

REAL TIME BIG DATA ANALYTICS FOR PUBLIC SAFETY IN SMART CITY

Thesis

Submitted in partial fulfilment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

Manjunatha



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE - 575 025

October, 2023

REAL TIME BIG DATA ANALYTICS FOR PUBLIC SAFETY IN SMART CITY

Thesis

Submitted in partial fulfilment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

Manjunatha

(155120CS15FV07)

Under the guidance of

Dr. Annappa

Professor, Dept of CSE, NITK



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA


SURATHKAL, MANGALORE - 575 025

October, 2023

DECLARATION

by the Ph.D. Research Scholar

I hereby declare that the Research Thesis entitled **Real-time Big Data Analytics for Public safety in Smart City** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfilment of the requirements for the award of the Degree of **Doctor of Philosophy** in Department of Computer Science and Engineering is a bonafide report of the research work carried out by me. The material contained in this Research Thesis has not been submitted to any University or Institution for the award of any degree.



Manjunatha

Register Number. 15120CS15FV07

Department of Computer Science and Engineering

Place: NITK, Surathkal.

Date: October 17, 2023

CERTIFICATE

This is to certify that the Research Thesis entitled **Real-time Big Data Analytics for Public safety in Smart City** submitted by **Manjunatha** (Register Number: 155120CS15FV07) as the record of the research work carried out by him, is accepted as the Research Thesis submission in partial fulfillment of the requirements for the award of degree of **Doctor of Philosophy**.

Dr. Annappa
Prof.
Dept. of Computer Science and Engineering
National Institute of Technology
Surathkal, Post
MANGALORE
annappa

Annappa 18/10/23

Dr. Annappa
Research Guide

(Signature with Date and Seal)

Chairman
DUGC / DPGC / DRPC
Dept. of Computer Engg.
NITK - Surathkal
Srinivasnagar - 575 025

Manu Basavaraju 18/10/23

Dr. Manu Basavaraju
Chairman - DRPC

(Signature with Date and Seal)

ACKNOWLEDGEMENT

I would like to take this opportunity to express my sincere gratitude and appreciation to all the individuals and institutions who have supported and contributed to the completion of this thesis.

Foremost, I would like to express my sincere gratitude to my Supervisor Dr. Annappa, Professor in Department of Computer Science & Engineering for his continuous encouragement, patience, motivation, enthusiasm, and immense knowledge. His guidance and insightful comments helped me in all the time of research and writing of this thesis. His endless support, trust, patience and honest feedback made this achievement possible. Thank you, Sir.

My sincere thanks go to research progress committee members Dr. B. R. Chandavarkar, Assistant Professor in Department of Computer Science & Engineering, and Dr. Nagamma Patil, Assistant Professor in Department of Information Technology, for giving their valuable suggestions, inspiration and moral support while evaluating our work time to time.

I humbly thank Dr. Manu Basaraju, HoD, Department of Computer Science & Engineering for helping me in research related aspects. I whole heartedly express my gratitude to Dr. P. Santhi Thilagam, Dr. Alwyn Roshan Pais, and Dr. Shashidhar G Koolagudi, Department of Computer Science & Engineering, for their support during my research work. I am also grateful to all faculty and staff members of Computer Science & Engineering Department for their generous support throughout this work. I acknowledge the kindhearted support from the Department of Computer Science & Engineering and National Institute of Technology Karnataka by providing all the resources and facilities for this research work. I also acknowledge Ministry of Electronics and Information Technology (MeitY), Government of India, for their

support in a part of the research.

I extend my appreciation to my colleagues and friends for their companionship and motivation during the ups and downs of this thesis. Their constructive discussions, brainstorming sessions, and moral support have been invaluable and have contributed significantly to the development of my ideas. Furthermore, I would like to thank my beloved wife and other family members for their unconditional love, encouragement, and understanding. Their constant belief in my abilities has been the driving force behind my academic pursuits.

Although it is challenging to list every individual who has contributed to this thesis, I want to acknowledge all those unnamed mentors, friends, and well-wishers whose impact on my life and work cannot be overstated.

In conclusion, this thesis would not have been possible without the generous contributions of all those mentioned above and the countless others who have played a part, however small, in shaping my academic and personal journey.

Thank you all.

Manjunatha

ABSTRACT

Advancement in Information Communication Technology (ICT) and the Internet of Things (IoT) has led to the continuous generation of a large amount of data. Smart city projects have been implemented in various parts of the world where public data analysis helps provide a better quality of life. Recently, big data analytics has played an essential role in many data-driven applications. Big data technologies are moving towards knowledge discovery from the raw data in real time. Real-time data analytics is essential in industries such as finance, healthcare and e-commerce, where decisions need to be made quickly to stay competitive. Multiple data sources also enable organizations to gain a more complete picture of their business, customers and operations. However, processing and analyzing data from multiple sources in real-time present significant challenges including data integration, data quality and scalability. To overcome these challenges, organizations must have a well-designed architecture, the right tools and technologies, skilled data analysts and engineers.

Real-time analytics for finding valuable insights at the right time using smart city data is crucial in making appropriate decisions for city administration. It is essential to use multiple data sources as input for the analysis to achieve more accurate data-driven solutions. Public safety is one of the major concerns in any smart city project in which real-time analytics is useful in the early detection of valuable data patterns. It is crucial to find early predictions of crime-related incidents and generate emergency alerts for making appropriate decisions to provide security to the people and the safety of the city's infrastructure. In this research, we propose a real-time big data analytics framework using multiple data sources from a smart city to find better data-driven solutions for public safety. Public safety is one of the major concerns in any smart city project where real-time analytics is much useful to provide security to the people and safety to city infrastructure. Analytics using multiple data sources for a specific data-driven solution helps to find more data patterns, increasing the accuracy of

analytics results. Data preprocessing is challenging in data analytics when data is ingested continuously in real time into the analytics system. The proposed system helps preprocess the real-time data with data blending of multiple data sources used in the analytics.

The proposed framework is beneficial when data from various sources are ingested in real time as input data and is also flexible to use any additional data source of interest. The experimental work was carried out with the proposed framework using multiple data sources to find real time crime-related insights that help the smart city's public safety solutions. The experimental outcome using the proposed data blending mechanism shows a significant increase in the number of identified useful data patterns as number of data sources increase. A real time emergency alert system to help the public safety solution is implemented using a machine learning based classification model with the proposed framework. Early detection and crime prevention are critical challenges to provide better safety within the city. Predictive policing has been around for a long time to monitor crimes based on past crime records by identifying the crime hotspots. Previous works on crime prediction have used historical data from specific sources to build the prediction model. In this work, real time data from multiple sources and historical data are used to improve the prediction model's performance.

CONTENTS

List of Figures	viii
List of Tables	ix
List of Abbreviations	xi
1 Introduction	1
1.1 Real-time Big data analytics	2
1.1.1 Big Data	3
1.1.2 Analytics	4
1.2 Real-time analytics in Smart city	5
1.2.1 Smart City	5
1.2.2 Analytics in Smart city	6
1.3 Real-time analytics for public safety	9
1.4 Motivation	10
1.5 Thesis Contributions	12
1.6 Thesis Organization	12
2 Literature Review	15
2.1 Real-time Big data Analytics	15
2.1.1 Real time big data analytics in twitter data	17
2.1.2 Real Time Big Data Analytics for Healthcare	18
2.1.3 Real Time Big Data Analytics for alerts and monitoring System.	19
2.2 Smart City data and Public safety	21
2.2.1 Real time big data analytics for public safety in the city	25
2.2.2 Data driven solutions using multiple data sources:	29

2.3	Research Gaps	31
2.4	Problem Statement:	31
2.5	Problem Description:	32
2.6	Objectives	32
2.7	Summary	32
3	Real-time Big data Analytics framework	35
3.1	Introduction	35
3.1.1	Lambda Architecture	36
3.1.2	Kappa Architecture	36
3.2	Proposed Design	38
3.2.1	Data ingestion	39
3.2.2	Data preprocessing	39
3.2.3	Data storage	41
3.2.4	Real-time analytics and Visualization	41
3.3	Tools and Evaluation Metrics	42
3.4	Experimental Evaluation	49
3.4.1	Data Ingestion Processor	50
3.4.2	Data blending mechanism	52
3.5	Discussion	55
3.6	Summary	57
4	Real-time emergency event detection system	59
4.1	Introduction	60
4.2	Public safety in Smart city	61
4.3	Proposed Work	63
4.4	Experimental Evaluation	65
4.4.1	Event Processor	66
4.5	Results and Discussion	68
4.6	Summary	71

