful consideration of compatibility and interoperability to ensure a cohesive and effective smart city traffic management infrastructure.
- Machine learning predictive models in smart transportation systems exhibit sensitivity to different performance metrics. Certain models may excel in specific measures, such as accuracy, precision, recall, or F1 score, suggesting a need for tailored evaluation criteria.
- The diversity in performance measures suggests that the definition of an optimal machine learning predictive model may vary based on the specific goals and priorities of the smart transportation system. Decision-makers should carefully consider the most relevant metrics for their intended outcomes.
- Decision-makers may need to prioritize specific metrics based on the objectives and constraints of the smart transportation system, acknowledging that improvements in one area may come at the expense of another.
- As different performance measures influence the assessment of machine learning models, ongoing evaluation and adaptation become crucial. Smart transportation systems should embrace a dynamic approach to model assessment, adjusting strategies based on evolving performance requirements.
- Adopting multi-criteria optimization strategies that consider various performance measures simultaneously can help identify models that strike a balance across different evaluation criteria.
- In reality, Machine learning contribute to user-centric approaches in smart city transportation. Through personalized recommendations, adaptive routing, and responsive services, commuters are likely to experience a more tailored and hassle-free journey.

In Summary The acceptance of Hypothesis 2 validates the prediction of performance score of all machine learning algorithm more than 75%. It not only validates the effectiveness of all machine learning algorithms but also suggests that the research implementation is proved in getting successful desired outcomes.

6.3 Challenges in Smart Transportation

Overloaded Wireless Networks: Increased device usage for traffic monitoring strains wireless networks, necessitating adaptive routing and data management solutions.

V2V Communication Concerns: Ensuring privacy and security in vehicle-to-vehicle communication requires robust certificate management systems to prevent intrusion and accidents.

Data Collection Complexity: Integrating various sensors in vehicles and transmitting data to network access points poses data collection challenges, requiring clear sensor descriptions and setups.

Data Privacy and Security: Smart transportation systems gather vast amounts of data, including personal information, making data privacy and security critical to prevent hacking and potential harm.

Interoperability Issues: Different technologies using distinct data formats and protocols hinder interoperability, especially in communities lacking the necessary technological expertise.

Costly Implementation: High hardware, software, and infrastructure expenses pose adoption challenges, particularly in smaller cities and villages.

Complex Systems: Smart transportation systems demand expertise in data analytics, artificial intelligence, and IoT technology, making them complex to implement and maintain.

Connectivity Dependence: Reliable data and communication networks are crucial for smart transportation; disruptions can lead to traffic congestion, delays, and safety risks.

Machine Learning Model Implementation: Implementing machine learning models presents several key challenges. Ensuring high data quality and quantity is paramount, as insufficient or low-quality data can lead to inaccurate models. Data privacy and security are significant concerns, especially when handling sensitive information. Model selection requires choosing the most suitable algorithm for a specific problem, and feature engineering involves identifying relevant data attributes. Balancing model complexity to prevent overfitting or underfitting is crucial, as is addressing scalability

for large datasets and computational demands. Making complex models interpretable and explainable remains a challenge, as does transitioning from development to production environments seamlessly. Model maintenance must ensure continued accuracy as data evolves, while managing computational resources efficiently is essential. Addressing bias and ensuring fairness, regulatory compliance, user acceptance, cost management, ethical considerations, fostering a data-driven culture, and addressing the shortage of skilled talent all contribute to the multifaceted landscape of implementing machine learning models.

6.4 Future Directions in Smart Transportation

Data Access and Standardization: Research should focus on improving access to standardized data for government agencies, businesses, and academics, enhancing integration and resilience through backup systems.

Security and Privacy: Addressing cyber threats on transportation infrastructure is vital, with research exploring data security, encryption, access control, and intrusion detection methods.

Autonomous Vehicle Impact: Investigate the effectiveness of autonomous vehicles in reducing traffic congestion, enhancing road safety, and their influence on transportation demand and environmental impact.

Blockchain Technology Integration: Explore the potential of blockchain technology in enhancing transportation security, efficiency, and reliability, creating new applications and use cases.

Accessibility for Disadvantaged Groups: Examine how smart transportation can improve accessibility and mobility for underserved populations like the elderly, disabled, and low-income individuals.

Promoting Green Transportation: Smart transportation systems can prioritize eco-friendly modes such as public transport, cycling, and electric vehicles to reduce greenhouse gas emissions and enhance air quality in urban areas.

Automated Feature Engineering: The future of machine learning will see advancements in automated feature engineering techniques. These methods will streamline the process of identifying and selecting relevant data attributes, reducing the manual effort required for feature engineering and improving model performance.

Explainable AI: As machine learning models become increasingly complex, there will be a growing emphasis on Explainable AI. Future developments will focus on creating more interpretable models and post-hoc explain ability techniques, allowing users to understand and trust AI-driven decisions.

Edge Computing for ML: Edge computing will play a pivotal role in the future of machine learning. With the proliferation of IoT devices and the need for real-time processing, machine learning models will be deployed at the edge, enabling faster decision-making and reduced reliance on centralized data centers.

AI Ethics and Responsible AI: Ethical AI frameworks and guidelines will gain prominence, addressing bias mitigation, fairness, transparency, and accountability. Future developments will prioritize responsible AI practices, ensuring AI systems benefit society without unintended consequences.

Continuous Model Monitoring and Adaptation: To maintain model accuracy over time, continuous model monitoring and adaptation will become standard. This approach will involve real-time data analysis, retraining models as data evolves, and automatic deployment of updated models, ensuring sustained performance and relevance.

Finally, Machine Learning has become instrumental in modern traffic management, revolutionizing urban transportation. ML models excel in predicting traffic patterns by analyzing historical data, real-time information, and weather conditions, enabling proactive congestion management and efficient route suggestions to drivers. Traffic lights are optimized dynamically using ML algorithms, reducing wait times and easing congestion. ML-driven navigation apps provide real-time route planning, considering accidents and closures, thus minimizing travel time and fuel consumption. Additionally, ML is used in parking systems to guide drivers to available spots, reducing search times and traffic congestion, while optimizing public transportation schedules for smoother commuting experiences.

REFERENCES

A. Journals and Books

- Syed and Kumar (2021). Elmaghraby, A. IoT in Smart Cities: “A Survey of Technologies, Practices and Challenges. Smart Cities”, 4, 429–475. <https://doi.org/10.3390/smartcities4020024>
- Moazzami, Majid, Shahvand, Niloufar, Kabalcı, Ersan, Shahinzadeh, Kabalci, Yasin, Gharehpetian and Gevork. (2021). “Internet of Things Architecture for Intelligent Transportation Systems in a Smart City”. 285-290. 10.1109/GPECOM52585.2021.9587692.
- Lakshmi Shankar Iyer (2021). “AI enabled applications towards intelligent transportation, Transportation Engineering”, Volume 5,100083, ISSN 2666-691X, <https://doi.org/10.1016/j.treng.2021.100083>.
- Agarwal, Gurjar, Birla “Application of Artificial intelligence for development of intelligent transport system in smart cities” Int. J. Transp. Eng. Traffic Syst., 1 (2015), pp. 20-30.
- Abbas, S., et al. (2011). “Bio-inspired neuro-fuzzy based dynamic route selection to avoid traffic congestion”. International Journal of Scientific and Engineering Research, 2(6), pp. 284-289.
- Abigail R., Jorge B., Franklin L., Angel S and Santiago M. (2019) “Integrated Information System for Urban Public Transport” International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958, Volume-8 Issue-4C, April 2019 pp. 97 – 102.
- Abduljabbar, R., Dia, H., Liyanage, S., and Bagloee, S. A. (2019). "Applications of artificial intelligence in transport: An overview". Sustainability, pp. 11, 189.
- Al-Dweik, A., Muresan, R., Mayhew, M., and Lieberman, M. (2017). “IoT-based multifunctional Scalable real-time Enhanced Road Side Unit for Intelligent Transportation Systems”. In Electrical and Computer Engineering (CCECE), 2017 IEEE 30th Canadian Conference on, pp. 1-6.
- Albino, V., Berardi, U., and Dangelico, R. M. (2015). “Smart cities: Definitions, dimensions, performance, and initiatives”. Journal of Urban Technology, 22(1), pp. 3-21.
- Ammara, U., Rasheed, K., Mansoor, A., Al-Fuqaha, A., and Qadir, J. (2022). “Understanding the Dynamics and Outcomes of Smart Cities: The Role of Interdependencies and Stakeholder Interactions”. In Proceedings of the International Conference on Smart City Applications, pp. 1-10.

- Amadeo, M., Campolo, C., Quevedo, J., Corujo, D., Molinaro, A., Iera, A., ... & Vasilakos, A. V. (2016). "Information-centric networking for the Internet of Things: Challenges and opportunities". *IEEE Network*, 30(2) pp. 92-100.
- Bai, M., Lin, Y., Ma, M., Wang, P., and Duan, L. (2020). PrePCT: "Traffic congestion prediction in smart cities with relative position congestion tensor. *Neurocomputing*". <https://doi.org/10.1016/j.neucom.2020.08.075> pp. 146-159.
- Balasubramanian, S. B., Balaji, P., Munshi, A., Almukadi, W., Prabhu, T. N., K, V., & Abouhawwash, M. (2023). "Machine learning based IoT system for secure traffic management and accident detection in smart cities". *PeerJ Computer Science*, 9, e1259. <https://doi.org/10.7717/peerj-cs.1259>, pp.39-56.
- Bhardwaj, S., Malik, H., Chauhan, D. S., & Rana, R. (2021). "A comprehensive survey on IoT-based traffic prediction models for smart cities". *Journal of Ambient Intelligence and Humanized Computing*, 12(10), pp. 14873-14892.
- Cai, C. (2009). "Adaptive Traffic Signal Control Using Approximate Dynamic Programming (Ph.D. Thesis)". University College, London, UK.
- Chowdhury, A. (2016). "Priority-based and secured traffic management system for an emergency vehicle using IoT". In *Engineering & MIS (ICE), International Conference on* pp. 1-6.
- Cunha, M., Amaral, A., Silva, A., and Pinheiro, C. (2021). "Smart public transportation for smart cities: A systematic literature review. *Sustainability*", 13(10), 5441. doi:10.3390/su13105441. pp 109-117.
- Collotta, M., Sun, Y., Di Persio, L., Ebeid, E. S. M., & Muradore, R. (2018). "Smart Green Applications: From Renewable Energy Management to Intelligent Transportation Systems. *Energies*", pp.11, 1317.
- Carmona, M. (2010). "Public places, urban spaces: The dimensions of urban design. *Routledge*".
- Chakraborty, A. D. A., Kumar, A., Roy, A., Roy, S., Chakraborty, D., Saha, H. N., et al. (2019). "Intelligent traffic control system: Towards smart city". *IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, <https://doi.org/10.1109/IEMCON.2019.8936188> pp. 1123-1129.
- Chong, H. F., and Ng, D. W. K. (2016). "Development of IoT device for traffic management system". In *Research and Development (SCOREd), 2016 IEEE Student Conference on* pp. 1-6.
- Coutard, O., et al. (2014). "Urban Megatrends: Towards a European research agenda" pp.1-17.

- Dresner, P. S. K. (2008). “A Multiagent approach to Autonomous Intersection Management”. *Journal of Artificial Intelligence Research*, 31, pp. 591–656.
- Dou E. (2015), “AI and ML Based Public transport Information center” University of Wollongong Thesis Collection Chapter 2.3 to 2.6 pp 12 – 30.
- Deak, G., & Walravens, N. (2019). “The future of urban mobility: Towards integrated smart mobility solutions. *Transport Reviews*”, 39(2), pp 157-181.
- Dimitrakopoulos, G., & Bravos, G. (2016). “Current Technologies in Vehicular Communication”. Springer. pp.39-52.
- Doolan, R., & Muntean, G.-M. (2016). “EcoTrec—A novel VANET-based approach to reducing vehicle emissions”. *IEEE Transactions on Intelligent Transportation Systems*, 18(3), pp.608–620.
- Eken S. and Sayar A. (2014) “A smart bus tracking system based on location-aware services and QR codes,” 2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings, pp. 299-303
- Erlmann, S., & Dantzig, S. V. (2018). Exploring the potential of shared mobility services to alleviate urban traffic congestion: A literature review. *Transport Reviews*, 38(6), pp. 769-793.
- Ersoy, P., and Börühan, G. (2015). “Intelligent Transportation Systems and Their Applications in Road Transportation Industry in Turkey”. In *Proceedings of the 12th International Conference on Logistics & Sustainable Transport*, Celje, Slovenia, pp. 11–13 June 2015.
- Ferrara, A., Sacone, S., and Siri, S. (2018). “Fundamental of traffic dynamics. In *Freeway traffic modelling and control*” (Advances in Industrial Control Series, Vol. 1185, pp. 44).
- Fintikakis, N., & Bourka, A. (2018). “Commuting patterns and smart cities: A comparative study of Stockholm and London”. *TeMA Journal of Land Use, Mobility and Environment*, 11(3), pp. 239-254.
- Gkiotsalitis, K., & Cats, O. (2018). “Assessing the impact of autonomous vehicles on our cities. *Transportation Research Part A: Policy and Practice*”, pp. 118, 407-417.
- Gohar, M., Sagheer, A., Javaid, N., Anpalagan, A., & Khan, F. (2019). “Fog computing-enabled intelligent transportation system: Architecture, opportunities, and challenges”. *IEEE Access*, 7, pp. 29037-29057.
- Giffinger, R., and Gudrun, H. (2010). “Smart cities ranking: An effective instrument for the positioning of cities? *ACE Architecture, City and Environment*”, 4(12), pp. 7–25.

- Ghasem-Aghaee, N., Tuncer Oren, and Levent Yilmaz. (2019). "Simulation, Intelligence and Agent: Exploring the Synergy. In *Current and Future Developments in Artificial Intelligence*". Bussum, The Netherlands: Bentham Science Books Publisher. pp.1-58.
- Goggin, G. (2012). "Driving the internet: Mobile internets, cars, and the social. *Future Internet*", 4(1), pp. 306-321.
- Hall, R. E., and Pfeiffer, U. (2016). "The impact of smart cities on travel behavior: An exploratory study. *Transportation Research Part A: Policy and Practice*", 88, pp. 96-114.
- Han, Q., Liang, S., & Zhang, H. (2015). "Mobile cloud sensing, big data, and 5G networks make an intelligent and smart world". *IEEE Network*, 29(2), pp. 40-45.
- Iker, Z., Alessandro, S., & Saioa, A. (2016). "Smart city concept: What it is and what it should be. *Journal of Urban Planning and Development*", 142(1), 04015005. pp. 178-189.
- Iyer L. (2021) "AI enabled applications towards intelligent transportation" *Transportation Engineering* 5 (2021) 100083 open access article under the CC BY-NC-ND license Published by Elsevier Ltd. pp 1 – 11.
- Jacobsen, R. H., Aliu, D., & Ebeid, E. (2017). "A Low-Cost Vehicle Tracking Platform Using Secure SMS". In *Proceedings of the 2nd International Conference on Internet of Things, Big Data and Security*, Porto, Portugal, pp. 24–26.
- Jacobsen, R. H., Gabioud, D., Basso, G., Alet, P. J., Azar, A. G., & Ebeid, E. (2015). "SEMIAH An Aggregator Framework for European Demand Response Programs". In *Proceedings of the Euromicro Conference on Digital System Design (DSD)*, Madeira, Portugal, 26–28 August 2015; pp. 470–477.
- Jarrah M. and Shrida F. (2017) "A multi-objective evolutionary solution to improve the quality of life in smart cities," 2017 14th International Conference on Smart Cities: Improving Quality of Life Using ICT and IOT (HONET-ICT), pp 36 - 39.
- Jung, C., and Couclelis, H. (2017). "Smart cities and the politics of urban data". *Journal of Planning Education and Research*, 37(4), pp. 457-466.
- Janahan, S. K., Veeramanickam, M., Arun, S., Kumar, N., Anandan, R., & Javed, S. (2018). "IoT-based smart traffic signal monitoring system using vehicle counts". *International Journal of Engineering and Technology*, <https://doi.org/10.14419/ijet.v7i2.21.12388> pp. 7,309.
- Jin, J., Ma, X., & Kosonen, I. (2017). "An intelligent control system for traffic lights with simulation-based evaluation. *Control Engineering Practice*," <https://doi.org/10.1016/j.conengprac.2016.09.009> pp. 58, 24-33.