5 Real-time analytics based crime prediction using multiple data sources	73
5.1 Introduction	74
5.2 Crime hotspot and predictive policing	75
5.3 Proposed Work	78
5.4 Experimental Evaluation	81
5.4.1 Data Ingestion Processor	83
5.4.2 Data Blending Adapter	84
5.4.3 Real-time analytics for crime prediction	85
5.5 Results and Discussion	86
5.6 Summary	94
6 Conclusions and Future Scope	97
6.1 Conclusion	97
6.2 Future Scope	98
Bibliography	100
Publications	113

LIST OF FIGURES

1.1 Real-time big data analytics process	5
1.2 Real-time big data analytics for Smart city	8
3.1 Lambda Architecture	37
3.2 Kappa Architecture	37
3.3 Proposed design for real-time big data analytics	40
3.4 Experimental setup for real-time big data analytics using multiple data sources	48
3.5 Workflow of data ingestion processor	51
3.6 Sample observation of data blending of real-time data from multiple sources	56
3.7 Sample observation of data blending of real-time data of different categories of crime using multiple data sources	57
4.1 Workflow of real-time emergency event detection using multiple data sources	64
4.2 Experimental setup for real-time emergency event detection using multiple data sources	67
4.3 Performance Comparison of Classification Algorithms for Emergency Alerts- (a) Logistic Regression and (b) Naive Bayes classifier	69
4.4 Performance Comparison of Classification Algorithms for Emergency Alerts- (a) Random Forest and (b) Linear SVM	70
5.1 Proposed design for real-time crime prediction using multiple data sources	79
5.2 Experimental setup for real-time prediction of Crime hotspots from multi-source data	82

5.3	Work-flow of Data Ingestion Process	83
5.4	Work-flow of Data Blending Adapter	84
5.5	Performance Comparison between single source and blending of sources	87
5.6	Prediction performance of different classification algorithms with blended data (a) Logistic Regression and (b) Naive Bayes classifier . . .	90
5.7	Prediction performance of different classification algorithms with blended data (a) Random Forest and (b) Linear SVM	91
5.8	Prediction performance comparison for different testing period- (a) Accuracy and (b) Precision	92
5.9	Prediction performance comparison for different testing period- (a) Recall and (b) Average Performance	93

LIST OF TABLES

2.1	Summary of Literature:Real Time Big Data Analytics	20
2.2	Summary of Literature:Real time big data analytics for public safety	28
5.1	Comparison of prediction performance of different classifiers with	
	blended data	88

LIST OF ABBREVIATIONS

Abbreviations	Expansion
AI	Artificial Intelligence
API	Application Program Interface
BDA	Big Data Analytics
CI	Continuous Intelligence
DL	Deep Learning
ETL	Extract, Transform & Load
FNR	False Negative Rate
FPR	False Positive Rate
GIS	Geographical Information System
HDFS	Hadoop Distributed File System
ICT	Information and Communication Technology
IoT	Internet of Things
IT	Information Technology
KDE	Kernel Density Estimation
LDA	Latent Dirichlet Allocation
LR	Logistic Regression
ML	Machine Learning
MR	Map Reduce
NB	Naive Bayes
RCM	Rich Context Model
RDD	Resilient Distributed Datasets
RF	Random Forest

Abbreviations	Expansion
SVM	Support Vector Machine
TNR	True Negative Rate
TPR	True Positive Rate

CHAPTER 1

INTRODUCTION

Data and analytics play an important role in almost all business applications across a wide range of industries. A confluence of advances in the technologies like big data, analytics, Artificial Intelligence (AI), and Machine Learning (ML) are the key attention of data analysts and top-level administrators of various enterprises. The tools and technologies for data analytics are continuously evolving, which creates opportunities for enterprises to prepare for transformations and challenges ahead. Integrating Information Technology (IT) across each organization leads to the creation of large amount of data. The digitization and growth in the number of mobile devices and sensors create ample space for generating data on a large scale. The total amount of digital data created worldwide will rise to 175 zettabytes by 2025, which was 40 zettabytes in 2019 as per the report from IDC Data Age 2025 (Reinsel et al. 2017). In this digital world, it is essential for organizations to become more data-driven and harness the data for valuable insights. As per Fortune Business Insights reports, the big data and analytics market was valued at 168.8 billion US dollars in 2018 and is expected to grow to 655.3 billion US dollars by 2029 (Report 2022). With this exponential growth, harnessing the data for useful insights is one of the challenging tasks for data analysts. By 2025, more than 90 percent of the most successful companies in the world will be deploying real-time intelligence and event-streaming technologies to improve data-driven decisions (Hopkin 2023). It is important for

organizations to focus more on analyzing real-time data rather than historical data for better solutions.

Real-time analytics refers to the process of analyzing and interpreting data in real-time as it is generated, rather than after it has been collected and stored. This means that data is analyzed as it is produced, providing immediate insights and actionable intelligence. Real-time analytics involves the use of technologies such as stream processing, machine learning, and artificial intelligence to process and analyze large volumes of data in real-time. This approach is particularly valuable in situations where quick decision-making is required, such as in financial trading, fraud detection, predictive maintenance, and real-time marketing. Real-time analytics can help organizations gain a competitive edge by allowing them to respond quickly to changing market conditions, customer behavior, and emerging trends.

1.1 REAL-TIME BIG DATA ANALYTICS

In simpler terms, real-time analytics refers to the ability to process the new data as and when it is generated in order to make data-driven decisions in real-time. As per the Gartner definition (Gartner 2020), "Real-time analytics is a discipline that uses logic and mathematics to analyze data in order to provide insights that help people make better decisions faster". Continuous Intelligence (CI) is an emerging trend in Big data analytics, in which real-time analytics are integrated into processing historical and real-time data to design a specific data-driven solution.

In the computing context, real-time data processing implies performing an operation on data just after it is generated. For some use cases, real-time ensures that the analytics are done within seconds or minutes of new data arriving. Real-time data, fast data, streaming data are the most emerging terms used in recent development in the big data world. For example, the applications involving monitoring of the environment should use the data generated continuously like temperature information, humidity values, etc. The analytics on this environmental data to find any predictions or data-driven decisions should use both historical data stored as well as real-time data gets generated continuously. Dynamic data generated continuously from different sources

is considered as streaming data. The data comes from social media feeds or sensors, IoT devices, and cameras; each record needs to be processed in a way that preserves its relation to other data and sequence in time. Fast data refers to a concept related to the processing and analysis of data in real-time or near-real-time. It involves handling and deriving insights from large volumes of data that are generated rapidly or at high velocity (Olmezogullari and Ari 2013). Fast data typically refers to streaming data, such as sensor data, social media feeds, log files, financial market data, and more. Here the volume of data generated per unit time interval in real-time is an important factor for further analytics.

1.1.1 Big Data

The 'big data' paradigm is expanding rapidly in recent days, where the term big data used for datasets that are so large that they cannot be processed and managed using traditional database concepts. The requirement for big data is working with data of any size. The term 'Big data' refers to all the data that is being created across the globe at an exponential rate. This data could be either structured or unstructured. Big data differs from traditional data in its volume, velocity, and variety, which are three V's defining big data.

- Volume-The data generated from different applications and services are huge in volume, which cannot be handled with our traditional database system. Storage systems should be able to manage terabytes and petabytes of data.
- Velocity- Applications and services generate data continuously. The storage and analytics system must be capable of handling these data. Real-time data is most important for timely decisions and predictions.
- Variety- The data generated is in different formats like text, audio, video, images. All these different formats of data from various sources are very important in the analysis.