- Joo, H., Ahmed, S. H., & Lim, Y. (2020). "Traffic signal control for smart cities using reinforcement learning. *Computer Communications*". <https://doi.org/10.1016/j.comcom.2020.03.005> pp. 111 - 124
- Kaur, A., and Kaur, P. (2017). "Internet of Things for transportation: A review". *IEEE Sensors Journal*, 17(14), pp.4371-4378.
- Khan, A., Salah, K., and Zeadally, S. (2019). "Fog computing for intelligent transportation systems: Opportunities and challenges. *Computer Networks*," 160, pp. 96-112.
- Khallouk, A., Echab, H., Ez-Zahraouy, H., and Lakouari, N. (2018). "Traffic flow behaviour at Un-signalized intersection with crossing pedestrians. *Physics Letters A*," 382, pp. 566–573.
- Kikuchi, S. (2009). "Artificial intelligence in transportation analysis: Approaches, methods, and applications". *Transportation Research Part C*, 17(5), pp. 455.
- Kuberkar and Singhal (2020) "Factors Influencing Adoption Intention of AI Powered Chatbot for Public Transport Services within a Smart City" *International Journal on Emerging Technologies* 11(3): 948-958(2020) ISSN No. (Print): 0975-8364 ISSN No. (Online): 2249-3255 pp. 948 – 958.
- Kumar, N., Rahman, S. S., & Dhakad, N. (2020). "Fuzzy inference enabled deep reinforcement learning-based traffic light control for intelligent transportation systems". *IEEE Transactions on Intelligent Transportation Systems*. <https://doi.org/10.1109/TITS.2020.2984033> pp. 147-169
- Kitchin, R. (2014). "Big Data, new epistemologies and paradigm shifts." *Big Data & Society*, 1(1), 2053951714528481. pp. 45-63.
- Lakshmi, I. (2016). "A literature survey on Big Data Analytics in Service Industry". *International Journal of Engineering and Computer Science*, 5(4).
- Lee, R., Blogg, M., Myers, E., Kyte, M., Dixon, M., List, G., Flannery, A., Troutbrck, R., Brilon, W., Werner, B., et al. (2007). "National Cooperative Highway Research Program (NCHRP)" Report 572 (Vol. 9). Washington, DC, USA: Transportation Research Board of the National Academics. pp. 327-346.
- Lee, W.-H., & Chiu, C.-Y. (2020). "Design and implementation of a smart traffic signal control system for smart city applications". *Sensors*, 20(2), <https://doi.org/10.3390/s20020508> pp. 508
- Liu, J., Li, J., Niu, X., Cui, X., & Sun, Y. (2014). GreenOCR: "An Energy-Efficient Optimal Clustering Routing Protocol". *Computer Journal*, 58, pp. 1344–1359.

- Lingani, G. M., Rawat, D. B., & Garuba, M. (2019). “Smart traffic management system using deep learning for smart city applications”. IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), <https://doi.org/10.1109/CCWC.2019.8666539> pp. 0101–0106.
- Mamoonah Humayun, Sadia Afsar, Maram Fahaad Almufareh, N. Z. Jhanjhi, Mashayel AlSuwailem, "Smart Traffic Management System for Metropolitan Cities of Kingdom Using Cutting Edge Technologies", *Journal of Advanced Transportation*, vol. 2022, Article ID 4687319,13. <https://doi.org/10.1155/2022/4687319>. pp. 197-213.
- Marko Pavelić, Zvonimir Lončarić, Marin Vuković, Mario Kušek. "Internet of Things Cyber Security: Smart Door Lock System". 978-1-5386-7189-4/18. pp. 213-222.
- Misbahuddin, S., Zubairi, J. A., Saggaf, A., Basuni, J., Sulaiman, A., and Al-Sofi, A. (2015). “IoT Based Dynamic Road Traffic Management for Smart Cities”. In *Proceedings of the 2015 12th International Conference on High-Capacity Optical Networks and Enabling/Emerging Technologies (HONET)*, Islamabad, Pakistan, 21–23 December 2015; pp. 142–146.
- Mukti, I. Y., & Prambudia, Y. (2018). “Challenges in Governing the Digital Transportation Ecosystem in Jakarta: A Research Direction in Smart City Frameworks”. *Challenges*, pp. 9, 14.
- M. S. Kamal, S. Parvin, K. Saleem, H. Al-Hamadi, and A. Ghawanmeh. “Efficient low cost supervisory system for internet of things enabled smart home”. In *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 864–869.
- Misuraca, G., Mureddu, F., & Osimo, D. (2014). “Policy-making 2.0: Unleashing the power of big data for public governance”. *Open Government*, Springer, pp. 171-188.
- Mondal, M. A., & Rehena, Z. (2019). “Intelligent traffic congestion classification system using artificial neural network”. *Companion Proceedings of the World Wide Web Conference*, <https://doi.org/10.1145/3308560.3317053> pp. 110–116.
- Malek, G., Li, C., Yang, Z., Hasan, N. A. H., & Zhang, X. (2012). “Improved the Energy of Ad hoc On-Demand Distance Vector Routing Protocol”. In *International Conference on Future Computer Supported Education*, Published by Elsevier, IERI, pp.355-361.
- Neelakandan, S. (2016). “Large scale optimization to minimize network traffic using MapReduce in big data applications”. In *International Conference on Computation of Power, Energy Information and Communication (ICCPEIC)* <https://doi.org/10.1109/ICCPEIC.2016.7557196> pp.193-199.

- Neelakandan, S., & Paulraj, D. (2020). "An Automated Exploring And Learning Model For Data Prediction Using Balanced CA-Svm". *Journal of Ambient Intelligence and Human Computing*, 1-12. Springer, Berlin pp.1868–5137.
- Nasr, E., Kfoury, E., and Khoury, D. (2016). "An IoT approach to vehicle accident detection, reporting, and navigation". In *Multidisciplinary Conference on Engineering Technology (IMCET)*, IEEE International, pp. 231-236.
- Neha Patil, Shrikant Ambatkar and Sandeep Kakde. "IoT Based Smart Surveillance Security System using Raspberry Pi" *International Conference on Communication and Signal Processing*, April pp. 6-8.
- Navarathna, Pramathi J., and Vindhya P. Malagi. "Artificial intelligence in smart city analysis." *2018 International conference on smart systems and inventive technology (ICSSIT)*. IEEE, 2018. pp. 96-104.
- Ou, H., Zhang, J., & Wang, Y. (2016). "Development of intelligent traffic control system based on Internet of Things and FPGA technology in Proteus. Traffic", pp. 20-22.
- Pattanaik, V., Singh, M., Gupta, P. K., & Singh, S. K. (2016). "Smart real-time traffic congestion estimation and clustering technique for urban vehicular roads". In *IEEE Region 10 Conference (TENCON)* , pp. 3420-3423.
- Prakash N. and Anudeep P. (2018), "Intelligent Passenger Information System Using IoT for Smart Cities" *AISC*, volume851 pp. 67 – 76.
- Pradhan M. and Kumar U. (2019), "Machine Learning using Python", *WILEY publication First Edition*, pp. 179 – 181.
- Pereira, F., Rodrigues, J. J., and Saleem, K. (2018). "IoT-based transportation systems: A survey on applications, challenges, and opportunities". *Journal of Network and Computer Applications*, 110, pp. 40-54.
- Pau, G., Campisi, T., Canale, A., Severino, A., Collotta, M., and Tesoriere, G. (2018). "Smart Pedestrian Crossing Management at Traffic Light Junctions through a Fuzzy-Based Approach". *Future Internet*, pp.10, 15.
- Psomakelis, E., et al. (2012). "Big IoT and Social Networking Data for Smart Cities" pp. 176 - 193
- Rahman A. and Ajala (2018) "Conceptualising Smart City for the Development of Nigeria's Urban Transportation" *Advances In Multidisciplinary and Scientific Research Vol. 4 No. 2, June 2018* pp. 65 – 72.
- Raj G Anvekar and Rajeshwari M Banakar. "IoT application development: Home security system". *2017 IEEE International Conference on Technological Innovations in ICT For Agriculture and Rural Development (TIAR 2017)* pp.159-172.

- Ravi Kishore Kodali, Vishal Jain, Suvadeep Bose, Lakshmi Boppana. "IoT based smart security and home automation system". International Conference on Computing, Communication and Automation (ICCCA2016) pp. 188-192.
- Rizwan, P., Suresh, K., and Rajasekhara Babu, M. (2016). "Real-time smart traffic management system for smart cities by using the Internet of Things and big data". In Emerging Technological Trends (ICETT), International Conference on, pp. 1-7.
- Ramachandra, S. H., Reddy, K. N., Vellore, V. R., Karanth, S., & Kamath, T. (2016). "A novel dynamic traffic management system using on-board diagnostics and Zigbee protocol". In Communication and Electronics Systems (ICCES), International Conference on pp. 1-6.
- Rego, A., Garcia, L., Sendra, S., & Lloret, J. (2018). "Software-defined Network-based control system for efficient traffic management for emergency situations in smart cities". Future Generation Computer Systems, pp. 88, 243-253.
- Satpathy, S., Mohan, P., Das, S., et al. (2020). "A new healthcare diagnosis system using an IoT-based fuzzy classifier with FPGA". Journal of Supercomputing, <https://doi.org/10.1007/s11227-019-03013-2> pp. 76, 5849-5861.
- Saeed, Y., et al. (2016). "Impact of Cognition on User Authentication Scheme in Vehicle using Fuzzy Logic and Artificial Neural Network". International Journal of Computer Science and Information Security, pp.14(10), 285.
- Sundar G. (2017), "IoT based Passenger Information System Optimized for Indian Metros" International Conference on Electronics, Communication and Aerospace Technology ICECA 2017 pp. 92 – 96.
- Sarrab, M., Pulparambil, S., & Awadalla, M. (2020). "Development of an IoT based real-time traffic monitoring system for city governance". Global Transitions, 2 <https://doi.org/10.1016/j.glt.2020.09.004>. pp. 230-245.
- Silva, F. M., & Silva, R. F. (2019). "Congestion charging and smart cities: A systematic review". Transportation Research Part A: Policy and Practice, 121, pp. 331-350.
- Sochor, J., & Dvorak, J. (2020). "Impact of smart city solutions on urban mobility: A systematic literature review. Sustainability," 12(7), 2771 pp. 314-323.
- Subramani, Neelakandan & Berlin, M. & Tripathi, Sandesh & Devi, V. & Bhardwaj, Indu & Natarajan, Arulkumar. (2021). "IoT-based traffic prediction and traffic signal control system for smart city. Soft Computing." 25. 10.1007/s00500-021-05896-x. pp. 152-160.

- Sharma, V., You, I., Pau, G., Collotta, M., Lim, J. D., & Kim, J. N. (2018). “LoRaWAN-Based Energy-Efficient Surveillance by Drones for Intelligent Transportation Systems”. *Energies*, pp.11, 573.
- Sun, Y., and Song, H. (2017). “Secure and Trustworthy Transportation Cyber-Physical Systems”. Springer. pp.103-121.
- Subbulakshmi, P. and Prakash, M. (2018). “Mitigating eavesdropping by using fuzzy-based MDPOP-Q learning approach and multilevel Stackelberg game theoretic approach in wireless CRN”. *Cognitive Systems Research*, pp. 52, 853-861.
- Tarawneh, M., Alzyoud, F., & Sharrab, Y. (2023). “Artificial Intelligence Traffic Analysis Framework for Smart Cities”.
- Tchuitcheu, W. C., Bobda, C., & Pantho, M. J. H. (2020). “Internet of smart cameras for traffic lights optimization in smart cities. Internet of Things”. <https://doi.org/10.1016/j.iot.2020.100207> pp. 32-47.
- Thakur, T. T., Naik, A., Vasari, S., & Gogate, M. (2016). “Real-time traffic management using the Internet of Things”. In *Communication and Signal Processing (ICCSP), 2016 International Conference on* pp. 1950-1953.
- Theodoridis, E., Mylonas, G., & Chatzigiannakis, I. (2013). “Developing an IoT smart city framework”. In *Information, Intelligence, Systems and Applications (LISA), 2013 Fourth International Conference on* pp. 1-6.
- Troutbeck, R. (2016). “Modelling Signalized and Un-Signalized Junctions”. In *Handbook of Transport Modelling* (pp. 443–460). Bingley, UK: Emerald Group Publishing Limited.
- Vakula D. and Raviteja B. (2017), “Smart Public Transport for Smart Cities” *Proceedings of the International Conference on Intelligent Sustainable Systems (ICISS 2017) IEEE Xplore Compliant - Part Number:CFP17M19-ART, ISBN:978-1-5386-1959-9* pp. 805 – 810.
- Vlahogianni, E. I., Karlaftis, M. G., & Golias, J. C. (2014). “Short-term traffic forecasting: Where we are and where we're going”. *Transportation Research Part C: Emerging Technologies*, 43, pp. 3-19.
- Wen, W. (2008). “A dynamic and automatic traffic light control expert system for solving the road congestion problem”. *Expert Systems with Applications*, 34, pp. 2370–2381.
- World Health Organization. (2018). “Global Status Report on Road Safety 2018”. Geneva, Switzerland: World Health Organization.

- Yao, H., Gao, P., Wang, J., Zhang, P., Jiang, C., & Han, Z. (2019). “Capsule network assisted IoT traffic classification mechanism for smart cities”. *IEEE Internet of Things Journal*, 6(5), pp. 7515-7525.
- Zafar, N., Ul Haq, I., Sohail, H., Chughtai, J.-U.-R., & Muneeb, M. (2022). “Traffic prediction in smart cities based on hybrid feature space”. *IEEE Access*, 10, <https://doi.org/10.1109/ACCESS.2022.3231448>. pp.134333-134348.
- Zanella, A., Bui, N., Castellani, A., Vangelista, L., & Zorzi, M. (2014). “Internet of Things for smart cities”. *IEEE Internet of Things Journal*, 1(1), pp. 22-32.
- Zhang, J., Wang, X., & He, J. (2020). “Fog computing-based intelligent transportation systems: Challenges and solutions”. *IEEE Transactions on Intelligent Transportation Systems*, 21(7), pp. 2909-2924.
- Zhou, H., Wang, X., Ma, J., Zhang, J., & Ma, L. (2020). “An IoT-based traffic prediction model for smart cities using deep learning algorithms. *Sensors*,” 20(6), 1763. pp. 142-149.
- Kumar, Ratan. (2023). “A research paper on AI for traffic management”. 10.13140/RG.2.2.21119.69287.
- Sreelatha, Mahalakshmi and Yadav(2023). “Artificial Intelligence Based Autonomous Traffic Regulator”, *Proceedings of CEUR Workshop Proceedings*, New Delhi, India.

B. Websites

- [1] <https://www.slideshare.net/paojean2000/urban-transport-problems>
Accessed on 19 - 08 - 2021
- [2] <https://rideamigos.com/smart-mobility-in-smart-cities/> Accessed on
19 - 08 – 2021
- [3] <https://seoulsolution.kr/en/content/7664> /Accessed on 19 - 08 - 2021
- [4] <http://ethesis.usm.my/jspui/handle/123456789/12348> Accessed on
19 - 08 – 2021
- [5] <https://www.digiteum.com/iot-data-collection/> Accessed on
19.08.2021
- [6] <https://www.intelligenttransport.com/transportarticles/21458/city-public-transportation-india/> Accessed on 19.08.2021
- [7] <https://www.economicdiscussion.net/sales/sales-forecasting-methods/32270>
Accessed on 15.09.2021

- [8] <https://iq.opengenus.org/advantages-and-disadvantages-of-linear-regression/>
Accessed on 15.09.2021
- [9] <https://towardsdatascience.com/prediction- engineering-how-to-set-up-your-machine- learning-problem-b3b8f622683b/> Accessed on 15.09.2021
- [10] <https://www.marketresearch.com/Schonfeld-Associates-Inc-v417 Ratios- Budgets-13373044/> Accessed on 20.09.2021
- [11] <https://www.keboola.com/blog/linear- regression-machine-learning />
Accessed on 20.09.2021
- [12] <https://internal.ncl.ac.uk/ask/numeracy-maths-statistics/statistics/regression- and- correlation/coefficient-of-determination-squared.html /> Accessed on 27.09.2021
- [13] <http://core.ecu.edu/psyc/wuenschk/docs30/EffectSizeConventions.pdf />
Accessed on 27.09.2021
- [14] <https://www.obviously.ai/post/the-difference-between-training-data-vs-test- data-in- machine-learning /> Accessed on 25.02.2022
- [15] <https://data.cityofchicago.org/> Accessed on 20.03.2023
- [16] <https://chicagotraffictracker.com/> Accessed on 10.09.2023.
- [17] https://data.cityofchicago.org/Transportation/Chicago-Traffic-Tracker- Historical Congestion-Esti/kf7e-cur8/about_data/ Accessed on 10.09.2023.
- [18] <https://www.tomtom.com/products/route-monitoring/> Accessed on 01.10.2023.
- [19] <https://www.tomtom.com/products/traffic-stats/> Accessed on 01.10.2023.
- [20] <https://www.tomtom.com/products/junction-analytics/> Accessed on 12.10.2023.
- [21] https://en.wikipedia.org/wiki/Feature_selection / Accessed on 20.02.2023
- [22] <https://towardsdatascience.com/feature-selection-and-dimensionality- reduction-f488d1a035de /> Accessed on 25.05.2023
- [23] <https://towardsdatascience.com/understanding-confusion-matrix- a9ad42dcfd62 /> Accessed on 10.06.2023.

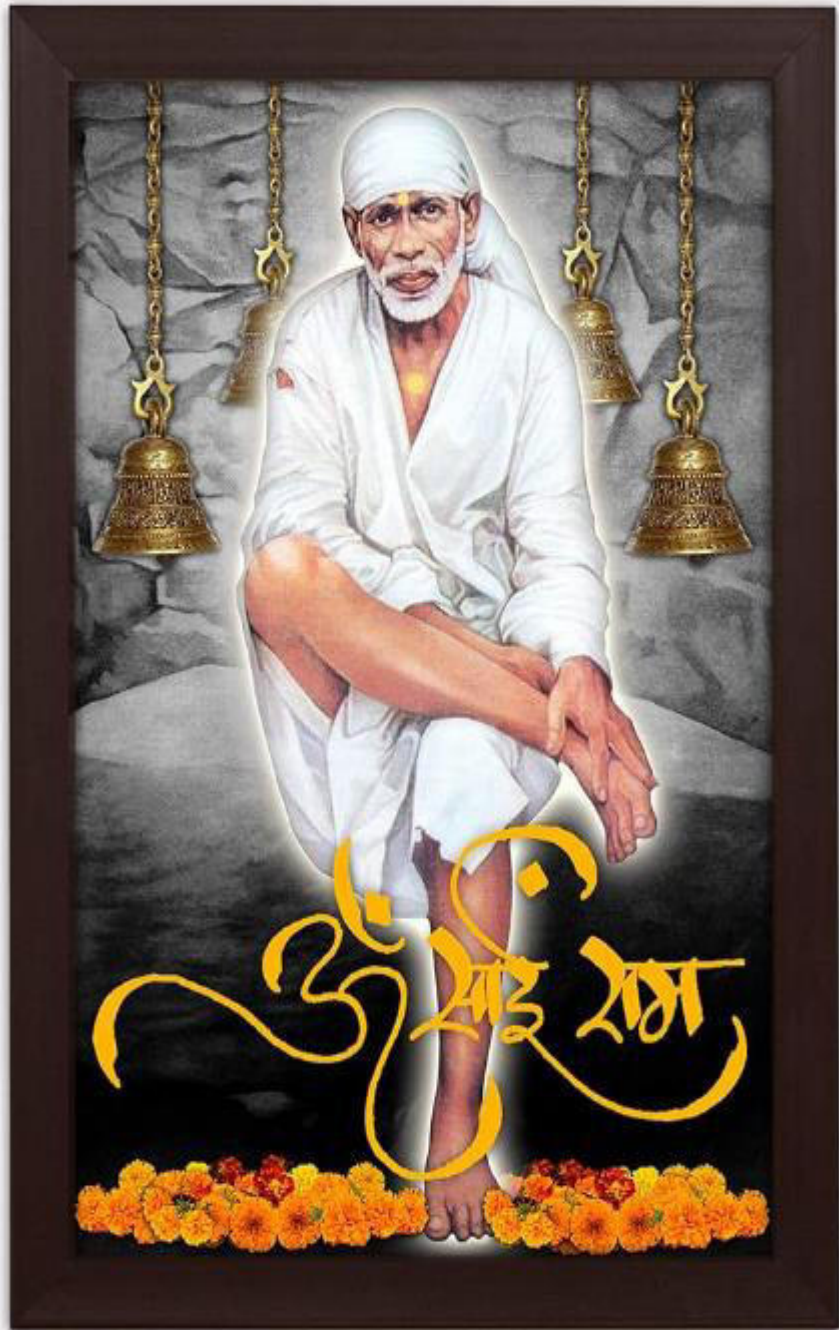
- [24] [https://c3.ai/glossary/machine-learning/precision/#:~:text= Precision%20is %20one%20indicator%20of, the%20number%20of%20false %20positives /](https://c3.ai/glossary/machine-learning/precision/#:~:text=Precision%20is%20one%20indicator%20of,the%20number%20of%20false%20positives/) Accessed on 12.06.2023
- [25] [https://www.shiksha.com/online-courses/articles/sensitivity-vs-specificity/ #:~: text= Sensitivity%20measures%20how%20well%20a, have%20 significantly%20fewer%20False%20Negatives. /](https://www.shiksha.com/online-courses/articles/sensitivity-vs-specificity/#:~:text=Sensitivity%20measures%20how%20well%20a,have%20significantly%20fewer%20False%20Negatives.)Access on 12.06.2023
- [26] [https://link.springer.com/referenceworkentry/10.1007/978-1-4419-9863-7255#:~: text=Definition,positives%20which%20are%20correctly %20identified /](https://link.springer.com/referenceworkentry/10.1007/978-1-4419-9863-7255#:~:text=Definition,positives%20which%20are%20correctly%20identified/) Accessed on 28.07.2023
- [27] [https://developers.google.com/machine-learning/crash-course/ classification /roc-and-auc /](https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc/)Accessed on 30.07.2023
- [28] [https://www.koshegio.com/chi-square-online-calculator /](https://www.koshegio.com/chi-square-online-calculator/) Accessed on 18.12.2023
- [29] <https://www.koshegio.com/t-test-distribution-free-calculator/>Accessed on 20.12.2023
- [30] <https://www.koshegio.com/f-distribution-calculator-graph/>Accessed on 20.12.2023

DISSEMINATION OF RESEARCH WORK

[P1] Avinash Dangwani, Dr. Ashok kumar Jetawat and Dr. Jayshree Jain (2023), “ Exploring the Role of Machine Learning Algorithms for Smart Commuting in Smart Cities”, 38th Indian Engineering Congress Conference of The Institution of Engineering(IEI) India, held at Jabalpur 27-29 December 2023.

[P2] Avinash Dangwani, Dr. Ashok kumar Jetawat and Dr. Chandansingh Rawat (2023), “Use of AI in Cloud based Certificate Authentication for Travel Concession”, 4th International Springer Conference on Mobile Computing and Sustainable informatics(ICMSCI 2023) Organized by Tribhuvan University Nepal, 11 – 12 January 2023. Research paper published in Lecture Notes on Data Engineering and Communications Technologies Springer Nature Singapore in May 2023 ISSN 2367-4512, 2367-4520 (electronic), ISBN 978-981-99-0834-9, 978-981-99-0835-6 (eBook). Indexed by SCOPUS, INSPEC, EI Compendex and UGC Care List Group – II.

[P3] Avinash Dangwani and Dr. Chandansingh Rawat (2021), “Data Analytics Sales Prediction Model“, 12th International Conference on Sustainable Global Trends: Planet, People and Profit by Pacific Institute of Management and Pacific Business School Udaipur, 16 – 17 April 2021. Research paper published in Pacific Business Review International Journal in Aug 2021 ISSN: 0974-438X August, 2021. Indexed by Web of Science Group and UGC Care List Group – II.



Annexure

HYPOTHESIS -1 SURVEY QUESTIONS

1. Age of Participant.
2. Gender
3. City of Residence:
4. Are you aware of the concept of smart cities ?
 - a) Yes
 - b) No
5. Have you used any smart transportation technologies in your city ?
 - a) Yes
 - b) No
6. Which of the following smart transportation technologies being used ?
 - a) AI-Based Systems
 - b) Fog Computing
 - c) IoT-Based Traffic Prediction Models
 - d) Machine Learning-Based Traffic Prediction Models
7. How would you rate the reliability of public transportation in your city ?
 - a) Low
 - b) Moderate
 - c) High
 - d) Very High
8. How would you rate the effectiveness of smart transportation technologies in enhancing the transportation system for smart cities ?
 - a) Low
 - b) High
9. Do you have any additional comments or suggestions regarding smart transportation technologies in smart cities ?

Published Research Papers



The Institution of Engineers (India)

Established 1920, Incorporated by Royal Charter 1935

A Century of Service to the Nation

38th Indian Engineering Congress

December 27-29, 2023



Certificate of Paper Presentation

This is to certify that

Hinash Danguwani, Ashok Kumar Jetawat, Jayshree Jain

Department of Computer, Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan

Have participated in the 38th Indian Engineering Congress held at Jabalpur during December 27-29, 2023 and presented a paper titled '**Exploring the Role of Machine Learning Algorithms for Smart Commuting in Smart Cities**'.

Maj Gen (Dr) MJS Syali, VSM (Retd)
Secretary & Director General

PRESIDENT

Technical Volume



38th

Indian Engineering Congress

December 27-29, 2023

Venue: Hotel Royal Orbit, Jabalpur

Theme

**Reimagining Tomorrow:
Shaping the Future
through Disruptive
and Interdisciplinary
Technologies**

Organised by



The Institution of Engineers (India)

8 Gokhale Road, Kolkata

Hosted by

Jabalpur Local Centre

Reimagining Tomorrow: Shaping the Future through Disruptive and Interdisciplinary Technologies

CP IEC 2023 TV 123	117
Enhancing Spam Detection with Machine Learning Aditi Singh Bais, Aisha Iqbal Haqqani, Vimmi Pandey and Kamaljeet Singh Kalsi	
CP IEC 2023 TV 128	125
Cognitive Machine Learning for Personality Analysis from Handwriting Lakshmi Durga, Deepu Rand Vidyashree K	
CP IEC 2023 TV 139	139
Fake News Detection Using Machine Learning: Where Technology Meets Truth in a World of Misinformation Prabhat Gautam, Vimmi Pandey and Kamaljeet Singh Kalsi	
CP IEC 2023 TV 148	145
Detection and Identification of Objects to Enhance Traffic Sign Recognition using Deep Convolutional Neural Networks Poonam Bhartiya, Mukta Bhatele and Akhilesh A Wao	
CP IEC 2023 TV 156	155
Digital Transformation: Pioneering Circular Economy in Engineering Labh Singh Bhari	
CP IEC 2023 TV 168	159
An Effective DWT SVD Based Watermarking Technique on Colored DICOM Image in Healthcare Applications Saurabh Verma and Mukta Bhatele	
CP IEC 2023 TV 169	167
Deep Learning Models-based Histopathological Image Classification for Automated Screening of Breast Cancer Arifa Anjum, Akhilesh A Wao and Mukta Bhatele	
CP IEC 2023 TV 171	177
An Efficient and Reliable Algorithm for Wireless Sensor Network Prashant Chaudhary, Akhilesh A Wao and Virendra Tiwari	
CP IEC 2023 TV 172	189
An Extensive Study for a Wide Utilization of Green Architecture Parameters in Built Environment Based on Genetic Schemes. Nitesh Kushwaha, Akhilesh A Wao and Ashwini A Wao	
CP IEC 2023 TV 173	199
AI Enabled Human Computer Interaction Dipti Y Patil and Mukta Bhatele	
CIVIL ENGINEERING DIVISION	
CV IEC 2023 TV 011	205
Exploring the Role of Machine Learning Algorithms for Smart Commuting in Smart Cities Avinash Dangwani, Ashok Kumar Jetawat and Jayshree Jain	
CV IEC 2023 TV 036	215
Bamboo Research and Training Centre MohdMueez Khan, Tobin Nainan and Abhay Gupta	

Exploring the Role of Machine Learning Algorithms for Smart Commuting in Smart Cities

Avinash Dangwani*, Ashok Kumar Jetawat and Jayshree Jain

Department of Computer, Pacific Academy of Higher Education and Research University, Udaipur

✉ avin861@gmail.com*

Abstract: Cities are developing with boundless boundaries, Policymakers and urban organizers are investigating ways to address smart Infrastructure in smart cities such as smart water management, Environmental monitoring, Green spaces, smart waste management, smart commuting, smart solution to traffic congestion and hassle-free day-by-day transportation. Smart commuting may be a key component of the smart framework, which points to making strides in efficiency, security, sustainability, and by and large quality of life in urban and country regions. Smart commuting leverages innovation and data-driven solutions to upgrade different angles of the transportation framework. Development in innovation such as IoT, Artificial intelligence, and Machine learning has invented various solutions for smart commuting in smart cities. Smart cities contribute capital in the present day and effective public transportation frameworks, including buses, metro, and rail systems. These frameworks regularly incorporate Traffic Monitoring, real-time tracking, ticket sales prediction, remote door unlocking, seat availability prediction and other advanced innovations to make commuting more helpful. Our paper investigate and discuss various ML algorithms and techniques to address traffic congestion problems in smart cities using traffic prediction methods.

Keywords: Classifiers; tomtom; Precision; Recall; TP Rate; FP Rate; Bayes; Naïve Bayes; Random Forest.

INTRODUCTION

Machine learning algorithms can be very effective in designing models for solving traffic congestion problems. They can help with various aspects of traffic management and optimization by analyzing historical data, making predictions, and making recommendations.

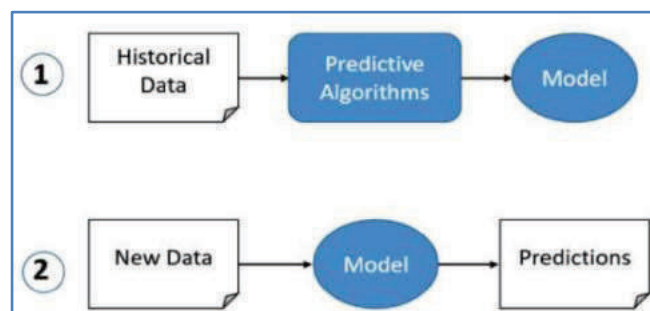


Figure 1 Model Devising

Some of the common cases are Traffic predictions, Traffic flow management, Demand Management, Parking optimization. Traffic prediction and control systems in smart cities are essential for managing urban congestion and improving overall transportation efficiency. Various machine learning and IoT-based models have been developed to address these challenges. Some of the existing approaches in this field are shown below.

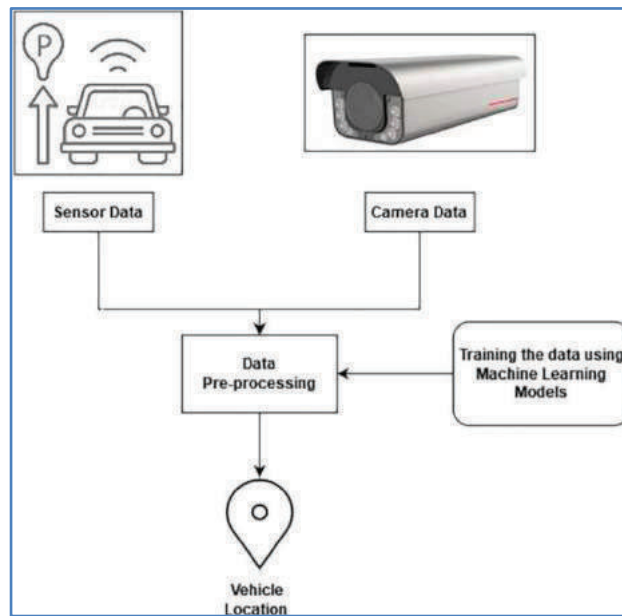


Figure 2 Data Processing Model (Courtesy : peerj.com)

IoT devices and sensors deployed across road networks can collect real-time data on traffic flow, vehicle speeds, and occupancy. By analysing this data, traffic prediction models can provide accurate and up-to-date traffic forecasts. Using real-time traffic data collected from IoT sensors, adaptive traffic signal control systems can adjust signal timings based on current traffic conditions. These systems aim to minimize congestion and improve traffic flow efficiency. By combining machine learning-based traffic prediction models with control algorithms, predictive traffic control systems can anticipate traffic conditions and adjust signal timings proactively. These systems help prevent traffic jams before they happen. Predictive machine learning models for smart transportation systems integrate data from various sources, including traffic cameras, GPS devices, and weather forecasts, to predict traffic congestion, optimize routing transit routes, increase the efficiency of public transportation, and improve the overall transportation experience. By analysing real-time and historical data, these models offer solutions such as traffic rerouting, recommending efficient logistics routes, forecasting public transport demand, improve security, revolutionizing the way transportation is managed and transform urban mobility for the better.

MACHINE LEARNING MODELLING

The machine learning algorithm includes the following steps for analysis.

- 1) **Data Collection:** IoT devices collect data from various sensors, such as traffic cameras, GPS devices, motion, or light sensors. The data can be collected continuously or at regular intervals and transmitted to a central server or cloud platform for processing.
- 2) **Preprocessing:** Raw data collected from IoT devices often requires preprocessing to remove noise, handle missing values, or normalize the data. This step ensures that the data is in a suitable format for further analysis.
- 3) **Feature Extraction:** Feature extraction involves identifying and extracting relevant features from the pre-processed data. Features are specific measurements or characteristics that capture the essential information for the intended analysis or application. For example, in a smart commuting, features could include speed, Num_Reads, Hour, Zip Codes, Region, Bus Count.