The recent definitions of big data also include five V's, seven V's, eight V's with Veracity, Variability, Value, Visualization, and Validity. Veracity is about the

trustworthiness of the data, which is how accurate and truthful the data is?. Variability refers to the inconsistency of the data where the meaning of the data is constantly changing. Value of the data is a more crucial characteristic referring to its business value. Visualization refers to representing or visualizes the data in more meaningful ways. Validity refers to the correctness or how accurate the data for a specific purpose.

1.1.2 Analytics

Analytics is the process that takes data as input from different sources, investigates it for valuable patterns, interprets those patterns, and finally communicates the results as per the required solution. It uses statistics, mathematics, and predictive models to find the knowledge from the datasets that are complex to analyze manually. Recent advancements in technology has increased the power of analytics. Many analytics tools in recent days are integrated with advanced technologies like pattern recognition, machine learning, and deep learning that help in better performance.

Technological advancement in data analytics is changing the business process by enabling faster and better decisions based on real-time analytics. When data analysts can harness valuable insights from data faster, it has a significant advantage in reducing costs, increasing efficiency and profit. Extracting valuable insights from raw data in real-time is critical for many real-time applications. The demand for real-time analytics is high in recent days in various fields where data-driven solutions are being used. In most data-driven solutions, real-time processing of data for making timely decisions can enhance the quality of service, improve the accuracy of predictions, and helps to make early decisions. It is challenging for the data analysts to process data from multiple sources in real-time for a specific analytical solution. The analytics outcomes are more effective and accurate when more data from appropriate data sources get processed for a specific analytical solution.

Figure [1.1](#) shows different processes involved in real-time big data analytics. Depending on the purpose of the analysis, it may consider both real-time data and historical data for the analysis. It consists of both real-time process and batch process systems. Batch processing is the processing of a large volume of data collected over a

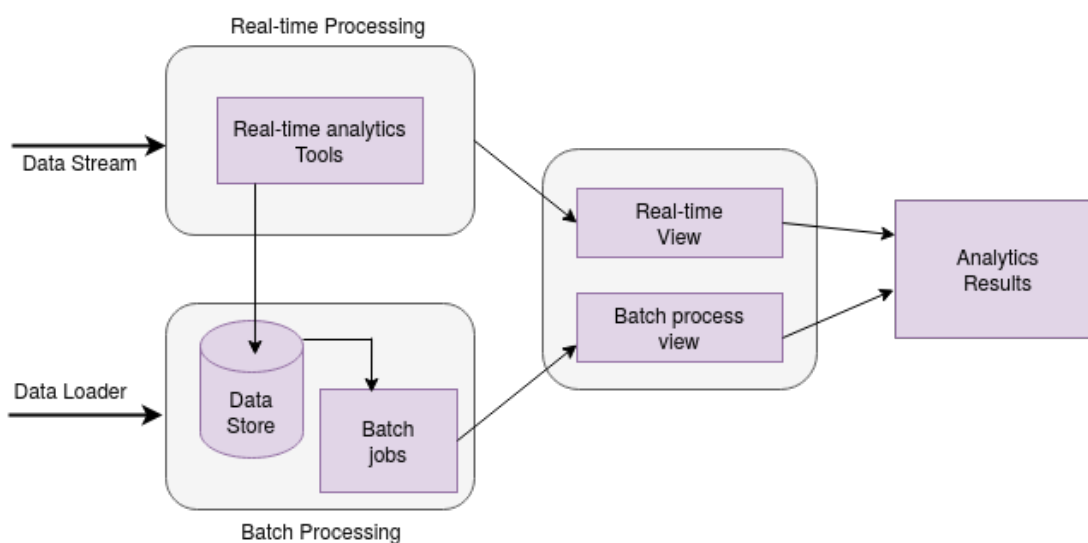


Figure 1.1: Real-time big data analytics process

period of time. Batch processing jobs are completed simultaneously nonstop, in sequential order. Real-time processing is the processing of the data instantly as and when data streams are collected from the data source. Analytics results are the results from the real-time views or batch views or the combined results of both.

1.2 REAL-TIME ANALYTICS IN SMART CITY

1.2.1 Smart City

Urban development is an important issue for any government as the urban population is increasing around the world in recent days, as stated by (Dirks et al. 2010). It is an essential task for any government to offer better services to the residents and managing those services in the city. Smart cities are popular urban development projects integrated with Information and Communication Technology (ICT) helps in the sustainability of the cities for a long time in terms of quality of life. Many smart city definitions are given by the industry and academia, some interesting definitions can be found in (McMillan et al. 2016), (Chourabi et al. 2012), (Vilajosana et al. 2013), and (Ismail 2016). The main goal of the smart city project is to provide better services for the people living in the cities. It includes better management of cities with good infrastructure within the city along with smart services. The most common services offered in the smart cities are effective traffic and parking management, monitoring of environment and

managing infrastructure, better platform for health monitoring and management, better security and safety to the citizens for a comfortable stay, creating opportunities for smart agriculture, business and skill development and many more. All these services and applications are integrated with Information and Communication Technology. Recent development in the field of Internet of Things also helps in a great deal to achieve this. Any smart city project design should focus on the following major components in its framework.

- People – Includes citizens living in the city, administrators, technical and non-technical people involved in managing the services, visitors to the city.
- Infrastructure - Important component and not limited to physical infrastructure including corporate and resident buildings, transport system like roads, rail tracks, parking spaces etc, power and water supply units, network infrastructure including telephone, mobile, internet, environment including natural resources.
- Services and Applications- Different services and applications for managing the infrastructure within the city and services for the people such as traffic management, waste management, health monitoring, parking management, safety and security management etc.
- Devices and Technology– Devices for monitoring the infrastructure in the city such as video surveillance cameras, different sensors, wireless access points etc. Technologies including data management and storage, cloud technology, software for managing the devices, etc.

1.2.2 Analytics in Smart city

Integrating advanced technologies in smart city projects leads to generate a vast amount of data. Various sensors, mobile devices, video surveillance cameras, social networks, and many intelligent applications are used to provide different services for people within the city. Digitization leads to produce a vast amount of data that can be used to make meaningful decisions and predictions for better services within the city. The data

generated on a large scale within the city are in different forms like text, audio, video, images. The big data tools and technologies help to analyze the data generated from different sources to discover valuable insights from it. The data analysts are focusing on advanced data-driven solutions by extracting valuable data patterns in real-time. Many intelligent applications and services offered in smart cities are generating data in continuous or streaming. Analysis of such data in real-time to find valuable insights are helpful to make decisions at the right time and make early predictions.

Big data technologies and applications play a crucial role in smart city projects for data-driven solutions. The challenge for the data analysts involved in these smart city projects is to make timely decisions and figure out the early predictions by analyzing the data in real-time. In smart cities, the data gets generated continuously in real-time from different applications, devices, and social media on a large scale. The valuable insights derived from the data generated within the city help effective management and administration of the city services. The data-driven solutions are widely used in smart city applications such as smart traffic management, smart parking systems, smart environment, smart policing, smart healthcare, etc. A vast amount of user-generated content within the city is analyzed to find valuable insights to enhance the services and performance of smart city applications. In turn, finding valuable data patterns in real-time greatly help in improving the performance of smart city applications and quality of service.

Figure 1.2 is the schematic representation of real-time big data analytics system for smart city applications. There are multiple data sources in smart city projects generated in real-time that can be used for real-time analytics. These data are to be collected and stored for further process in the analytics system. Since the analytics to be done using real-time data, a better data flow to be supported for data ingestion into the analytics system in real-time. Here Apache NiFi is used for this mechanism for data flow in real-time (NiFi 2017). Apache NiFi does a secure and reliable transfer of data from different sources to the analytical platform. Here NiFi brings the data from the sources from wherever we wish to collect and makes it sense to and from Apache Kafka (Kafka 2017). The data streams being pushed to Kafka are consumed by Apache Flink for

1. Introduction

further process in the analytics. Apache Flink helps in achieving real-time analytics from the data pushed to it (Flink 2017). The data expected to be used in the later stages are stored with the help of Apache Hadoop (Hadoop 2016). The insights or results in the analytics process are used for appropriate visualization and alerts and used to find any recommendations, predictions, and data-driven decisions as per the application requirements.

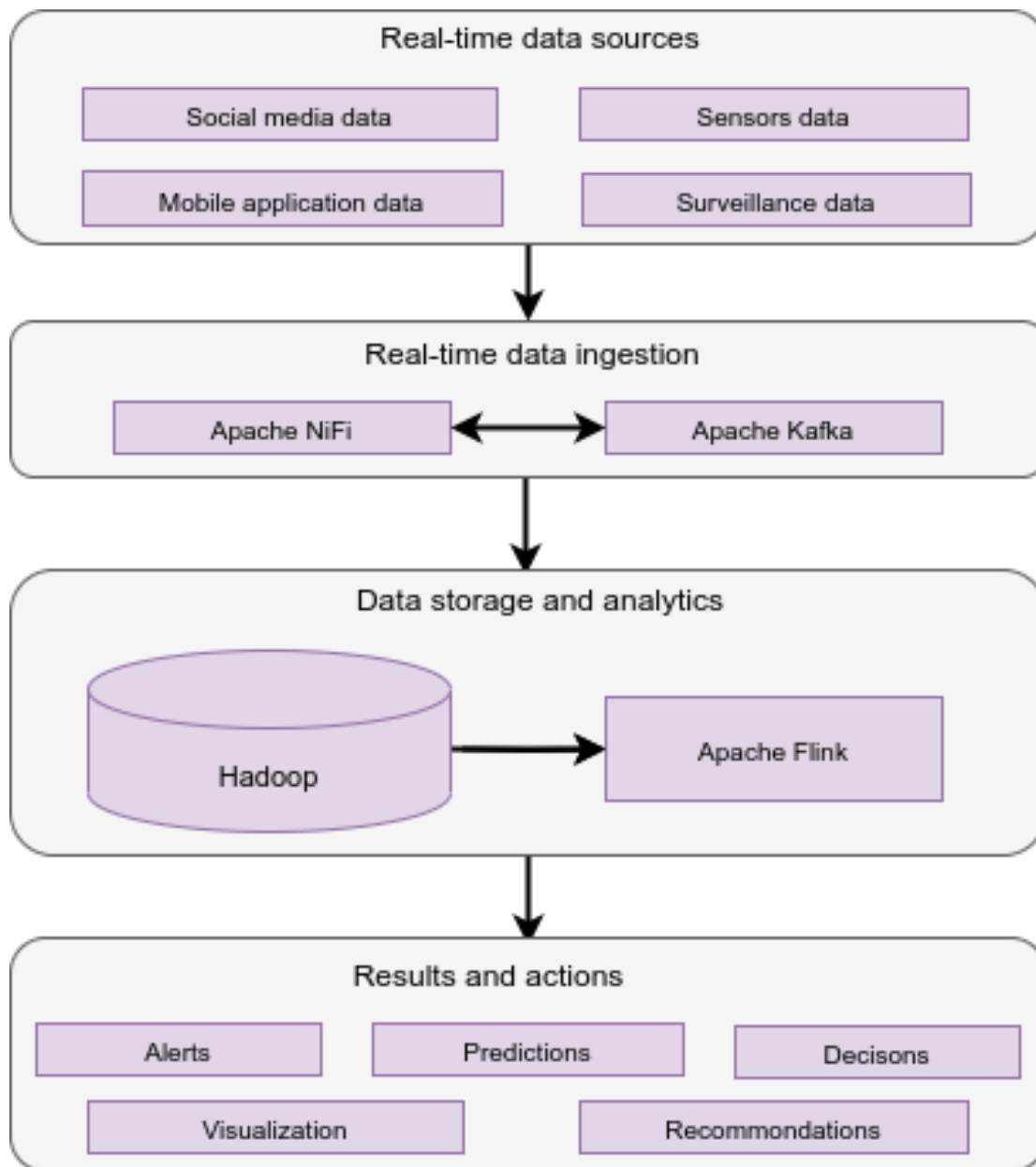


Figure 1.2: Real-time big data analytics for Smart city

In comparison with traditional big data applications, real-time big data analytics

has higher requirements in terms of data collection, analytical tools, security, and management. Many smart city applications are using the data generated within the city from different data sources. The data collected with this heterogeneity of data in real-time streaming is an important task in the success of the data-driven solutions for the smart city. In real-time analytics, the data get stored once but may read many times, so the data storage technologies used must be capable of data synchronization and sufficient to handle data at scale irrespective of time. In the applications where the user-generated data are considered for analytics, user participation is essential to get the expected result. It is challenging to generate and collect sufficient and suitable data for accurate predictions, recommendations, or desired data-driven decisions. Real-time big data analytics is a more challenging task as data from multiple sources in a smart city are to be considered and analyzed for specific solutions.

1.3 REAL-TIME ANALYTICS FOR PUBLIC SAFETY

Smart city projects embrace the concept of a ‘Safe City’ through smart policing solutions that provide safety and security within the city impact quality of life. Smart policing solutions are widely used for public safety due to the technological adoption of the Internet of Things (IoT) and Cloud, as stated by (Dhapte 2018). Transport and traffic security, infrastructure security, emergency services for fire and medical, crisis management, and law enforcement are the most common solutions in smart city public safety services. Real-time information is crucial for better implementation of such applications to provide timely services. Real-time crime centers are established in some cities to keep the cities safe by monitoring the activities in real-time within the city. Intelligent analytics on real-time data generated within the city is the solution for smarter crime responses, monitoring and prevention. Law enforcement agencies are switching towards predictive policing for their routine and investigation procedures. It involves advanced analytics techniques to predict what and where an incident likely to happen.

In recent days, the advancement in digitization opens up possible creation of user-generated content from various sources — analytics on all available data as input

to discover valuable data patterns results in finding accurate data-driven solutions. For example, public safety for smart policing applications is to collect user-generated content from different social networking applications and any specific smart application designed for the same purpose. Real-time analysis of the data collected from these data sources helps in early predictions and monitoring of crime-related incidents within the city. It is challenging for analysts to use multiple data sources with different properties in a specific data-driven solution.

In smart policing applications for public safety in smart cities, data analysts collect the data within the city from different sources for finding crime-related data patterns that are further used for crime detection and administrative decisions for crime prevention. In this scenario, many popular social media platforms used by the public and any specific applications offered by the police department are the major data sources of information. All these data generated within the city are analyzed for making better decisions or more accurate predictions for crime detection and prevention. The rapid growth of digitization in various fields created ample space for more and more new data sources, which are added regularly from the sources such as social media and smart applications. It is challenging for the data analysts to use additional data sources in their existing data-driven solutions with minimal cost and time.

1.4 MOTIVATION

The recent advancements in Information Communication Technology created scope for data-driven applications in all fields. To find intelligent solutions by using the appropriate data sources is becoming an important research topic for Industry and academia. The data to be considered for analysis is not only limited to archival or historical data but it is also proved to use real-time data for better performance and accuracy of the solution in most of the solutions. The rapid growth in the big data world supports different tools for processing historical data, streaming data, real-time data on a large scale. The data-driven models which use both historic and real-time data together are the most effective solution. The researchers provide many such solutions in different areas like healthcare, business analytics, environmental predictions, monitoring, etc.

Many developed and developing countries, including India, are focusing on smart city projects to offer better services for the people in the city. These smart city applications are integrated with Information and Communication Technology (ICT) and the Internet of Things (IoT), generates a vast amount of data. This data can be applied with proper analytics to find useful information that helps in better governance in the city. Public safety solutions with security for the people and infrastructure within the city are essential for the city to be safer and more sustainable. There are some smart policing concepts used in smart cities to predict crimes, terror attacks, location-based services, crime investigation using some data-driven decisions by statistical analytics. Designing data-driven solutions help city governance to overcome the crime rates in the city and faster investigation of crimes. These solutions make streets and homes safer by applying robust analysis to trusted information and linking that intelligence with law enforcement officers. It helps the city to reduce crime and threats.