- 4) **Selection and Dimensionality Reduction:** Depending on the application, it may be necessary to select a subset of features or reduce the dimensionality of the data. This step aims to eliminate irrelevant or redundant features, improving computational efficiency and reducing the risk of overfitting in machine learning models.
- 5) **Data Integration:** In some cases, data from multiple IoT devices or sources may need to be integrated to derive meaningful insights. Integration can involve combining data from various sensors, time synchronization, or merging data from different locations or devices.
- 6) **Data Analytics:** Once the relevant features have been extracted and processed, various analytics techniques can be applied to gain insights or make predictions. This can include statistical analysis, data mining, machine learning algorithms, or artificial intelligence models.
- 7) **Visualization and Reporting:** The processed data and analytics results can be visualized using charts, graphs, or dashboards to provide a clear representation of the information. Visualizations aid in understanding patterns, trends, or anomalies in the data. Additionally, reports or alerts can be generated to notify users or stakeholders of important findings or events.
- 8) **Real-Time Processing:** IoT systems often require real-time processing to enable timely decision-making or immediate actions based on the collected data. Real-time processing involves analysing data as it arrives and generating responses or triggers in near real-time.
- 9) **Feedback Loop:** The insights or actions derived from the processed data can be used to provide feedback and optimize the IoT system's performance. For example, adjusting sensor thresholds, improving predictive models, or triggering automated responses based on the analysis results.

Overall, feature extraction and data processing using IoT play a crucial role in transforming raw data collected from IoT devices into meaningful information and actionable insights for various applications such as smart homes, industrial monitoring, healthcare, or environmental monitoring and traffic management. Using Rank and percentile approach for the feature selection the following six features were found to be most relevant attributes for the data analysis and modelling.

Table 1 Rank and percentile

Sl. No.	Attribute Selection		
	Attribute Name	Rank	Percent
1	Speed	1	95.00
2	Num Reads	1	95.00
3	Hour	3	85.00
4	Zip Codes	3	85.00
5	Region	5	55.00
6	Bus Count	5	55.00

To analyse different prediction models, performance metrics such as TP Rate, FP Rate, precision and recall were used. These parameters are derived from a confusion matrix that shows the different ways in which the classification model gets confused when making predictions.

True Positive (TP) as in [11] refers to the number of predictions where the classifier correctly predicts the positive class as positive.

True Negative (TN) refers to the number of predictions where the classifier correctly predicts the negative class as negative.



False Positive (FP) refers to the number of predictions where the classifier incorrectly predicts the negative class as positive.

False Negative (FN) refers to the number of predictions where the classifier incorrectly predicts the positive class as negative.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 3 Confusion Matrix

ML PREDICTIVE MODEL ANALYSIS (LOW TRAFFIC)

Predictive machine learning models for smart transportation provide insights and data-driven solutions to solve complex urban mobility challenges. These models leverage large data sets from traffic sensors, GPS devices, and a variety of sources to predict traffic patterns, optimize routes, and improve public transportation systems. By leveraging real-time and historical data, they enable more efficient traffic management, reduce congestion, improve user experience, and promote sustainable transportation practices. Furthermore, these predictive models have the potential to play a central role in shaping future transportation policies and infrastructure developments for smarter and more accessible cities. The Weka tool is used to analyze Nine different classification machine learning algorithms. For the analysis of various predictive models, the performance measure like TP Rate, FP Rate, Precision and Recall were being used. Udaipur data set extracted from tomtom Server as in [12] was used with following features to analyze Machine Learning algorithms.

Table 2 Classifiers Performance Measure

Sl. No.	Class Label: Low Traffic	TP Rate	FP Rate	Precision	Recall
	Classifier				
1	Bayes Net	0.976	0.000	1.000	0.976
2	Naïve Bayes	0.971	0.060	0.996	0.971
3	Logistic	0.972	0.134	0.990	0.972
4	SM0	0.987	0.164	0.988	0.987
5	IBK	0.991	0.328	0.977	0.991
6	KStar	0.991	0.328	0.977	0.991
7	MultiClass Classifier	0.972	0.134	0.990	0.972
8	Random Forest	1.000	0.179	0.987	1.000
9	Random Tress	0.994	0.284	0.980	0.994

TP Rate as in [13] is used to measure the percentage of actual positives which are correctly identified by model. According to the performance measure TP rate for class label: Low Traffic it was found that the highest true positive rate was of the classifier Random Forest with value 1, followed by 0.994, 0.991 and 0.991 of Random Tree, IBK and KStar respectively whereas the lowest TP rate was found to be of the classifiers Naïve Bayes, Logistic and

Multiclass with values 0.971, 0.972 and 0.972 respectively. Overall, it can be interpreted the most appropriate classifier based on the performance measure TP rate is found to be Random Forest. The "Random Forest" classifier appears to perform exceptionally well, with a TP Rate of 1.000, indicating perfect performance in distinguishing "Low Traffic" instances.

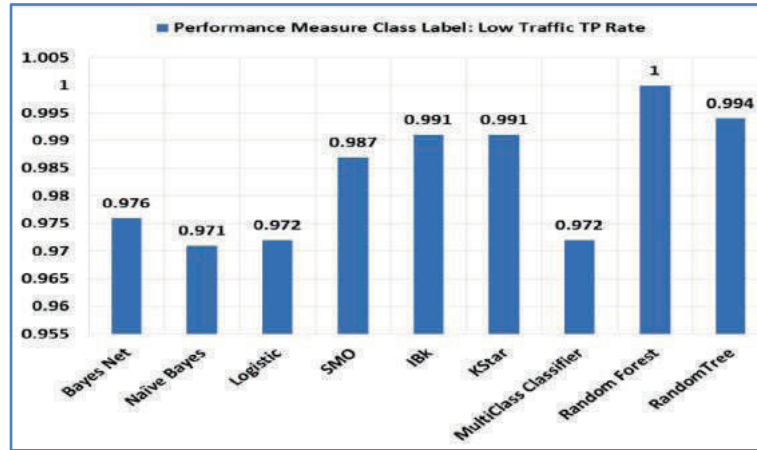


Figure 4 TP Rate for Class Label: Low Traffic

FP Rate also known as Type - I error is used to measure the percentage of actual positives which are incorrectly identified by model. Based on the performance measure FP rate for class label: Low Traffic it was found that the lowest false positive rate was of the classifier Bayes Net with value 0.000, followed by 0.06 of Naive Bayes whereas the highest FP rate was found to be of the classifiers IBK and KStar with values 0.328 each. Overall, it can be interpreted the most appropriate classifier based on the performance measure FP rate is found to be Bayes Net with lowest FP rate value.

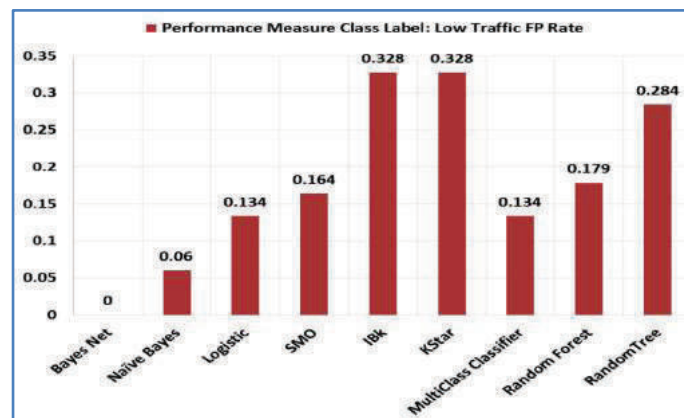


Figure 5 FP Rate for Class Label: Low Traffic

Precision as in [14] is the quality of a positive prediction made by the model. Precision refers to the number of True Positives divided by the total number of Positive predictions.

$$\text{Precision} = \frac{T_p}{T_p + F_p} \quad (1)$$

According to the performance measure precision for class label: Low Traffic it was found that the highest precision value was of the classifier Bayes Net with value 1, followed by 0.996, 0.99 and 0.99 of Naive Bayes, SMO and

Multiclass classifier respectively whereas the lowest precision value was found to be of the classifiers IBK and KStar with values 0.977 each. Overall, it can be interpreted the most appropriate classifier based on the performance measure precision is found to be Bayes Net.

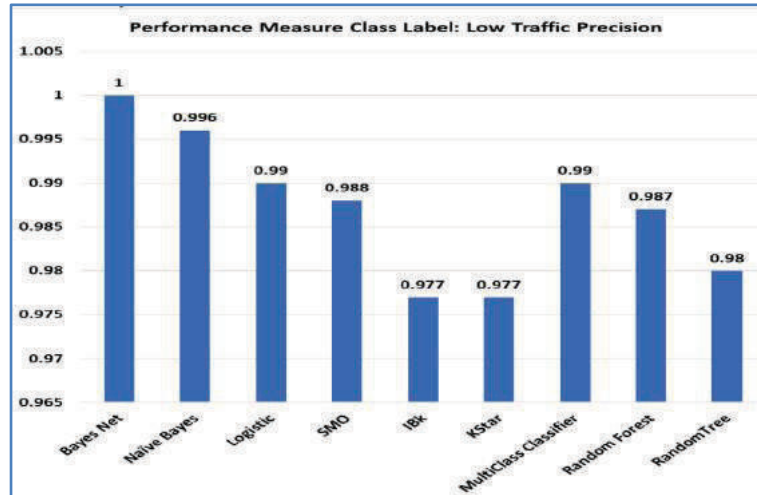


Figure 6 Precision for Class Label: Low Traffic

Recall as in[15] is the measures of how well a machine learning model can detect positive instances. It is also called as Sensitivity. Sensitivity refers to the number of true positives divided by the sum of True Positives and False Negatives. The model with high Sensitivity will have significantly fewer False Negatives.

$$\text{Sensitivity or Recall} = \frac{T_P}{T_P + F_N} \quad (2)$$

Based on the performance measure recall for class label: Low Traffic it was found that the highest recall value was of the classifier Random Forest with value 1.00, followed by 0.991 and 0.991 of IBK and KStar respectively whereas the lowest recall value was found to be of the classifiers Naive Bayes, Logistic and Multiclass with values 0.971, 0.972 and 0.972 respectively. Overall, it can be interpreted the most appropriate classifier based on the performance measure recall is found to be Random Forest.

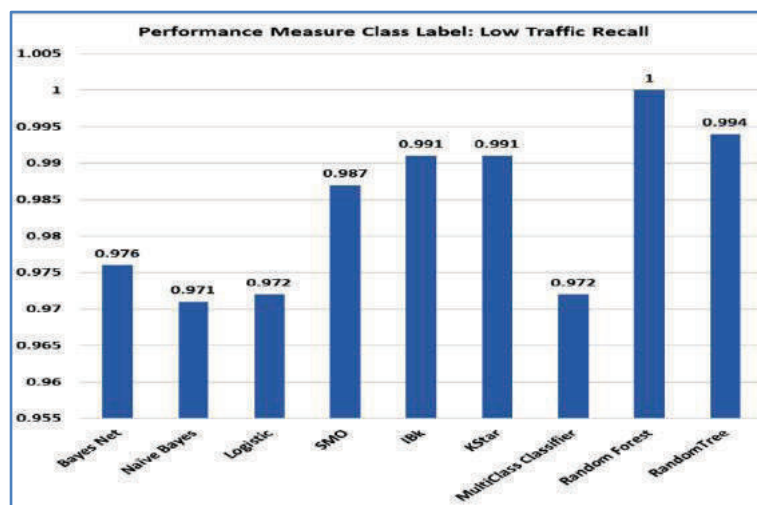


Figure 7 Recall for Class Label: Low Traffic

ML Predictive Model Analysis (High Traffic)

Sl. No.	Class Label: High Traffic				
	Classifier	TP Rate	FP Rate	Precision	Recall
1	Bayes Net	1.000	0.024	0.753	1.000
2	Naïve Bayes	0.940	0.029	0.700	0.940
3	Logistic	0.866	0.028	0.690	0.866
4	SMO	0.836	0.013	0.824	0.836
5	IBK	0.672	0.009	0.849	0.672
6	KStar	0.672	0.009	0.849	0.672
7	MultiClass Classifier	0.866	0.028	0.690	0.866
8	Random Forest	0.821	0.000	1.000	0.821
9	Random Tress	0.716	0.006	0.716	

According to the performance measure TP rate for class label: Heavy Traffic it was found that the highest true positive rate was of the classifier Bayes Net with value 1.0, followed by 0.94 , 0.866 and 0.866 of Naïve Bayes, Logistic and Multiclass respectively whereas the lowest TP rate was found to be of the classifiers IBK and KStar with values 0.672 each. Overall, it can be interpreted the most appropriate classifier based on the performance measure TP rate is Bayes Net.

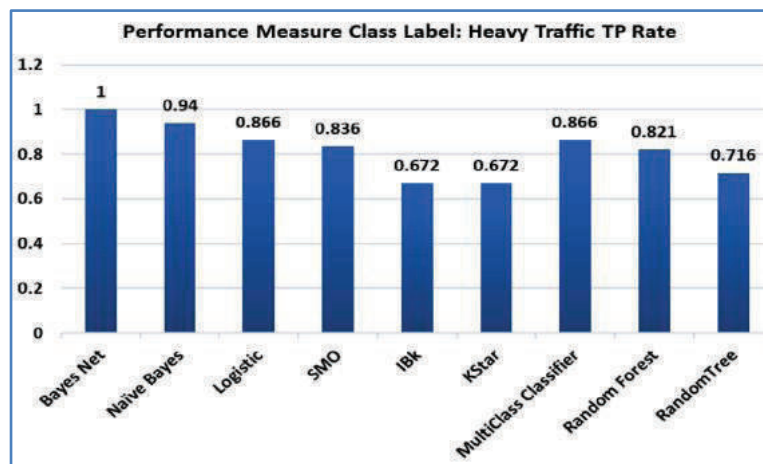


Figure 8 TP Rate for Class Label: Heavy Traffic

Based on the performance measure FP rate for class label: Heavy Traffic it was found that the lowest false positive rate was of the classifier Random Forest with value 0.00, followed by 0.006 of Random Tree whereas the highest FP rate was found to be of the classifiers Naïve Bayes, Logistic and Multiclass with values 0.029, 0.028 and 0.028 respectively. Overall, it can be interpreted the most appropriate classifier based on the performance measure FP rate is found to be Random Forest with lowest FP rate value.

According to the performance measure precision class label: Heavy Traffic it was found that the highest precision value was of the classifier Random Forest with value 1.0, followed by 0.889, 0.849 and 0.849 of Random Tree, IBK and KStar respectively whereas the lowest precision value was found to be of the classifiers Logistic, Multiclass and Naïve Bayes with values 0.69, 0.69 and 0.7 respectively. Overall, it can be interpreted the most appropriate classifier based on the performance measure precision is found to be Random Forest.

Based on the performance measure recall class label: Heavy Traffic it was found that the highest recall value was of the classifier Bayes Net with value 1.0, followed by 0.94 , 0.866 and 0.866 of Naïve Bayes, Logistic and Multiclass respectively whereas the lowest recall value was found to be of the classifiers IBK and KStar with values 0.821

each. Overall, it can be interpreted the most appropriate classifier based on the performance measure recall is found to be Bayes Net.

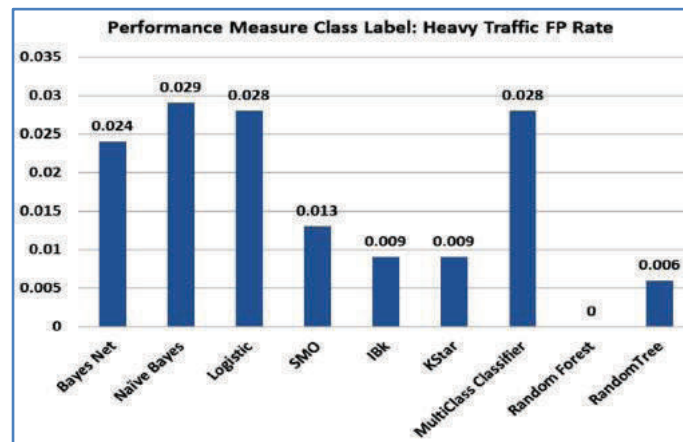


Figure 9 FP Rate for Class Label: Heavy Traffic

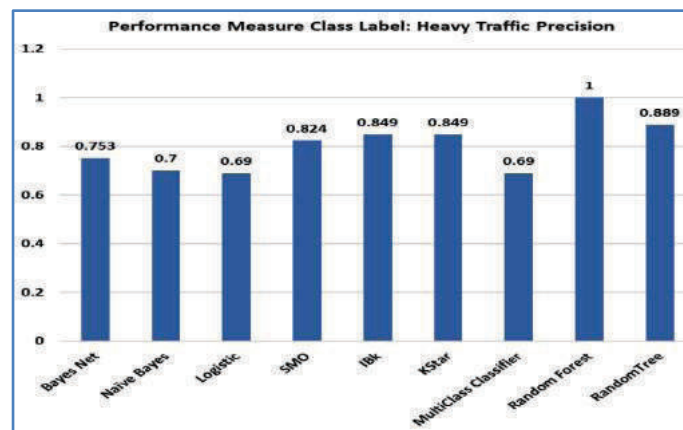


Figure 10 Precision for Class Label: Heavy Traffic

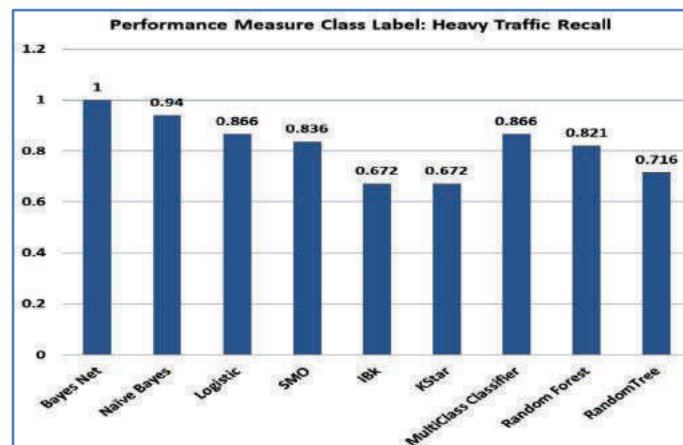


Figure 11 Recall for Class Label: Heavy Traffic



SUMMARY

The major findings related to the comparative analysis of machine learning predictive models using Weighted Sum Model (WSM) approach is shown below.