Existing systems for crime monitoring and predictions are designed to collect historical criminal records from the police departments and generate the prediction. However, the data often collected yearly, which may be less effective over time in cities because of a large number of floating populations in urban areas. It is necessary to collect the data in real-time as a large amount of data in smart cities is accessible in a streaming manner. The rapid growth in big data analytics and streaming data analytics made it possible to analyze data immediately as and when it gets generated at the source. More and more technological adaptation in day-to-day life in the city generates a large amount of data continuously. Such data generated from the users and digital infrastructure can be analyzed instantly to derive useful data patterns. It is also essential to use all available data sources as input to the analytical system. To get a holistic view of the situation, as more and more data sources are used instead of targeting on a single source, it can produce more useful insights into the prediction system, which helps increase the performance of the system.

In this thesis, an attempt is made to propose a real-time big data analytics framework for public safety system in smart city applications that can use real-

time data from multiple sources for more accurate predictions in crime detection and monitoring.

1.5 THESIS CONTRIBUTIONS

A real-time big data analytics system is proposed for public safety solutions in smart city using multiple data sources.

1. A real-time big data analytics framework with data blending approach using multiple data sources for smart city applications is proposed. Analytics using multiple data sources for a specific data-driven solution helps find more data patterns, which increases the accuracy of analytics results [Publication-1].
2. Using the proposed real-time big data analytics framework, a real-time based emergency alert system to help the public safety solution is implemented using a machine learning-based classification algorithm. The experiment is carried out with different classification algorithms, and the results show that Naive Bayes classification performs with an accuracy of 73% which is better than the other algorithms used [Publication-2].
3. Using the proposed real-time big data analytics framework, a real-time crime prediction system is designed that incorporates a real-time data ingestion mechanism accompanied by a data blending approach for multiple data sources. The results show that real-time data, along with the historical data, attains better performance. It is also tested with different time intervals to update the prediction model. Naive Bayes classification performs with an accuracy of 81% which is better than the other classification methods used in the experiment [Publication-3].

1.6 THESIS ORGANIZATION

The rest of the thesis is organized as follows: Literature survey on various techniques and solutions with real-time big data analytics is discussed in Chapter 2. Chapter 3 discusses the proposed real-time big data analytics framework using multiple data

sources for public safety. Chapter 4 presents a real-time emergency event detection system for Public safety using multi-source data. Chapter 5 discusses real-time analytics based crime prediction using proposed framework using real-time data along with historical data. Finally, the summary of all the proposed techniques and the future research directions are given in Chapter 6.

CHAPTER 2

LITERATURE REVIEW

Big data has been a progressive aspect of the industry due to the data explosion, which occupied all business categories for a few years. The data-driven applications target competitive advantage with the help of real-time data warehouses and big data streaming. The combination of big data and real-time analytics is becoming most popular in data-driven applications. The academic and industry research produced many applications using real-time big data analytics in healthcare, fraud detection, smart grid, social media data, crime monitoring, sensor data analytics, etc. There is considerable work in real-time data analytics for crime detection and prevention. In this chapter, we survey existing work in real-time big data analytics for different applications. The chapter also provides a brief details on various solutions for crime detection and monitoring for the city safety and its limitations.

2.1 REAL-TIME BIG DATA ANALYTICS

The “big data” paradigm has been expanding rapidly in recent years. The term big data is used for datasets that are so large that they cannot be processed and managed using the traditional database tools and technologies. Real-time data, streaming data, and fast data are emerging in the big data world. Real-time big data analytics means big data is processed as it gets generated from a specific source to find valuable insights with minimal decision making time. The crucial part of real-time big data analytics is the

input data and response time frame. Input data to be collected as and when data gets generated using a real-time processing system to push or pull data. The most commonly used method is pulling flowing high volume data known as streaming data. However, the real-time processing system is not only focusing on ingesting the streaming data, it also focuses on pulling the data into the system when data is generated or available at the source. Real-time analytics and streaming analytics have become more prevalent in big data applications, where timely decisions are more crucial and beneficial. It is a need in many big data applications in recent days to generate results in real-time for better performance. The value of data in real-time analytics is indeed considered to be highest when it is fresh and analyzed as soon as it arrives in the system. Real-time analytics focuses on processing and analyzing data in near real-time or with minimal delay, enabling organizations to make timely and informed decisions based on up-to-date information. Hence, data are analyzed as and when they arrive in the system to find a result, whereas data are stored and then analyzed in batch processing.

Real-time big data analytics is becoming a high priority in many business applications where timely decisions are crucial. However, the challenging task in the real-time big data processing system is extracting, transforming, and loading (ETL) into the data warehouse compared to the traditional big data system. It is a big challenge to gather a massive amount of data that are heterogeneous in real-time. Due to continuous updates at the data warehouse, it is also a challenging task for data cleaning, query processing, and data transformation. Various tools and technologies have been developed so far to address these challenges in the real-time big data processing. The standard framework of the Map-Reduce model for big data is based on batch processing which does not support stream processing. However, it can partially handle the streaming data using a technique called micro-batching, which is not an efficient solution for stream processing, as stated by (Peng et al. 2012). In micro-batching, the stream is treated as a sequence of small-batch chunks of data. In small intervals of time, the incoming stream is created as a chunk of data and sent for processing in the batch system. A complete solution to this stream processing is achieved in streaming analytics frameworks like Apache Spark, Storm, and Flink;

detailed information was given by (Marcu et al. 2016), (Yang et al. 2013), and (Katsifodimos and Schelter 2016). In Spark, the data stream is represented as a sequence of Resilient Distributed Datasets (RDDs). The in-memory computing feature in Spark enables it to compute data batches quicker than Hadoop. Spark has good streaming support integrated, which supports the design of real-time predictive analytics services with fast and scalable streaming data processing. Apache Storm is another distributed computing framework for stream data processing, but there are limited streaming machine learning libraries available. Apache Flink is introduced as an alternative for Spark with its defining characteristics as real-time processing and low data latency. Spark processes chunks of data known as RDDs, whereas Flink can process rows after rows of data in real-time.

Big data has been a progressive aspect of the industry due to the data explosion, which occupied all business categories for a few years. The advance in this big data world is focusing on real-time data for better performance in data analytics. The academic and industry research produced many applications using real-time big data analytics in healthcare, fraud detection, smart grid, social media data, sensor data analytics, etc. Some of the research works done in real-time big data analytics are listed here.

2.1.1 Real time big data analytics in twitter data

Social media data such as Twitter data are the most commonly used data source in real-time analytics. Plenty of work has been done with Twitter data analytics for various applications. Using a rich context model for real-time big data analytics on Twitter is proposed by (Sotsenko et al. 2016). It is an approach for contextual big data analytics in social media analytics, particularly for Twitter data. A combination of the Rich Context Model (RCM) with machine learning is used to improve the performance of data mining techniques. The proposed architecture is for real-time contextual analysis of tweets that can be used in predictive analytics or relevant context-aware recommendation. Rich Context Model is used to find similar tweets and integrate this with a machine learning algorithm for clustering tweets based on contextual similarity. Similar work

was proposed by (Voskarides et al. 2014), where linking a tweet to an entity is described by context information. Both approaches are similar in terms of linking each tweet to an entity, but the first approach uses additional resources such as Web Service APIs and Open Data APIs to describe a rich context of the tweet, which helps in the improvement of recommendation or prediction accuracy. The various approaches used in twitter data analytics can be broadly categorized into sentiment and emotion analytics approaches, lexicon-based approaches, and hashtag recommendation approaches. The majority of approaches in different applications focus on using lexicon resources with whole tweet textual content (Voskarides et al. 2014), (Taboada et al. 2011). (Thelwall et al. 2012) used SentiWordNet (Bueno et al. 2013) and WordNet (Baccianella et al. 2010) to find polarity and performed rule-based classification on Twitter data. A lexicon-based approach is proposed by (Miller 1995) using co-occurrence patterns of words in different contexts to identify the sentiment orientation of words.