1. Random Forest is the best and most appropriate classifier for traffic congestion control and traffic flow as it is having the highest score of 0.98.
2. Bayes Net is the second most appropriate algorithm with score of 0.6530
3. The third best classifier being identified is SMO with score of 0.4871.

For predicting the Udaipur traffic flow Random Forest is the most appropriate algorithm.

Table 3 Weighted sum model

Classification Models	Score (Weighted Sum)
Random Forest	0.9800
Bayes Net	0.6530
SMO	0.4871
Random Tree	0.4110
Naïve Bayes	0.3111
KStar	0.1745
Logistic	0.1406
MultiClass Classifier	0.1406
IBK	0.0679

CONCLUSION AND FUTURE WORK

Machine learning algorithms provide powerful capabilities for a variety of tasks when provided with high-quality, validated data. Maintaining model accuracy over time requires continuous model monitoring and adjustment. This approach includes real-time data analysis, model retraining as data evolves, and automatic deployment of updated models to ensure sustained performance and relevancy. A K-fold cross-validation approach is required. A fold is a set of records in a dataset, and k is the number of folds that affects the performance and reliability of the model. In general, higher values of k increase variance and computational cost, but decrease bias. Furthermore, Data Analysis with class label of different traffic densities apart from Low Traffic and high traffic will increase the accuracy and reliability of the model.

REFERENCES

1. M. King, B. Zhu, and S. Tang, "Optimal path planning," *Mobile Robots*, vol. 8, no. 2, pp. 520-531, March 2001.
2. Deak, G., & Walravens, N. (2019). The future of urban mobility: Towards integrated smart mobility solutions. *Transport Reviews*, 39(2), 157-181.
3. Erlmann, S., & Dantzig, S. V. (2018). Exploring the potential of shared mobility services to alleviate urban traffic congestion: A literature review. *Transport Reviews*, 38(6), 769-793.
4. Fintikakis, N., & Bourka, A. (2018). Commuting patterns and smart cities: A comparative study of Stockholm and London. *TeMA Journal of Land Use, Mobility and Environment*, 11(3), 239-254.
5. Gkiotsalitis, K., & Cats, O. (2018). Assessing the impact of autonomous vehicles on our cities. *Transportation Research Part A: Policy and Practice*, 118, 407-417.
6. Gohar, M., Sagheer, A., Javaid, N., Anpalagan, A., & Khan, F. (2019). Fog computing-enabled intelligent transportation system: Architecture, opportunities, and challenges. *IEEE Access*, 7, 29037-29057.
7. Hall, R. E., & Pfeiffer, U. (2016). The impact of smart cities on travel behavior: An exploratory study. *Transportation Research Part A: Policy and Practice*, 88, 96-114.



8. Jung, C., & Couclelis, H. (2017). Smart cities and the politics of urban data. *Journal of Planning Education and Research*, 37(4), 457-466.
9. Kaur, A., & Kaur, P. (2017). Internet of Things for transportation: A review. *IEEE Sensors Journal*, 17(14), 4371-4378.
10. Khan, A., Salah, K., & Zeadally, S. (2019). Fog computing for intelligent transportation systems: Opportunities and challenges. *Computer Networks*, 160, 96-112.
11. <https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826>
12. <https://www.tomtom.com/products/traffic-stats/>
13. https://link.springer.com/referenceworkentry/10.1007/978-1-4419-9863-7_255#:~:text=Definition,positives%20which%20are%20correctly%20identified.
14. [https://c3.ai/glossary/machine-learning/precision/#:~:text=Precision%20is%20one%20indicator%20of,the%20number%20of%20false%20positives\).](https://c3.ai/glossary/machine-learning/precision/#:~:text=Precision%20is%20one%20indicator%20of,the%20number%20of%20false%20positives).)
15. <https://www.shiksha.com/online-courses/articles/sensitivity-vs-specificity/#:~:text=Sensitivity%20measures%20how%20well%20a,have%20significantly%20fewer%20False%20Negativ>
es.



Springer



Representation Certificate

This certificate is awarded to

Dangwani Avinash

have successfully presented the paper entitled

Use of AI in Cloud based Certificate Authentication for Travel Concession.

at the

4th International Conference on

“Mobile Computing and Sustainable Informatics”

(ICMCSI 2023)

organized by

Tribhuvan University, Nepal on 11th & 12th January, 2023


Session Chair


Organizing Secretary
Dr. S. Smys


Conference Chair
Prof. Dr. Subarna Shakya



Proceedings

of the

4th International Conference on Mobile Computing and Sustainable Informatics ICMCSI 2023

11-12, January 2023 | Lalitpur, Nepal



Springer

organized by



Tribhuvan University
Nepal



14	The Application of Mobile Phones to Enable Traffic Flow Optimisation <i>T Shilowa, J P van Deventer, M J Hattingh</i>
15	Authentication in Cloud Computing: Open Problems <i>Zarif Khudoykulov, Abdukodir Karimov, Ravshan Abdurakhmanov, Mirkomil Mirzabekov</i>
16	A Hypothesis on Cloud Sourcing Sharing Users' Mobile Devices through Virtualization <i>Nazmus Sakib, Al Hasib Mahamud</i>
17	AN ANALYTICAL ASSESSMENT OF MACHINE LEARNING ALGORITHMS FOR PREDICTING CAMPUS PLACEMENTS <i>S Bala Dhanalakshmi, Raksheka Rajakumar, Swetha Shankar, R Sowndharya Rani, N Deepa, C Subha Priyadharshini</i>
18	A Systematic Literature Review on Symmetric and Asymmetric Encryption Comparison Key Size <i>Mohammed Althamir, Abdullah Alabdulhay, Muhammad M Yasin</i>
19	Building Hindi Text Dataset on Stock Market Tweets and Sentiment Analysis Using NLP <i>Anushka Choudhary, Mohit Gupta, Lavanya S K</i>
20	Fine Tuning of RoBERTa for Document Classification of Arxiv Dataset <i>Kshetrphal Bohara, Aman Shakya, Bishal Debb Pande</i>
21	Comparative Analysis of Using Event Sourcing Approach in Web Application Based on the LAMP stack <i>Marian Slabinoha, Stepan Melnychuk, Vitalia Kropyvnytska, Bohdan Pashkovskyi</i>
22	Profitability Improvement for CV. XYZ with Improvement of Work Posture and Workflow Using REBA and Modelling Simulation <i>Louis Valentino, Lina Gozali, Frans Jusuf Daywin, Ariawan Gunadi</i>
23	Demand Forecasting Using Time Series and ANN with Inventory Control to Reduce Bullwhip Effect on Home Appliances Electronics Distributors <i>Stiven Tjen, Lina Gozali, Helena Juliana Kristina, Ariawan Gunadi, Agustinus Purna Irawan</i>
24	Memory Malware Identification via Machine Learning <i>Maysa Khalil, Qasem Abu Al-Haija</i>
25	Basketball Shot Conversion Prediction using various ML techniques and its analysis <i>Sanyam Raina, Shreedhar Bhatt, Vaidehi Shah, Heem Amin, Vinay Khilwani, Dr. Samir Patel</i>
26	Progressive Web App Implementation in Omah Wayang Klaten Website <i>Budi Susanto, Gloria Virginia, Umi Proboyekti, Jeysy Carmila Dewi Ester</i>
27	Use of AI in cloud based certificate authentication for travel concession. <i>Dangwani Avinash, Jetawat Ashok Kumar, Rawat Chandansingh</i>
28	Media Player Controller using Hand Gestures <i>Vijay Mane, Harshal Baru, Abhishek Kashid, Prasanna Kshirsagar, Aniket Kulkarni, Prathamesh Londe</i>

Use of AI in cloud based certificate authentication for travel concession.

* Dangwani Avinash¹, Jetawat Ashok kumar², and Rawat Chandansingh³

¹ Computer Department Pacific Hills, Pratap Nagar Extension, Airport Road,
Debari, Udaipur—313024, Rajasthan, India
^{1*}avin861@gmail.com

² Computer Department Pacific Hills, Pratap Nagar Extension, Airport Road,
Debari, Udaipur—313024, Rajasthan, India
²drashokjain61@gmail.com

³ Vivekanand Education society institute of technology collectors colony,
Mumbai– 400074, Maharashtra, India
³chandansingh.rawat@ves.ac.in

Abstract. Certificate Authentication is a big challenging task in socio economic country like India. 28 states and 8 union territories with diversified cultures and languages makes it big voluminous task. There is urgent need to automate this authentication task. This paper proposes the cloud based certificate authentication. Google cloud services are used to automate authentication process. Vision API and flask framework is explored which allows developers to easily integrate vision detection features within applications including Image labeling, Face and landmark detection, Optical character recognition, Tagging of explicit contents. The proposed arrangement make use Cloud Vision API optical character recognition to infer presence of required fields in scanned pdf. Recognized fields will be communicated in the output report.

Keywords: Caste Authentication, Google cloud vision, Open computer vision, Optical character recognition, APISetu, Google Cloud Vision, Application program interface, Cloud bucket.

1 Literature review

H. Gaikwad, N. D'Souza, R. Gupta and A. K. Tripathy(2021) found millions of students every year go through lengthy and cumbersome process of document verification for their higher studies. This results in significant overhead as documents are transferred between institutions for verification. There is a need for an automated credential verification system which can reduce the time required for the document verification process. They have used Blockchain Technology that can be used to reduce overhead and reduce the time taken for document verification

from days to mere seconds.

Jignasha Dalal Meenaland Chaturvedi Himani Gandre and Sanjana Thombare(2020) proposed solution of biometric to access all the previous degree certificates of a students using blockchain. The students will submit the hash of their biometric and a unique phrase. This hash will be stored on the blockchain. The degree certificate for a student will be issued by the college authorities. They will upload the hash of digitally signed certificate on the blockchain. They proposed to link documents with person's identity without involving a third party.

D. Vaithiyanathan and M. Muniraj(2019) has worked on "Cloud based Text extraction using Google Cloud Vision for Visually Impaired applications". They have designed assistive device called smart reader that is capable of capturing an image from a camera and extract the text from the captured image. Text is converted to speech as voice based output to assist the visually impaired people.

2 Introduction

This paper will emphasize on Google services and it's API for developing cloud based certificate authentication system. Manual authentication of certificates for senior citizens, students & citizens belonging to certain gender and community becomes sometimes confusing and time consuming. The inspector who validates documents for giving travel concession needs substantial amount of proof and supporting documents to authenticate certificates. Citizens specially senior citizens sometimes have to face stressful situation due to delayed and cumbersome authentication process. Recently Maharashtra state in India has given complete state transport travel free to senior citizens above 75 years. Our proposed solution consists of a cloud web app which will have a front-end to get applicant details, certificate submission and requesting authentication, along with a Google cloud backend which will have three modules: An OCR API module to extract applicant details from certificates, An compare module to compare extracted details with submitted details and a report module to send and display authenticated data stored in the cloud storage. Google cloud is acting as third party trusted centralised authority which provides software as service and centralized storage for authentication of certificates against distributed systems which uses blockchain technology to authenticate academic certificates[1].

With the advancement in the field of computer vision Artificial intelligence it is possible to extract text using OCR[2]. Pattern recognition technique in optical character recognition provides accurate results in extracting text from the various document formats such as JPG & PDF. Automation of authentication process reduces the administrative overhead by minimizing the use of paper and curtailing

the verification process. It provides a Real-Time verification module enabling agencies to verify data directly from issuers after obtaining user consent[3-10].

The proposed certificate authentication model for travel concession is Organized into following sections, Authentication System, System Design, Experimental Results, Summary, Conclusion and Future Work.

3 Authentication System

The main aim is to develop quick and easy to use cloud based certificate authentication system which uses centralize cloud SQL database to compare OCR API extracted features with the stored features collected from user with the help of front end designed using HTML and JavaScript. Google cloud vision OCR Functions are used to recognize the text in pdf document and are used to convert pdf image into accessible electronic text[10-17].

To overcome the drawback of fraudulent intentions of applicant and forged documents two level Authentication system Local Level and Government Server level is proposed as in Fig 1.

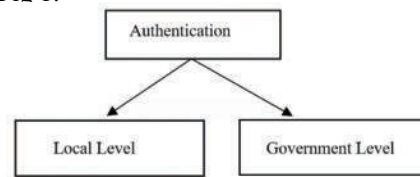


Fig. 1. Authentication types

3.1 Local Level Authentication

For Local level authentication OCR extracted text is compared with the stored cloud SQL database. Cloud SQL is Fully managed relational database service for MySQL, PostgreSQL, and SQL Server with rich extension collections, configuration flags, and developer ecosystems. Cloud SQL has many advantages like reduced maintenance cost, fully reliable and secured 24/7 service, server instances can be scaled effortlessly when demand increases.

3.2 Government Level Authentication

After successful comparison of all required attributes second level is proposed which has some government restrictions like GST Number. For Government server level APISetu or uidia.gov.in site can be used. APISetu provides single platform access to information from multiple sources. It can be used for a variety of use cases such Know Your Client (KYC) and other authentication services.

4 System Design

The cloud web application program is designed to authenticate local fields like Certificate type, name of applicant, state, Aadhaar card, Date of Birth, Age & School name. Front end is designed in HTML and backend code is written in python and flask framework.

Google Cloud vision API technology identify the content of an image with the help of powerful machine learning models. REST API in Google Cloud Vision API enables developers to understand the content of an image by encapsulating powerful machine learning models. It has powerful pre-trained machine learning models. It offers powerful image analysis, insight from your images, detect and classify multiple objects including the location of each object within the image. & detect required content. It can read printed and handwritten text, and build valuable metadata into your image Cloud vision API can be used with Auto ML Vision to automate the training of your own custom machine learning models. These models can be optimized for accuracy, latency and size.

Google Cloud vision AI is designed to understand text with pre trained vision API models. The research motivation and the will to explore more about Cloud vision API is created after knowing the applications of the big companies like New York Times and box. New York Times is using Google cloud to preserve the visual history by finding out many untold stories in millions of archived photo's & Box company is using image recognition and OCR API of Google Cloud vision for content management. Cloud vision technology has enabled New York Times to unveil more than a century of global events that have shaped our modern world. The New York Times built a processing pipeline that stores and processes the photos and will use cloud technology to process and recognize text, handwriting and other details that can be found in the images. Box company extracted printed words from the scanned image and then returned labels and recognized characters in JSON responses. Fig. 2 shows the System Design.

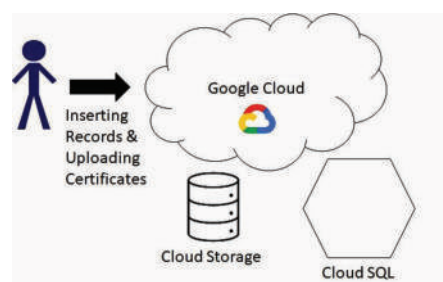


Fig. 2. Uploading credentials

Since certificate authentication is most widely used for students bus fare concession, Senior citizen bus fare in india therefore sample student certificates, Aadhaar card

and senior citizen Aadhaar card were selected for developed cloud based software testing. Another area where certificate authentication is used for students belonging to Schedule caste, Schedule Tribe and Other backward class category in India.

Front end is designed using HTML forms to get required details / records from the student. All these records are maintained in cloud SQL tables. Cloud SQL is fully managed relational database service for MySQL, PostgreSQL, and SQL Server with rich extension collections, configuration flags and developer ecosystems. It provides reliable, secure and scalable solution for cloud data base which ensure that operations should run 24 * 7 for 365 days without any disruptions. Cloud SQL automates all your backups, replication, encryption patches, and capacity increases—while ensuring greater than 99.95% availability, anywhere in the world.

Three cloud SQL Tables, Testify table, Reference Table and Flag Table were created using SQL CREATE TABLE Query command and used for HTML form backend storage.

4.1 Testify Table

Table 1. Testify table

Name	Email id	State	Aadhar Card Number	Date of Birth	Age	School Name

Testify_table stores personal details of applicant like name, Email id, state, Unique identification number (Aadhaar card), Date of Birth, Age, School name.

Unique id Aadhaar Card Number is the key which is used to search particular record in Testify_table. To make system more easy to use and user friendly, another search options are also included. Different ways SQL data can be searched from Testify_table are, Search by Aadhaar Card Number , Search by Name , Search by Email id and Search by School Name.

4.2 Reference Table

Table 2. Reference table

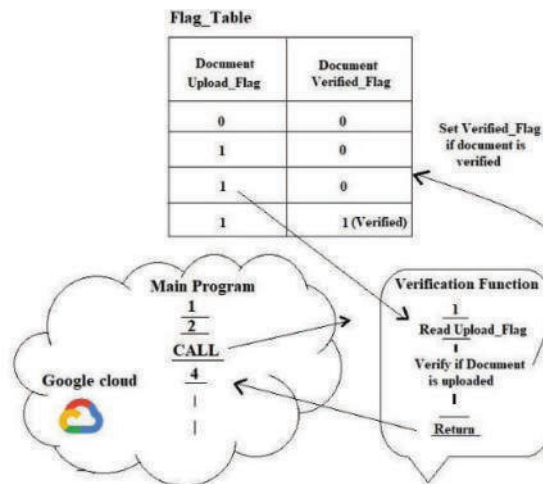
Aadhar Card Number	File_Name	Certificate type

Reference table was created with the aim to provide reference for caste certificate file storage in Google Cloud storage buckets. Reference table helps designed code

to understand which certificates are uploaded in backend. The Null entry in Certificate type column indicates that file is not uploaded in cloud bucket. Code can be designed to send email message “Certificate type not uploaded” to applicant.

4.3 Flag Table

Table. 3 Flag Table



Flag table was created to mark different conditions of file upload to know whether required caste certificates for authentication are loaded or not. Flags in flag table are basically sequence of pre-defined binary bits which holds true / logic 1 when required certificates are uploaded and false / logic 0 when required certificates are not available in cloud storage bucket.

4.4 Process Model

Process flow is depicted in Fig.3 When user types URL on address bar of Browser, it get's designed HTML index Form for inputting data like Name, Email id, state, Aadhaar card Number, Date of Birth, Age, School name. All these details are loaded in Testify_table (Table 1). Upon pressing next button in HTML index Form it will display another form to upload required scanned pdf files like Aadhaar card & Bus ticket coession application form. All these pdf files will be loaded in Google cloud bucket which is created in Google cloud storage. Once the files will be loaded in Google cloud bucket then reference table (Table 2) will be updated with Aadhaar card Number, File name prefixed by Aadhaar card Number & Certificate type. To know whether applicant has uploaded required document or not flag_table is updated with logic 1 for particular document. Logic is shown in (Table 3). Main Program in Google cloud will read Upload_Flag for particular certificate from Flag_Table and will invoke OCR API code for reading Certificates.

OCR API has `async_batch_annotate_files()` annotation function which detect text and image for a batch of generic files, such as PDF documents at once. This function can be invoked in python language by importing vision module from google cloud which will generate unstructured data which is stored in Blob list. Blob is object storage solution for the cloud. It is optimized for storing massive amounts of unstructured data such as text or image binary data. Blob data is converted into JSON string data, which is finally converted into text format to search required text field is present or not.

Training code is required to train designed software about the format used in the uploaded document. Execute Verify function to find required text field is available as per trained model. If required text (name, state, city, date of birth, age etc) is found then document verified_flag will be updated to logic 1 and output report is generated where the verified column holds identified / not identified condition. The fields identified will be notified in “Fields verified” column.

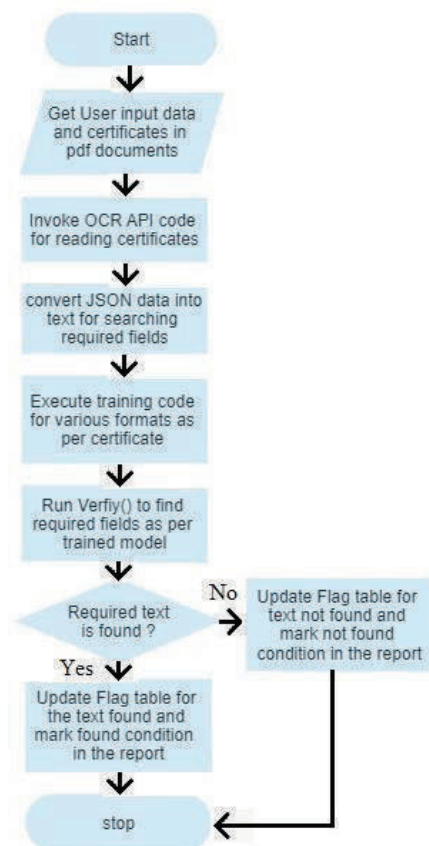


Fig. 3. Design Flow chart

5 Experimental Results

5.1 Front end Dashboard

The screenshot shows a web application titled "DOCUMENT VERIFICATION PORTAL". It features a navigation bar with "Dashboard", "Search", and "Reports" options. Below the navigation bar, there are three status messages: "The last date to fill the form is 31/12/2022", "The last date to fill the form is 31/12/2022", and "The last date to fill the form is 31/12/2022". The main content area is divided into two sections: "Applicant Details" and "Document Upload". The "Applicant Details" section contains a form with the following fields: "Applicant name", "Applicant email", "Applicant state", "Applicant aadhar", "Applicant Date of Birth", "Applicant age", and "Applicant School Name". A "NEXT" button is located at the bottom of the form.

Fig. 4. Front end

Snap shot of front end is given in fig 4. The objective of front end is pleasant and convenient user interface. Front end has three menu options, Dashboard, Search and Reports. Dashboard has designed HTML form to take applicant details and tab for Document Upload. All entries are recieved from user are loaded in SQL testify_table of SQL instance data base as shown in Table 4.

Table. 4 Flag Table

Name	Email_id	State	Aadhaar_Card_Number	Date_of_Birth	Age	School_Name
Romil	Romil_raj@gmail.com	Goa	790195155740	26/10/2004	18	BITS Pilani
Riya	Riya2005@gmail.com	Kerala	527295155654	01/11/2005	17	St. Joseph

After Applicant details are entered, user is prompted to upload Aadhaar card and concession form details as shown in fig. 5. Due to functional limitations we have restricted file upload to pdf only. With modification other files format upload can also be allowed. After uploading, documents are stored in Google cloud bucket and designed code will update Reference table with the details as shown in Table 5.

Fig. 5. Document upload

Table. 5 Flag Table

Aadhaar_card_Number	File_name	Certificate_type
790195155740	790195155740_Aadhaarcad.pdf	Aadhaarcad
527295155654	527295155654_Concessionform	Concessionform

5.2 Front end Search

Effective search always improve productivity, enhances decision making and makes management easier. Fig. 6. Shows search options.

Fig. 6. Search options

Different search options enable administrator to search records by Aadhaar card Number (Unique identification number), Search by Name, Search by Email id and Search by School Name.

5.3 Front end Reports

Reports gives you real time information at your finger tips and improves speed in decision making. Efficient reports always save allot of time and makes management easier. If required fields in certificate are not authenticated or wrong certificates are uploaded then software is generating output shown in fig. 7. Output for correct certificate upload is shown in fig. 8.

Certificate Type	Verified	Fields Verified
Aadhaar Card	X	Name , Aadhaar Card Number State
Student Concession Application Form	X	Name, State, Age , School name

Fig. 7. Report with wrong certificate upload

Certificate Type	Verified	Fields Verified
Aadhaar Card	✓	Name , Aadhaar Card Number State
Student Concession Application Form	✓	Name, State, Age , School name

Fig. 8. Report with valid certificate upload

6 Summary

To overcome misery of slow and manual authentication work and to keep pace with emerging trends and innovation this project was selected. This project has explored the world of artificial intelligence to uplift society by revolutionizing smart work culture in society.

There was big question in front of us whether to develop independent PC Application or to develop cloud based software. Looking at several merits like scalability, availability, advance security, data loss prevention and collaborative work environment the cloud project was selected. There are four major cloud service providers in the world, Microsoft AZURE, Amazon Web Services(AWS), IBM cloud services and google cloud platform (GCP). Google cloud platform was selected due to its economical charges and localised services.

Artificial intelligence computer vision technology has opened doors for various innovations, image to text conversion using OCR API is one of them. various functions are developed in cloud technology to convert single or group of images into text. Various formats like PDF and JPEG are converted into JSON formats and finally into text formats for editing, this text can be used to compare with required fields for getting desired results. Lakhs of images can be converted and compared in very less time using cloud technology, which is practically not possible for human being.

7 Conclusion and Future Work

We have investigated use of Google cloud vision AI - API for certificate authentication. This cloud automation will simplify authentication process of concession issuing authority up to great extent and speed up process of issuing concession in smart cities. The main practical limitation of this project is second level authentication, which is not possible without the help of Government servers. With the help of Govt API available on site <https://www.uidai.gov.in/914-developer-section.html> we can opt for second level authentication also in future. Face detect feature of Google cloud vision AI – API can be explored for better results in future. Allot of work is in progress in cloud vision API and cloud technologies. Government of India vision for NRC (National register for citizens) will speed up cloud authentication and automation.

References

- [1] H. Gaikwad, N. D'Souza, R. Gupta and A. K. Tripathy, "A Blockchain-Based Verification System for Academic Certificates," 2021 International Conference on System, Computation, Automation and Networking (ICSCAN), 2021, pp. 1-6, doi: 10.1109/ICSCAN53069.2021.9526377.
- [2] D. Vaithyanathan and M. Muniraj, "Cloud based Text extraction using Google Cloud Vision for Visually Impaired applications," 2019 11th International Conference on Advanced Computing (ICoAC), 2019, pp. 90-96, doi: 10.1109/ICoAC48765.2019.246822.
- [3] Q. Liu Q. Guan X. Yang H. Zhu G. Green and S. Yin "Education-Industry Cooperative System Based on Blockchain" 1st IEEE International Conference on Hot Information-Centric Networking (HotICN) pp. 207-211 15–17 August 2018.
- [4] Jignasha Dalal Meenaland Chaturvedi Himani Gandre and Sanjana Thombare "Verification of Identity and Educational Certificates of Students Using Biometric and Blockchain (April 8 2020)" Proceedings of the 3rd International Conference on Advances in Science & Technology (ICAST) 2020.
- [5] Yasuhisa Fujii "Optical Character Recognition Research at Google" IEEE 7th Global Conference on Consumer Electronics (GCCE) December 2018.

- [6] Heba Saleous Anza Shaikh Ragini Gupta and Assim Sagahyroon "Read2Me: A cloud-based reading aid for the visually impaired" International Conference on Industrial Informatics and Computer Systems (CIICS) May 2016.
- [7] Digilocker facility provided by Government of india
<https://www.livemint.com/money/personal-finance/all-you-need-to-know-about-digilocker-and-how-to-use-it-11612943898102.html>
- [8] Blockchain vs cloud comparison
<https://www.upgrad.com/blog/blockchain-vs-cloud-computing/>
- [9] Cloud computing vs distribute computing
<https://www.projectpro.io/article/cloud-computing-vs-distributed-computing/94>
- [10] Use of Python for fetching data from database.
<https://towardsdatascience.com/pull-and-analyze-financial-data-using-a-simple-python-package-83e47759c4a7>
- [11] Atlas platform for paperless KYC.
<https://documenter.getpostman.com/view/12409759/TVCZaWzp#intro>
- [12] Postman use with google cloud platform
<https://pnatraj.medium.com/google-cloud-api-with-postman-f4cf070e665f>
- [13] Unique Identification Authority of India
<https://www.uidai.gov.in/ecosystem/authentication-devices-documents/developer-section/916-developer-section/data-and-downloads-section.html>
- [14] Government of India Online API Library.
<https://apisetu.gov.in/api/directory>
- [15] JSON Formatter site.
<https://chrome.google.com/webstore/detail/json-formatter/bcjindcccaagfpapjjmafapmmgkkhgoa?hl=en>
- [16] Google cloud vision API
<https://stackshare.io/stackups/google-cloud-vision-api-vs-opencv>
- [17] Maharashtra state transport concession form.
<https://msrtcblog.blogspot.com/2017/10/download-msrtc-student-concession-form.html>.

12th INTERNATIONAL CONFERENCE

SUSTAINABLE GLOBAL TRENDS
: PLANET, PEOPLE AND PROFIT



PACIFIC INSTITUTE OF MANAGEMENT

PACIFIC ACADEMY OF HIGHER EDUCATION AND RESEARCH UNIVERSITY, UDAIPUR

&



PACIFIC BUSINESS SCHOOL

RAJASTHAN TECHNICAL UNIVERSITY, KOTA

This is to certify that

Mr. Avinash Dangwani

of PAHER University, Udaipur (Research Scholar)

has presented his/her paper entitled Data Analytics Sales Prediction Model

during the two-day International Conference jointly organized by Pacific Institute of Management & Pacific Business School, Udaipur in online mode. The Organizers wish his/her success in future endeavors.

Prof Mahima Birla
Conference Chair

Prof Dipin Mathur
Director Conference

Dr Shivoham Singh
Organizing Secretary

Prof. Mahima Birla (Editor in Chief)	Dr. Khushbu Agarwal (Editor)	Dr. Asha Galundia (Circulation Manager)	Editorial Team
---	---------------------------------	--	-----------------------

August 2021

Name	:	Index
Download	:	

Name	:	The Effect of Product Photograph and Information on Digital Apparel Marketing
Author	:	Ahmet Özbek, Cansu Tor-Kadıoğlu, Ahasanul Haque

Already have a manuscript?

Use our Manuscript Matcher to find the best relevant journals!

Find a Match

Refine Your Search Results

0974-438X

Sort By: Title (A-Z)

Search

Search Results

Found 1 results (Page 1)

Share These Results

Exact Match Found

PACIFIC BUSINESS REVIEW INTERNATIONAL

Publisher:

PACIFIC INST MANAGEMENT . PACIFIC HILLS. PRATAP NAGAR EXTENSION. AIR PORT RD. UDAIPUR. RAJASTHAN. INDIA. 313 001

ISSN / eISSN:

0974-438X

Web of Science Core Collection:

Emerging Sources Citation Index

[View profile page](#)

* Requires free login.

[Share This Journal](#)

Data Analytics Sales Prediction Model

Avinash Dangwani

PhD Scholar
Department of Engineering (Computers)
Pacific University
Airport Road, Debari, Udaipur (Rajasthan)

Dr. Chandansingh Rawat

Associate Professor
Dept of Electronics & Telecommunication
Engineering VESIT HAMC Collectors Colony,
Chembur Mumbai

Abstract

In every New financial year Company propose Advertisement Budget to improve their sales. Estimation of Advertisement Budget is not easy task as it involves financial parameters. Managers are always interested to know prediction model for sales which is function of Advertisement expenses.

This paper will develop Sales prediction model using simple linear regression. The model will be built using the training dataset to estimate the regression parameters. The method of Ordinary Least Squares (OLS) will be used to estimate the regression parameters using python. Regression model will be validated to ensure goodness of fit before it can be used for practical application. The single variable regression is the limitation of this model. In future multiple variables can be calculated using multiple linear regression model using python.

Keywords:

Simple linear regression, Ordinary Least Square (OLS), Training & validation data, Sum square regression (SST), Total sum of squares (SSR).

Introduction

This paper will develop sales prediction model using Simple Linear regression. sales prediction has two main methods (1) Qualitative method, (2) Quantitative method [3]. Some of the Qualitative methods are Expert's Opinion Method, Sales Force Composite Method, Survey of Buyer's Expectations, Historical Analogy Method, Jury of Executive Opinions & Leading Indicators Method.

Some of the Quantitative methods are Test Marketing, Time Series Analysis, Moving Average Method, Exponential Smoothing Method, Regression Analysis & Econometric Models.

This paper will explore sales prediction using regression analysis due to its lower time complexity as compared to some of the other algorithm. Furthermore, these models can be trained easily and efficiently even on systems with relatively low computational power when compared to other complex algorithms. Building a regression model is an iterative process and several iterations may be required before finalizing the appropriate model [2]. Regression model is Organized in following sections.

ØSection – I: Simple Linear regression

ØSection – II: Ordinary least square(OLS) Method.

ØSection – III: Results& Model Diagnostics.

ØSection – IV: Conclusion

Simple Linear Regression

Simple linear regression (SLR) is a statistical technique which uses the existence of an association relationship between a dependent variable (outcome variable) and an independent variable(predictor/feature variable).