2.1.2 Real Time Big Data Analytics for Healthcare

Designing a medical emergency response system using real-time big data analytics is proposed by (Rathore et al. 2016). The system proposed here is to make an intelligent decision by analyzing medical data collected from sensors attached to a human body. Sensor data here are from different types of sensors used for collecting medical data like blood pressure, heartbeat, glucose level, temperature, and many more. Here, an intelligent building algorithm is implemented where the intelligent building is a smart block used for sorting, processing, and executing certain actions based on the data context. In this work, Apache Spark is used as a real-time processing tool on top of the Hadoop ecosystem. Here multiple data nodes are used to store the block of data. Each data node is equipped with the intelligent building algorithm designed in the proposed work. Real-time big data stream computing in Healthcare is discussed by (Ta et al. 2016). In this, Apache Kafka and Apache Storm are used as real-time analytics tools for healthcare data streams. This architecture supports healthcare data analytics by providing both batch and stream processing.

2.1.3 Real Time Big Data Analytics for alerts and monitoring System.

A real-time monitoring system for disaster management using social big data is proposed by (Choi and Bae 2015). This system uses social media data, particularly Twitter data, and analyses tweets in real-time for any disaster-related information. This system collects data through Twitter data stream crawling where tweets are written in Korean and analyses it for the disaster-related tweets using the procedure of Korean language processing. Then it displays disaster situations and trends on a map in real-time. (Wang et al. 2015) proposed an approach for road traffic monitoring by estimating online vacancies on the road using a traffic sensor data stream. Here only real-time data are considered for the analysis. This method uses a multiple linear regression approach to traffic vacancy estimation with the real-time stream processing tool Apache Storm.

Real Time Big Data analytics for predictions

Real-time big data analytics for predicting terrorist incidents is proposed by (Toure and Gangopadhyay 2016). In this system, the data from many news sources are collected automatically and predict the future incident by a proposed risk model. This risk model was developed based on different factors like incidents, time periods, and time factors. The model calculates the terrorism risk level of different locations. (Zhang and Yuan 2015) proposed a prediction method for air quality index levels using a random forest algorithm. This work used Apache Spark to implement the predictive model for air quality through analysis of real-time meteorology data from Beijing city. A distributed random forest algorithm is implemented on the basis of resilient distributed datasets and shared variables, and an air quality prediction model is built using parallelized random forest algorithm. Some of the work in real-time big data analytics are summarized in Table 2.1

Table 2.1: Summary of Literature:Real Time Big Data Analytics

Paper Details	Author, Published Year	Issues addressed
Towards Resilient and Smart Cities: A Real-Time Urban Analytical and Geo-Visual System for Social Media Streaming Data	(Yao and Wang 2020)	A real-time urban analytical and geo-visual system using social media streaming data to find undefined urban extreme events and early detect emergency events
Real-time event detection from the Twitter data stream using the TwitterNews+ Framework	(Hasan et al. 2018)	An event detection system that incorporates specialized inverted indices and an incremental clustering approach
A survey on data preprocessing for data stream mining: Current status and future directions	(Ramirez-Gallego et al. 2017)	feature and instance selection, and discretization
T-Hoarder: A framework to process Twitter data streams	(Congosto et al. 2017)	Framework that enables tweet crawling, data filtering, and summarization
Exploiting IoT and Big Data Analytics: Defining Smart Digital City using Real-Time Urban Data	(Rathore et al. 2018)	City data such as weather data collected and preprocessing, computing, and decision making
“Using a Rich Context Model for Real-Time Big Data Analytics in Twitter”	(Sotsenko et al. 2016)	Rich Context Model (RCM) with machine Learnig for twitter data

Real-time medical emergency response system: Exploiting IoT and big data for public Health	(Rathore et al. 2016)	Sensor data stream with medical information is analyzed for real time monitoring of patient
Big data stream computing in healthcare real-time analytics	(Ta et al. 2016)	Here both batch and stream processing used for monitoring healthcare data
The Real-Time Monitoring System of Social Big Data for Disaster Management	(Seonhwa and Byunggul 2015)	Social media data analyzed for disaster related information
Estimating online vacancies in real-time road traffic monitoring with traffic sensor data stream	(Wang et al. 2015)	Real time streaming of sensor data for traffic monitoring
Real time big data analytics for predicting terrorist incidents	(Toure and Gangopadhyay 2016)	Real time news sources are analyzed to predict the future incident
Fast fine-grained air quality index level prediction using random forest algorithm on cluster computing of spark	(Zhang and Yuan 2015)	Real time air quality index level monitoring by prediction

2.2 SMART CITY DATA AND PUBLIC SAFETY

Smart cities are the most popular urban development project in many countries due to increasing urbanization. Smart city ecosystem consists of various intelligent components such as smart devices, smart applications, and solutions to enhance the city operations, city resource management, and improve the services to make daily life easier. While 'smart city' means different things to different people, one common

2. Literature Review

thing everyone agrees on is that digital technologies are used in the smart city to improve the quality of the services within the city. According to a report by (ISO/IEC-JTC 2015) smart and sustainable city is an innovative city that uses ICT and other technologies to improve quality of life, the efficiency of urban operation and services, and competitiveness while ensuring that it meets the needs of the present and future generations concerning economic, social, and environmental aspects. However, the most commonly adopted smart city framework describes the smart city in six dimensions that are (i) smart economy, (ii) smart mobility, (iii) smart environment, (iv) smart people, (v) smart living, and (vi) smart governance (Giffinger and Gudrun 2010).

The technological growth in digitization and advancement in communication technologies made smart cities incorporate it for better city administration. Internet of Things (IoT) and Information Technology (IT) are the most common part of any smart application in smart cities. It leads to generating a large amount of data in different formats. The advancement in big data technologies exploits to analyze the data generated within the city for enhanced services in the smart city. The smart applications used in smart cities, such as smart traffic, smart environment, smart governance, smart agriculture, and smart health care, generate a massive amount of data that can be used to extract valuable insights to improve the quality of service. The different types of video surveillance cameras, sensors, and smart mobile applications used by smart city applications are the primary source for generating data. The smart applications designed for smart city services and popular social media applications used by the people cause huge amounts of user-generated content generated within the city that helps to enhance the quality of service within the city. The research studies by (Nuaimi et al. 2015), (Leonidas 2017) claim that the data produced within the city by its various components are the most critical asset for smart city deployment.

The data generated on a large scale from various smart city domains consists of structured, semi-structured, and unstructured data, commonly known as Big Data (Lau et al. 2019). Hence Big data analytics (BDA) to extract valuable insights from the raw data generated within the city helps in decision-making for city administration.

Data-driven decisions (DDD) using Big data is the most common trend in various domains, including smart city applications. Various tools and techniques are used to perform big data analytics in smart cities. It includes different tools and techniques for data modeling, data storage, data transfer, data management, and data processing and analysis. Along with the traditional statistical methods, modern methods such as Artificial Intelligence (AI), Data Mining, Machine Learning (ML), and Deep Learning (DL) algorithms are the most popular in making data-driven decisions. The key challenges associated with big data analytics in a smart city are data integration, data privacy, and filling the skills gap. Data gets generated at various organizations, a variety of data sources and intelligent devices, and diverse environments. Integration of these data within the city is one of the key challenges for analysts due to various organizational and political barriers. It is challenging to shape the new data environment using organizational and personal data by addressing data privacy.

Smart city initiatives always aim to provide better infrastructure within the city, such as smart lighting systems, smart traffic, smart parking management, smart waste management, etc. In later stages, it is found that infrastructure can be built up more efficiently so that it can also be used in providing security to the residents. For example, the smart lighting system installed within the city can also be used in the public safety system. Smart policing systems are widely used for public safety due to the technological adoption of the Internet of Things and Cloud. Transport and traffic safety, infrastructure safety, emergency services for medical and fire, crisis monitoring and management, and law enforcement are the primary solutions in any smart city public safety services. Real-time data is crucial for better implementation of public safety applications to provide better and timely services. Real-time crime centers are most common in some cities to keep the cities safe by monitoring the activities in real-time within the city. Intelligent analytics on real-time data generated within the city is the solution for crime monitoring, responses, and prevention. The city administration and Law enforcement agencies are switching towards predictive policing for their routine and investigation procedures. It involves advanced analytics techniques to predict what and where an incident is likely to happen.