The functional form of SLR is as follows

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \quad (1)$$

Where

Y_i = Value of the i th observation of the dependent variable

X_i = Independent variable of i th observation

ε_i = random error (residuals) in predicting the value of Y_i

β_0 & β_1 = regression parameters

Ordinary least square (OLS) Method

Equation (1) can be re written as

$$\varepsilon_i = Y_i - \beta_0 - \beta_1 X_i \quad (2)$$

The regression parameters β_0 & β_1 are estimated by minimizing the sum of squared errors(SSE)

$$SSE = \sum_{i=1}^n \varepsilon_i^2 = \sum_{i=1}^n (Y_i - \beta_0 - \beta_1 X_i)^2 \quad (3)$$

The estimated values of regression parameters are given by taking partial derivative of SSE with respect to β_0 & β_1 and solving resulting equation for the regression parameters. The estimated parameters are given by

$$\hat{\beta}_1 = \sum_{i=1}^n \frac{(x_i - \bar{X})(y_i - \bar{Y})}{(x_i - \bar{X})^2} \quad (4)$$

$$\hat{\beta}_0 = \bar{Y} - \beta_1 \bar{X} \quad (5)$$

Where $\hat{\beta}_1$ & $\hat{\beta}_0$ are estimated values of the regression parameters β_1 & β_0 and \bar{X} , \bar{Y} are mean values of X & Y .

A. Data Source

Sample Data is taken from Advertising Ratios & Budgets provided in annual report by Schonfeld & Associates, Inc [6]. which covers over 2,400 companies and 320 industries with information on fiscal 2018 and 2019 advertising& revenue spending.

For OLS Analysis total 52 sample companies data is taken from 12 different industries.

Table - 1 shows the sample percentage revenue growth & percentage advertisement growth for Electromedical & Electrotherapeutic Appartus.

Growth is taken from 2018 to 2019		Adv Grw & Rev Grw are in %	
<u>Sr.No</u>	Company	Ad Grw	Rev Grw
1	Adm Tronics Unlimited, Inc.	-3.88	-19.58
2	Axogen, Inc.	-85.78	27.13
3	Biolife Solutions Inc	43.33	38.64
4	Cutera Inc	0	11.67
5	Digirad Corp	0	9.6
6	Edap Tms Sa	-2.7	8.51
7	Electromed, Inc.	21.52	10.57
8	Escalon Medical Corp	22.22	-15.57
9	Fonar Corp	-11.37	6.96
10	Iridex Corp	-40	1.99
11	Masimo Corp	-21.79	9.27
12	Medifirst Solutions, Inc.	-92.13	-2.73

Table – 1:Data Source[6]: June 2020 Sample data of Advertising Ratios & Budgets from Schonfeld-Associates-Inc-v417 of Market Research.com

We will develop an simple regression model to understand and predictpercentage sales revenue growth on the percentage advertisement growth.

B. Creating Feature Set(X) and Outcome Variable(Y) Using Python

The OLS model takes two parameters Y and X.In our example percentage advertisement growth will be X and percentage sales revenue growth will be Y.We will split data set into two sets, training & validation set. Training set will be used to train algorithm to predict output. Validation set will be used to test accuracy & efficiency.

C. Python for Building Regression Model

Python language is used as tool for building regression model for sales prediction. The statsmodel library is used in

Python for building statistical models. OLS(Ordinary least square) API available in statsmodel.api is used for estimation of parameters for simple linear rgression model.

D. Splitting the Dataset into Training and Validation Set

The data is divided into two subsets training data set and validation data set. The proportion of training dataset is usually between 70% and 80% of the data and the remaining data is used for validation data. We have taken `train_size = 0.8` which implies that 80% of the data will be used for training the model and remaining 20% will be used for validating the model. The records that are selected for training and test set are randomly sampled using python functions which returns four variables as shown below.

`train_X` = feature values of the training set

`train_Y` = response values of the training set

`test_X` = feature values of the test set

test_Y = response values of the test set

E. Finding estimated parameters

Regression parameters $\hat{\beta}_1$ & $\hat{\beta}_0$ are estimated from equations (4) & (5) using Python functions as tool.

F. Fitting the Model

Linear regression calculates an equation that minimizes the distance between the fitted line and all of the observed data points. Technically, ordinary least squares (OLS) regression minimizes the sum of the squared residuals. In general, a model fits the data well if the differences between the observed values and the model's predicted values are small and unbiased.

G. Co-efficient of Determination (R-Squared / R^2)

R-squared is a statistical measure of how close the data are to the fitted regression line. It is defined as

$$R^2 = 1 - \frac{\text{sum squared regression (SSR)}}{\text{total sum of squares (SST)}} \quad (6)$$

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \quad (7)$$

SSR = The *sum squared regression (SSR)* is the sum of the square residuals $(y_i - \hat{y}_i)^2$. Residual is the difference between observed value y_i & estimated value \hat{y}_i as shown below in Fig - 1.

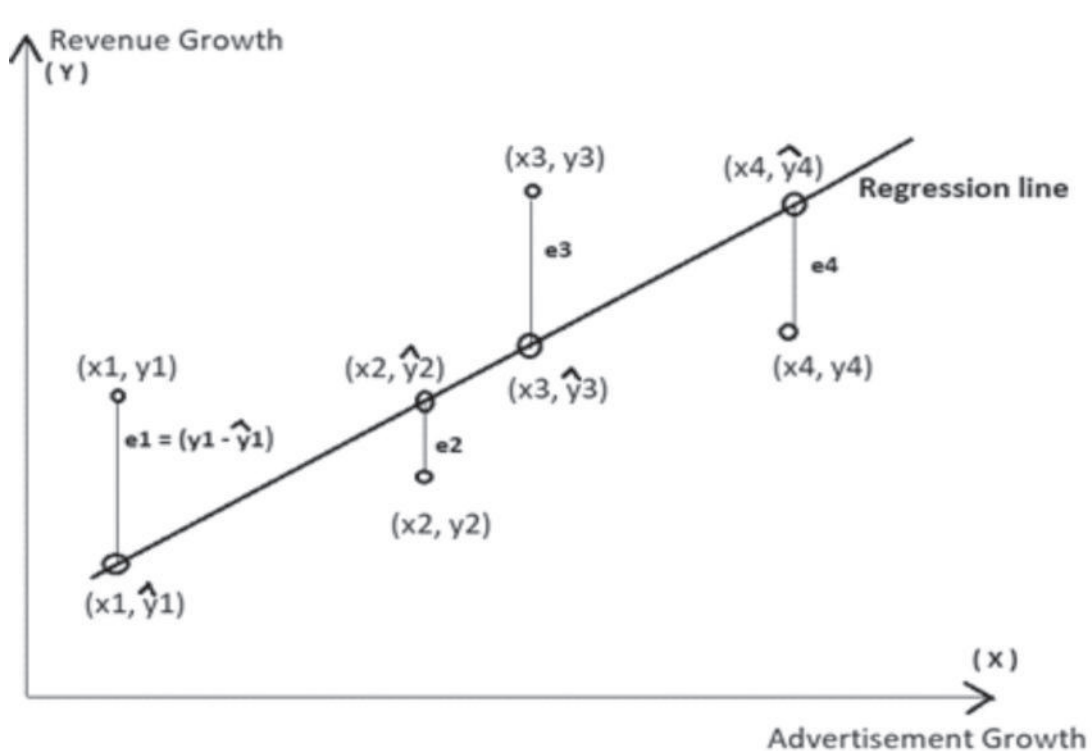


Fig – 1: Residuals as function of Actual value & estimated value

$$SSR = \sum (y_i - \hat{y}_i)^2 = e_1^2 + e_2^2 + e_3^2 + e_4^2 \quad (8)$$

= square sum of variations w.r.t to estimated value

$SST = \text{total sum of squares}$ is the sum of the distance the data is away from the mean (central tendency) all squared as shown below in Fig - 2.

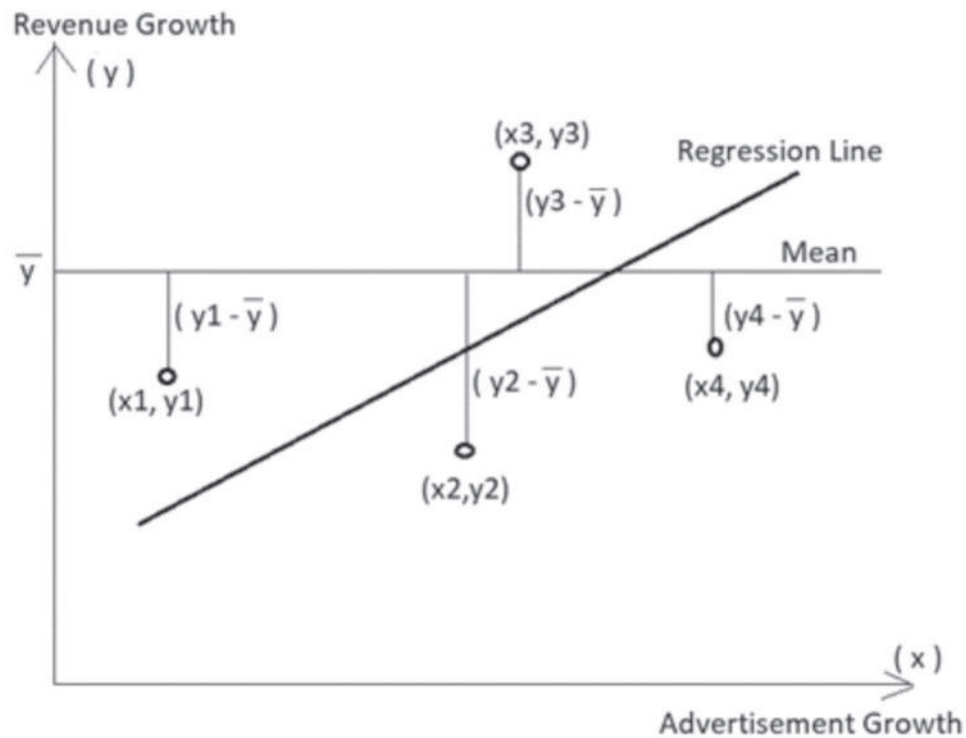


Fig – 2: Residuals as function of Actual value & Mean value

$$SST = \sum (y_i - \bar{y})^2 = (y_1 - \bar{y})^2 + (y_2 - \bar{y})^2 + (y_3 - \bar{y})^2 + (y_4 - \bar{y})^2 \quad (9)$$

$$R^2 = 1 - \frac{\text{sum squared regression (SSR)}}{\text{total sum of squares (SST)}}$$

$$R^2 = \frac{SST - SSR}{SST}$$

$$R^2 = \frac{\sum (y_i - \bar{y})^2 - \sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \quad (10)$$

The above equation indicates that R^2 is directly proportional to difference between the square sum of variations in y w.r.t mean and square sum of variations in y w.r.t estimated value.

Not good fit:

Smaller R^2 value indicates that SSR value is large and close to SST which indicates that variation in y w.r.t estimated value is large & close to variation in y w.r.t mean, which is not good fit.

Good fit:

Large R^2 value indicates that SSR value is very small (actual values of y are close to estimated values of y) and not close to SST, which indicates that variation in y w.r.t estimated value is not close to variation in y w.r.t mean, which is a good fit.

Results & Model Diagnostics

A. Estimated Model

Using python as tool parameters of regression model are calculated as shown below.

Using 80% training data set

✓ Constant $\hat{\beta}_0 = 6.101$

✓ Regression coefficient $\hat{\beta}_1 = 0.160$

The estimated model can be written as

$$Y_i = \beta_0 + \beta_1 X_i \quad (11)$$

$$\text{Rev Grw}(\%) = 6.101 + 0.160 * (\text{Ad Grw}(\%))$$

B. Interpretation of Estimated Model

Model estimates that 1% Ad Growth will increase Revenue by 0.160 %. For example, if the sales revenue was 2 Million in year 2018 then according to our model sales revenue in year 2019 will increase by 0.0032 million i.e. estimated sales revenue can be 2.0032 millions that is rise of 3200/- in revenue.

C. Model Diagnostics (Validation)

Before using regression model in practical applications, it should be validated & tested for goodness of fit. We will be using Co-efficient of determination (R-squared) method to determine goodness of fit. Using python as a tool following value of R^2 is calculated

$$R^2 = 0.208$$

According to Cohen – 1992 [9] r-square value 0.12 (12%)

or below indicate low, between 0.13 (13%) to 0.25 (25%) values indicate medium & 0.26 (26%) or above values indicate high. Our model explains 20.8% of the variance in the validation set, so it is reasonably good fit.

Conclusion

The simple linear regression model using ordinary least square (OLS) method shows functional relationship between the outcome variable (Sales revenue growth in %) and the feature (advertisement growth in %). The model validation is investigated using R^2 technique to ensure goodness of fit. While an R-square as low as 10% is generally accepted for studies in the field of arts, humanities and social sciences because human behavior cannot be accurately predicted, therefore, a low R-square is often not a problem in studies in the arts, humanities and social science field. There are various other control parameters which affects the value of R-square. Therefore, in order to extend scope of this research various social science characteristics like age, gender, motivation towards product and festive season should be included as control variables in analysis.

References:

- [1] Core Python Programming by Dr.R.Nageswara Rao second edition dreamtech Press.
- [2] Machine Learning Using Python by Manaranjan Pradhan & U Dinesh Kumar first reprint edition Wiley publications
- [3] Sales prediction types available online at URL: <https://www.economicsdiscussion.net/sales/sales-forecasting-methods/32270>
- [4] Advantages of Linear regression model available online at URL: <https://iq.opengenus.org/advantages-and-disadvantages-of-linear-regression/>
- [5] Will Koehrsen Article how to setup your-machine learning problem can be found at following URL: <https://towardsdatascience.com/prediction-engineering-how-to-set-up-your-machine-learning-problem-b3b8f622683b>
- [6] Data source of Ratios & Budgets can be found at following URL: <https://www.marketresearch.com/Schonfeld-Associates-Inc-v417/Advertising-Ratios-Budgets-13373044/>
- [7] <https://www.keboola.com/blog/linear-regression-machine-learning>.

- [8] <https://internal.ncl.ac.uk/ask/numeracy-maths-statistics/statistics/regression-and-correlation/coefficient-of-determination-r-squared.html#:~:text=R2%3D1%E2%88%92sum%20squared,from%20the%20mean%20all%20squared>.
- [9] Cohen's Conventions for Small, Medium, and Large R^2 values can be found on following URL <http://core.ecu.edu/psyc/wuenschk/docs30/EffectSizeConventions.pdf>
- [10] Small is beautiful. The use and interpretation of R^2 in social Research by Ferenc Moksony Pages 6 & 7

Certificates



The Institution of Engineers (India)

NAVI MUMBAI LOCAL CENTRE

and

Agnel Charities'

FR. C. RODRIGUES INSTITUTE OF TECHNOLOGY, VASHI



Jointly present

IEI NMLC-FCRIT EXCELLENCE AWARDS - 2023

Certificate of Participation

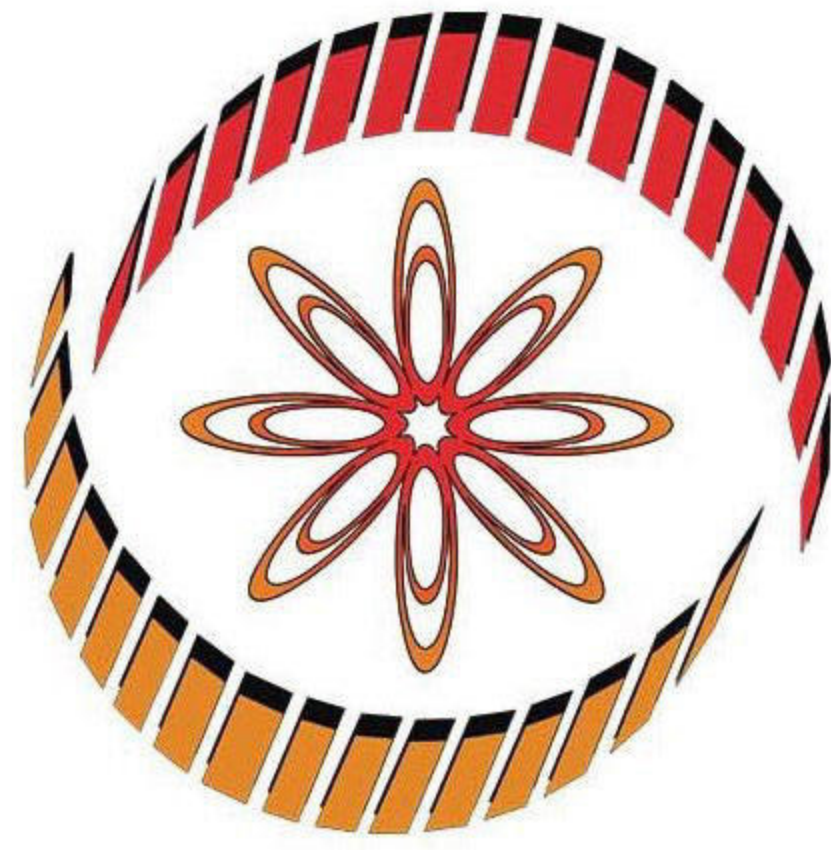
This is to certify that **Mr. Avinash Dangwani** of Pacific Academy of Higher Education and Research University, Udaipur has participated in **IEI NMLC-FCRIT EXCELLENCE AWARDS** ceremony held on September 15th, 2023 and the nomination is found to be worthy for consideration under **Research Excellence in Students (PhD) Regional Category**.