Predictive policing in real-time can help in early monitoring of the crime and preventing it before it happens. It is crucial to predict the possibility of crime within the city for better management of the security system. The analytics on the dataset consisting of past criminal records, case history, and real-time data helps in the early predictions of crime patterns to prevent crime in the city. Analytics on social media applications in real-time helps in predicting the possibilities of crimes as well as policing actions to be taken in the city. Identifying the crime pattern in the messages, tweets, and complaint data along with the geographical location on social media and performing sentiment analysis helps in detecting the crime zone in real-time. In this process, it is also challenging to use historical data along with real-time data for better predictions. It is proposed to use some applications and devices to generate data for this purpose other than social media information for better management of public safety. Creating a dedicated application for the police department to collect real-time information from the public can indeed be a beneficial approach in enhancing communication and gathering relevant data. Such applications can facilitate efficient reporting of incidents, enable faster response times, and promote community involvement in maintaining law and order. The data generated from such applications are analyzed for detection and prediction of the crime and making real-time decisions to act upon that. Real-time analytics can be used to investigate cases to avoid delays in this process.

Data Preprocessing in Big Data Analytics:

Data preprocessing is a crucial and significant phase of the data analytics process (Zhang et al. 2003). The raw data used as an input into the analytics system is likely to be noisy, inconsistent, and imperfect. The data preprocessing phase is the set of techniques used for making raw data as analytics-ready in the data analytics process as stated by (Garcia et al. 2014). The preprocessing phase in real-time data analytics becomes challenging, where the raw data is continuously entered into the data collection system. The critical part of data preprocessing includes mainly two concepts, namely data cleaning and feature engineering.

Data preprocessing is a crucial stage in data analysis for achieving better accuracy and performance in the analytical model. Most of the effort made in the preprocessing of big data mainly focuses on developing feature selection methods (Garcia et al. 2016). Noise reduction, instance reduction, and missing value imputations are the major preprocessing methods focused on by data analysts. When the data is collected from various sources, combining this to form consistent data is essential in making the data ready for analysis. Data blending is a technique in preprocessing for combining data from multiple sources to create a common data set for decision-making (Wessler 2015). It is one of the quick methods to extract common information from multiple data sources.

2.2.1 Real time big data analytics for public safety in the city

Forecasting crimes using autoregressive models

(Cesario et al. 2016) proposed an approach based on autoregressive models for reliably forecasting crime trends in urban areas. The main work here is to design a predictive model to forecast the number of crimes that will happen in the city of Chicago. The methodology proposed in this work is able to predict the number of crimes with an accuracy of 84% one year ahead and 80% two years ahead of the forecast, which is proved with an experimental evaluation. Autoregression is the regression of a variable against itself. In this model, the variable of interest is forecasted using a linear combination of its past values, while the moving average model uses past forecast errors. In this work, only historical data in the city were used.

Location aware Mobile crime information framework for fast tracking response to accidents and crimes in big cities.

(Mantoro et al. 2014) proposed a framework in which the mobile app can send and receive the location of crimes, including the images to the nearest/central police station. Here the system uses a built-in database with a combination of Google Maps APIs. The system allows the police to find the location of the accident right away and increase the safety of the resident in big cities. This Mobile Crime Assistance

Architecture instantly provides mobile phone users position information when there is a crime against themselves or others. This framework is built specifically on an android system to read the crime location based on location-based service. Android mobile client will keep the record of the individual user, and it will sync with the mobile server, which connects to a crime database repository. On the user side, a menu button that transmits a signal to the central police station was added to the application. The user can use only one crime location information which can be directly sent to the central police station. At the central station, the location of the complaint can be seen based on the coordinates of the location that is sent over the GSM network to the central police station to analyze the position of the victim or complaint.

CityPulse: Large Scale data analytics framework for Smart cities

(Puiu et al. 2016) designed a CityPulse project, a big data analytics framework for smart city applications. A general framework is designed which can be used for a distributed, large-scale approach for semantic discovery, data analytics, and reasoning of large-scale real-time IoT and social media data streams for knowledge extraction in a city. Here large-scale data stream processing modules are configured to process real-time parking and traffic data coming from the city sensors with the scope of detecting relevant events for the users traveling in the city. Here, data wrappers for both parking and traffic data streams are implemented for fetching data. The data streams related to parking provide parking spaces in the garage and the number of occupied spaces/vehicles in the garage, and the traffic data stream with the number of vehicles passing two points and their average speed. Both data wrappers are deployed in resource management. During runtime, new observations for the traffic stream are fetched in a five-minute interval and the parking stream in a one-minute interval. This work greatly contributes toward integrating heterogeneous data streams and real-time data analytics in a scalable framework.

Big Data based smart city platform- Real Time crime analysis

(Ghosh et al. 2016) proposed a safer city concept by enabling crime and risk analysis

of unstructured crime reports, criminal history, suspects, auto license data, location-specific data, etc. This work includes an intelligent solution for data based on a smart city platform in Newark, NJ. The solution is based on a machine learning approach to automate and help crime analysts to identify the insights that can be used for better decision-making and optimized actions. Machine learning for automatic incident classification is the key concept used in this work. This is achieved by various concepts like Document indexing, latent semantic analysis, Text categorization, etc. In document indexing, more often, a term occurs in a document, representing its content. The latent semantic analysis considers documents that have many words common to be semantically close and those with few words common to be semantically distant. R-Text tools, an integrated interface that provides a comprehensive approach to text classification, are used in Text categorization. This work is an attempt at a safe city initiative where a solution was proposed for the auto-theft report at a particular time in the city and an incident report of a shooting near specific location.

A spectral analysis of crimes in San Francisco

(Venturini and Baralis 2016) attempted an exploratory analysis of Spatio-temporal patterns of crimes using data from San Francisco. The spectral analysis applied to the temporal evolution of all crime categories, finding that many have a weekly or monthly periodicity. Similar work is stated by (Parvez et al. 2016), in which a novel approach is designed to identify the Spatio-temporal crime pattern in Dhaka city. Both of these work exploits the historical crime data of a particular city to predict the possible crime incidents in a particular region at a specific time. The model captures both the space and time proximity of past crimes while predicting future crimes.

Crime prediction and monitoring based on spatial analysis

(ToppiReddy et al. 2018) proposed a framework for crime prediction and monitoring based on spatial analysis. Various visualization techniques and machine learning algorithms are used to predict the crime distribution over an area. This helps the crime analyst to analyze the crime networks by means of various interactive visualization.

2. Literature Review

The interactive and visual feature applications will help report and discover the crime patterns. (Catlett et al. 2019) proposed a Spatio-temporal crime forecasting model to detect high-risk crime regions using an auto-regressive model. There are many applications and research has been progress over the years for finding different solutions for crisis monitoring and public safety systems. Some of the work in crime data analytics using city data are summarized in Table 2.2

Table 2.2: Summary of Literature:Real time big data analytics for public safety

Paper Details	Author, Published Year	Issues addressed
Spatio-temporal crime predictions in smart cities: A data-driven approach and experiments.	(Catlett et al. 2019)	Crime forecasting with an auto-regressive model to detect high risk crime regions
Crime prediction and monitoring framework based on spatial analysis	(ToppiReddy et al. 2018)	Machine learning algorithms are adopted for predicting the crime distribution over an area.
Towards Real-Time Road Traffic Analytics using Telco Big Data	(Costa et al. 2017)	Road traffic analytic and prediction system to provide micro-level traffic modeling and prediction;
IoV distributed architecture for real-time traffic data analytics	(Nahri et al. 2018)	Real time collecting and processing events generated by intelligent vehicles, visualizing traffic on each road section.
Forecasting crimes using autoregressive models	(Cesario et al. 2016)	Predictive model to forecast number of crimes in Chicago city

Location aware Mobile crime information framework for fast tracking response to accidents and crimes in big cities	(Mantoro et al. 2014)	Using mobile application, location information and along with images is collected at station for quick response
Large Scale data analytics framework for Smart cities	(Puiu et al. 2016)	Smart city large scale real time IoT and social data analytics framework for traffic and parking monitoring
Big Data based smart city platform- Real Time crime analysis	(Ghosh et al. 2016)	Using past data and real time data an intelligent solution proposed for auto theft reporting
A spectral analysis of crimes in San Francisco	(Venturini and Baralis 2016)	Exploratory analysis of crime data to predict possibility of crimes in the city

2.2.2 Data driven solutions using multiple data sources:

With increasingly digital applications and systems being adapted by organizations and people, the data gets generated at various sources. The user's interest and comfort made them choose different applications and services for the same purpose. This creates scope for data to get generated at different sources rather than restricted to a single source. The data-driven applications designed using input data from such data sources must aim to collect data from all sources to design a better system. The data-driven decisions made by security and law enforcement agencies must focus on multi-source data rather than a specific data source for the analysis. In recent days, many researchers have been aiming to design data-driven solutions in different domains with multi-source data analysis. The authors integrated features of news events, public sentiment, and quantitative indices into a tensor-based learning framework to improve performance (Li et al. 2016). (Baboshkin and Uandykova 2021) designed a multi-source data analytics

method for future price fluctuation prediction using market commentary, review, and news data.