DR. NILAJ DESHMUKH
CHAIR, IEI NMLC-FCRIT
EXCELLENCE AWARDS-2023

DR. S. M. KHOT
PRINCIPAL,
FCRIT, VASHI

ER. SANJAY R. BAGUL
SECRETARY,
IEI, NMLC

DR. K. M. GODBOLE
CHAIRMAN,
IEI, NMLC



Elite

NPTEL Online Certification

(Funded by the MoE, Govt. of India)



This certificate is awarded to

AVINASH DANGWANI

for successfully completing the course



Python for Data Science

with a consolidated score of **78** %

Online Assignments	21.67/25	Proctored Exam	56.67/75
--------------------	----------	----------------	----------

Total number of candidates certified in this course: **7596**

Devendra Jalihal

Prof. Devendra Jalihal

Chairperson,
Centre for Outreach and Digital Education, IITM

Jul-Aug 2023

(4 week course)

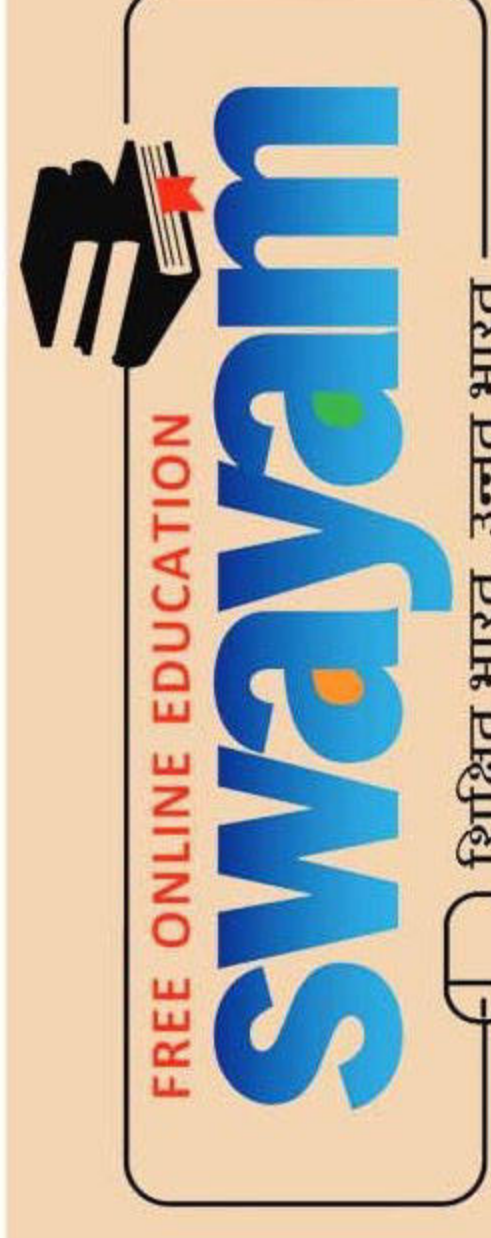
TH

Prof. Andrew Thangaraj

NPTEL, Coordinator
IIT Madras



Indian Institute of Technology Madras



Roll No: NPTEL23CS99S35002545

To verify the certificate



No. of credits recommended: 1 or 2



Elite

NPTEL Online Certification

(Funded by the MoE, Govt. of India)



This certificate is awarded to

AVINASH DANGWANI

for successfully completing the course

Data Analytics with Python

with a consolidated score of **82** %

Online Assignments	24.69/25	Proctored Exam	57.45/75
--------------------	----------	----------------	----------

Total number of candidates certified in this course: **7860**

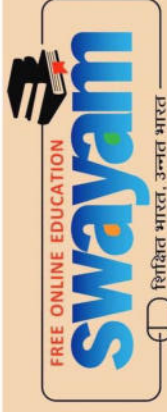
Prof. Sanjeev Manhas
Coordinator, Continuing Education Centre
IIT Roorkee

Jan-Apr 2023
(12 week course)

Prof. Priti Maheshwari
NPTEL Coordinator
IIT Roorkee



Indian Institute of Technology Roorkee



Roll No: NPTEL23CS08S54221311

To validate the certificate



No. of credits recommended: 3 or 4



DEPARTMENT OF PG STUDIES AND DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(Pacific Academy of Higher Education and Research University, Udaipur)

INTERNATIONAL VIRTUAL CONFERENCE on EMERGING ERA OF APPLICATIONS OF COMPUTER

Certificate of Merit

This is to certify that *Mr. Arunash Sunder Dangwani*
of *Pacific Academy of Higher Education & Research University*
has been awarded as Young Scientist for Best Oral Presentation in this International Virtual
Conference on Emerging Era of Applications of Computer on 15th-16th January 2022.

Prof. Hemant Kothari
Conference Chairman
PAHER University, Udaipur

Dr. Dilendra Hiran
Conference Co-Chairman
PAHER University, Udaipur

Dr. Tanveer Ahmed Kazi
Conference Director
PAHER University, Udaipur

Dr. Prashant Sharma
Conference Organizing Secretary
PAHER University, Udaipur



PACIFIC ACADEMY OF HIGHER EDUCATION AND RESEARCH UNIVERSITY, UDAIPUR

IQAC Cell

in association with



PACIFIC BUSINESS SCHOOL, UDAIPUR Faculty Development Program

January 2-7, 2021

Certificate of Participation

This is certified that

*Dr./Ms./Mr. **Avinash Dangwani, Associate Professor (Selection Grade)***

*of **VES Polytechnic***

has successfully participated in

Faculty Development Program on 'Research and Publication Ethics'

Prof. K.K. Dave
President, PAHER University
(FDP Chair)

Prof. Hemant Kothari
Dean, P.G. Studies, PAHER University
(Convener FDP)

Prof. Mahima Birla
Dean, Faculty of Management
(Director FDP)

Prof. Dipin Mathur
Director, Pacific Business School
(Co-convenor FDP)



CERTIFICATE OF COMPLETION

THIS IS TO CERTIFY THAT DR./PROF./MR./MS.

Avinash Dangwani

has successfully completed Live Online Instructor-led
10-Days (35 Hours) Faculty Development Program / Short Training on

Applied Deep Learning for Medical Data Analysis

[MRI, CT-SCAN, ECG, & COVID-19 Chest X-RAY]

From 27th October to 9th November, 2020

Organized by

EduxLabs (Esoir Business Solution LLP)

ISSUED ON:

10TH NOVEMBER 2020



Esoir Business Solution LLP

MD. NAFISH

MD.NAFISH Director

EDUXLABS DIRECTOR

(ESOIR BUSINESS SOLUTION)

Plagiarism Report

To Explore and Analyze the Role of IOT, Artificial Intelligence and Machine Learning in Solving the Commuting Problems of Smart Cities

by Avinash Dangwani

Submission date: 25-Apr-2024 10:41AM (UTC+0530)

Submission ID: 2361197734

File name: Final_Thesis_Formatted__Copy2.pdf (10.72M)

Word count: 55751

Character count: 308164

**To Explore and Analyze ¹the Role of IOT,
Artificial Intelligence and Machine
Learning in Solving the Commuting
Problems of Smart Cities**

By

AVINASH DANGWANI

To Explore and Analyze the Role of IOT, Artificial Intelligence and Machine Learning in Solving the Commuting Problems of Smart Cities

ORIGINALITY REPORT

6%

SIMILARITY INDEX

5%

INTERNET SOURCES

4%

PUBLICATIONS

2%

STUDENT PAPERS

PRIMARY SOURCES

1

www.researchgate.net

Internet Source

1%

2

fastercapital.com

Internet Source

1%

3

Kuldeep Singh Kaswan, Jagjit Singh Dhatteval, Vivek Jaglan, Balamurugan Balusamy, Kiran Sood. "Enabling Technologies for Smart Fog Computing", Institution of Engineering and Technology (IET), 2023

Publication

<1%

4

en.atlasconference.org

Internet Source

<1%

5

ijarse.com

Internet Source

<1%

6

dokumen.pub

Internet Source

<1%

7

Kashif Iqbal, Muhammad Adnan, Sagheer Abbas, Zahid Hasan, Areej Fatima. "Intelligent

<1%

Transportation System (ITS) for Smart-Cities using Mamdani Fuzzy Inference System", International Journal of Advanced Computer Science and Applications, 2018

Publication

8

techfor-today.com

Internet Source

<1 %

9

Chu Xiao Hui, Ge Dan, Sagr Alamri, Davood Toghraie. "Greening smart cities: An investigation of the integration of urban natural resources and smart city technologies for promoting environmental sustainability", Sustainable Cities and Society, 2023

Publication

<1 %

10

testmagazine.biz

Internet Source

<1 %

11

Damilola Oladimeji, Khushi Gupta, Nuri Alperen Kose, Kubra Gundogan, Linqiang Ge, Fan Liang. "Smart Transportation: An Overview of Technologies and Applications", Sensors, 2023

Publication

<1 %

12

N. Jothy, Komala James, N. Subhashini, A. K. Mariselvam. "chapter 13 Efficient Parking Solutions Powered by IoT and Transportation Integration", IGI Global, 2023

Publication

<1 %

13	www.leewayhertz.com Internet Source	<1 %
14	"Interconnected Modern Multi-Energy Networks and Intelligent Transportation Systems", Wiley, 2024 Publication	<1 %
15	mdpi-res.com Internet Source	<1 %
16	ijsrst.com Internet Source	<1 %
17	studylib.net Internet Source	<1 %
18	www.ncbi.nlm.nih.gov Internet Source	<1 %
19	Submitted to Liverpool John Moores University Student Paper	<1 %
20	robots.net Internet Source	<1 %
21	tnsroindia.org.in Internet Source	<1 %
22	Submitted to HCUC Student Paper	<1 %
23	eprajournals.com Internet Source	<1 %

24	"Recent Advances in Intelligent Systems and Smart Applications", Springer Science and Business Media LLC, 2021 Publication	<1 %
25	"Integrated Intelligent Computing, Communication and Security", Springer Science and Business Media LLC, 2019 Publication	<1 %
26	Ismail Zrigui, Samira Khouli, Mohamed Larbi Kerkeb, Ayoub Ennassiri, Salmane Bourekadi. "Reducing Carbon Footprint with Real-Time Transport Planning and Big Data Analytics", E3S Web of Conferences, 2023 Publication	<1 %
27	Submitted to Bahrain Polytechnic Student Paper	<1 %
28	Submitted to De Montfort University Student Paper	<1 %
29	Elizeu Jacques, Alvaro Neuenfeldt Júnior, Sabine De Paris, Matheus Francescato, Julio Siluk. "Smart cities and innovative urban management: Perspectives of integrated technological solutions in urban environments", Heliyon, 2024 Publication	<1 %
30	Submitted to Erasmus University of Rotterdam	<1 %

31

Submitted to South Bank University

Student Paper

<1 %

32

Tinku Singh, Majid Kundroo, Taehong Kim.
"WSN-Driven Advances in Soil Moisture
Estimation: A Machine Learning Approach",
Electronics, 2024

Publication

<1 %

33

www.duvaryayinlari.com

Internet Source

<1 %

34

www.ijirset.com

Internet Source

<1 %

35

Fuli Zhang, Ling Zhou, Zhichen Wang, Congna
Lv, Qi Zhang, Jing Wang, Jing Zhang,
Yongpeng Zhang. "Empowering urban energy
transition through data-driven decision-
making: A statistical examination of
technological innovations in transportation
and mobility", Sustainable Cities and Society,
2024

Publication

<1 %

36

Submitted to Nottingham Trent University

Student Paper

<1 %

37

Submitted to Pacific University

Student Paper

<1 %

38	Barrios, Romero. "Decision Tree Methods for Predicting Surface Roughness in Fused Deposition Modeling Parts", Materials, 2019 Publication	<1 %
39	Submitted to University of Leicester Student Paper	<1 %
40	docs.neu.edu.tr Internet Source	<1 %
41	"Intelligent Road Transport Systems", Springer Science and Business Media LLC, 2022 Publication	<1 %
42	Submitted to BPP College of Professional Studies Limited Student Paper	<1 %
43	Submitted to Bournemouth University Student Paper	<1 %
44	www.geeksforgeeks.org Internet Source	<1 %
45	Husain Aljazzar, Stefan Leue. "K _* : A heuristic search algorithm for finding the k shortest paths", Artificial Intelligence, 2011 Publication	<1 %
46	Parikshit N. Mahalle, Pravin P. Hujare, Gitanjali Rahul Shinde. "Predictive Analytics for Mechanical Engineering: A Beginners	<1 %

47

Mohammed M. Abo-Zahhad. "A Methodology for the Design of IoT-Based Intelligent Vehicular Management Systems in Smart Cities", 2022 10th International Japan-Africa Conference on Electronics, Communications, and Computations (JAC-ECC), 2022

Publication

<1 %

48

Open Government, 2014.

Publication

<1 %

49

www.giiresearch.com

Internet Source

<1 %

50

Kesavan Gunasekaran, V Vinoth Kumar, A. C. Kaladevi, T R Mahesh, C Rohith Bhat, Krishnamoorthy Venkatesan. "Smart Decision-making and Communication Strategy in Industrial Internet of Things", IEEE Access, 2023

Publication

<1 %

51

frostandullivaninstitute.org

Internet Source

<1 %

52

"Image Processing and Capsule Networks", Springer Science and Business Media LLC, 2021

Publication

<1 %

53	Submitted to Harrisburg University of Science and Technology Student Paper	<1 %
54	iosrjournals.org Internet Source	<1 %
55	listens.online Internet Source	<1 %
56	repository.iaa.ac.tz:8080 Internet Source	<1 %
57	www.iieta.org Internet Source	<1 %
58	www.intechopen.com Internet Source	<1 %
59	Submitted to Karabük Üniversitesi Student Paper	<1 %
60	Lecture Notes in Computer Science, 2005. Publication	<1 %
61	Submitted to Manchester Metropolitan University Student Paper	<1 %
62	Umesh Kumar Lilhore, Agbotiname Lucky Imoize, Chun-Ta Li, Sarita Simaiya et al. "Design and Implementation of an ML and IoT Based Adaptive Traffic-Management System for Smart Cities", Sensors, 2022 Publication	<1 %

63	download.atlantis-press.com Internet Source	<1 %
64	ebin.pub Internet Source	<1 %
65	Submitted to Arts, Sciences & Technology University In Lebanon Student Paper	<1 %
66	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1 %
67	Gongquan Zhang, Fangrong Chang, Jieling Jin, Fan Yang, Helai Huang. "Multi-objective deep reinforcement learning approach for adaptive traffic signal control system with concurrent optimization of safety, efficiency, and decarbonization at intersections", Accident Analysis & Prevention, 2024 Publication	<1 %
68	Submitted to Queensland Academy of Health Sciences Student Paper	<1 %
69	link.springer.com Internet Source	<1 %
70	www.ijraset.com Internet Source	<1 %

71	Gen Chen, Jia wan Zhang. "Intelligent transportation systems: Machine learning approaches for urban mobility in smart cities", Sustainable Cities and Society, 2024 Publication	<1 %
72	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
73	Nguyen Anh Phong, Nguyen Cao Long. "Netflix Stock Price Movements Prediction Using News Sentiment Analysis", Research Square Platform LLC, 2024 Publication	<1 %
74	Submitted to University of Fort Hare Student Paper	<1 %
75	Submitted to University of Wales Institute, Cardiff Student Paper	<1 %
76	rfppl.co.in Internet Source	<1 %
77	www.mdpi.com Internet Source	<1 %
78	www2.mdpi.com Internet Source	<1 %
79	"Proceedings of International Conference on Machine Intelligence and Data Science	<1 %

Applications", Springer Science and Business Media LLC, 2021

Publication

80

Submitted to Brunel University

Student Paper

<1 %

81

Submitted to Polytechnics Mauritius

Student Paper

<1 %

82

Submitted to Quest International University
Perak

Student Paper

<1 %

83

Submitted to Swinburne University of
Technology

Student Paper

<1 %

84

Submitted to University of Hertfordshire

Student Paper

<1 %

85

Vladimir Baskov, Ekaterina Isaeva.
"Information Support of Efficiency of the
Transportation Process", MATEC Web of
Conferences, 2021

Publication

<1 %

86

developers.google.cn

Internet Source

<1 %

87

etis.ee

Internet Source

<1 %