Data Preprocessing in Analytics:

Over the past few years, the research articles on streaming data analytics have highlighted the need for the preprocessing mechanism of the data collected in the streaming manner for the analytics (Ramirez-Gallego et al. 2017). As mentioned by (Zhang et al. 2003) and (Garcia et al. 2016), data preprocessing is an essential and major phase of the data analytics process. The raw data input into the analytics system will likely be noisy, inconsistent, and imperfect. The set of techniques to be used for raw data as analytics-ready is the data preprocessing phase in the data analytics process, as explained by (Garcia et al. 2014). The preprocessing phase in real-time data analytics becomes challenging when the raw data is continuously entered into the data collection system. Most of the effort made in preprocessing of big data is mainly focused on developing feature selection methods mentioned by (Garcia et al. 2014). Noise reduction, instance reduction, and missing value imputations are the other important preprocessing methods focused on by data analysts.

Data blending is a technique in data preprocessing stage for combining data from multiple sources to create a common data set for decision-making, as stated by (Wessler 2015). It is one of the quicker methods to extract common information from multiple data sources. (Pina-Garcia and Ramirez-Ramirez 2019) proposed that data generated from different social media platforms can be integrated to enhance big data-driven models for crime prediction. Harnessing multi-source data about public sentiments and activities for informed design is proposed by (You et al. 2019) that addresses the process from data collection to data visualization. (Xu et al. 2019) proposed a framework for collecting and analyzing data from social media and surveillance cameras to describe public safety events.

2.3 RESEARCH GAPS

In the previous section, we have discussed various methods in different applications using real-time big data analytics and also have given brief overview of the role of real-time analytics in public safety system in a smart city. Based on the literature study, following research gaps have been identified.

- Most of the real-time big data analytics approaches proposed in the literature use the data from a single source collected in a real-time or streaming manner. This increases the scope for using the data from multiple sources in real-time for analytics to find more efficient insights.
- The preprocessing methods for big data analytics in real-time are more complex, where data are collected from more than one source in real-time. Enhanced preprocessing models and algorithms are required to handle data from multiple data sources.
- Some applications can produce more accurate results, predictions, and decisions when data is collected from more than one data source. It increases the opportunity for researchers to enhance predictive analytics models and decision algorithms by treating all appropriate data sources at the same time.
- Real-time big data analytics is used for very few categories of applications like health care, Twitter analytics, and financial data. It gives an opportunity for designing real-time solutions by implementing appropriate algorithms and models useful for data-driven solutions for other areas.

2.4 PROBLEM STATEMENT:

Design of a data driven solution using real time big data analytics for public safety in smart city through crime prediction and monitoring.

2.5 PROBLEM DESCRIPTION:

With the ever-increasing use of data-driven applications in day-to-day activities, various sources generate a massive volume of data. Data-driven solutions offered in the security domains such as public safety systems need to analyze the data for valuable insights as and when data get generated at the source. Real-time analytics helps data analysts to glean essential insights quickly and find data-driven solutions instantly. The challenging task in real-time big data analytics is collecting the data and extracting valuable information from it as and when it generates at the source. The increase in social media platforms and mobile applications generates valuable data from multiple sources. The public safety system must use data from all possible data sources to design an effective security system. The aim of the public safety and security system in smart city applications is early detection and monitoring the crimes. Hence we need a better system for real-time analytics of multiple data sources for early detection and monitoring of crimes.

2.6 OBJECTIVES

- Design an approach for real time pre-processing of big data collected from multiple sources in smart city applications.
- Design a new model for data analysis on pre-processed data for finding valuable insights for crime pattern detection in real time.
- Develop an intelligent methodology to exploit past data along with real time data to predict the crime hotspots which helps in smart policing for monitoring the crime in the city.

2.7 SUMMARY

This chapter has provided a survey of different applications and methods designed in real-time big data analytics for various domains. Various sources generate data in real-time in a smart city; it is more beneficial to the city administration to design applications using real-time analytics for better services. We have also discussed some

of the work done for the public safety system in monitoring crime incidents using real-time analytics. This discussion leads to the open issues and research gaps, and challenges in real-time big data analytics for public safety systems.

In the following chapters discusses the proposed framework for real-time big data analytics using multiple data sources. The data blending mechanism in the proposed framework is more beneficial in using all possible data generated from various sources that can be used in the analytics process, leading to better results. The advancement in digitization over the years made multiple applications lead to data getting generated from multiple sources. The proposed framework is also evaluated by designing an emergency event detection system and a crime prediction model. Both historical data and real-time data are used in the crime prediction model, which is important to achieve better results.

CHAPTER 3

REAL-TIME BIG DATA ANALYTICS FRAMEWORK

Real-time analytics for finding valuable insights at the right time using smart city data is crucial in making appropriate decisions for city administration. It is essential to use multiple data sources as input data for the analysis to achieve better and more accurate data-driven solutions. It helps in finding more accurate solutions and making appropriate decisions. As discussed in Chapter 2, most of the real-time analytics systems proposed for different applications use a specific data source. This chapter proposes a real-time big data analytics framework for the public safety system using multiple data sources in the smart city.

3.1 INTRODUCTION

Real-time analytics and streaming analytics have become more prevalent in big data applications, in which timely decisions are more crucial and beneficial. It is a need in many big data applications to generate results in real-time for better performance. In Real-time analytics, data processed at the very moment it arrives into the system rather than processing at a later stage from data storage wherein it gets stored. Some applications generate data continuously in real-time, which affects the outcome of the analytical results. For example, the applications such as environmental monitoring need to collect real-time data such as temperature, humidity readings continuously. Real-time analytics helps the analysts to glean essential insights quickly and find the data-driven

solution instantly. The critical part of real-time big data analytics is extracting valuable information from the incoming data as and when data enters into an analytics system. The predictions or decision-making in these applications are affected by both historical data stored and real-time data generated continuously. A real-time analytics system must be capable of managing and analyzing the data as and when it enters the database.

Real-time Big data analytics is an iterative process. A good real-time processing architecture in a big data environment needs to be fault-tolerant and scalable. It must support both batch processing and real-time processing. Most of the real-time processing applications are implemented using two popular approaches provided by Lambda and Kappa architecture. It is important to accurately evaluate which architecture best suits a specific use case to implement a data analytics solution.

3.1.1 Lambda Architecture

Nathan Marz, the creator of Apache Storm, came up with this architecture. It is a data processing architecture designed for handling both batch and stream methods as proposed by (Marz and Warren 2015). This architecture has proven to be relevant to many use cases and used by a lot of real-time applications. Figure 3.1 shows the different processes involved in three-layer lambda architecture for real-time big data analytics. The architecture describes the system as a batch processing layer, a stream layer or real-time processing layer, and the service layer. Here the batch processing layer manages the historical data and processes a substantial quantity of data. It can fix any errors by re-computing the complete data set to update the existing views. The real-time processing layer processes the data streams in real-time. It will produce a more updated view for batch layer view using the most recent data. The service layer collects the outputs from the batch layer and real-time layer, and it is used for processing the final queries or solutions.

3.1.2 Kappa Architecture

This is a simplification of Lambda architecture but not as a replacement proposed by (Kreps 2014). In this batch processing system is removed, and the data is fed through

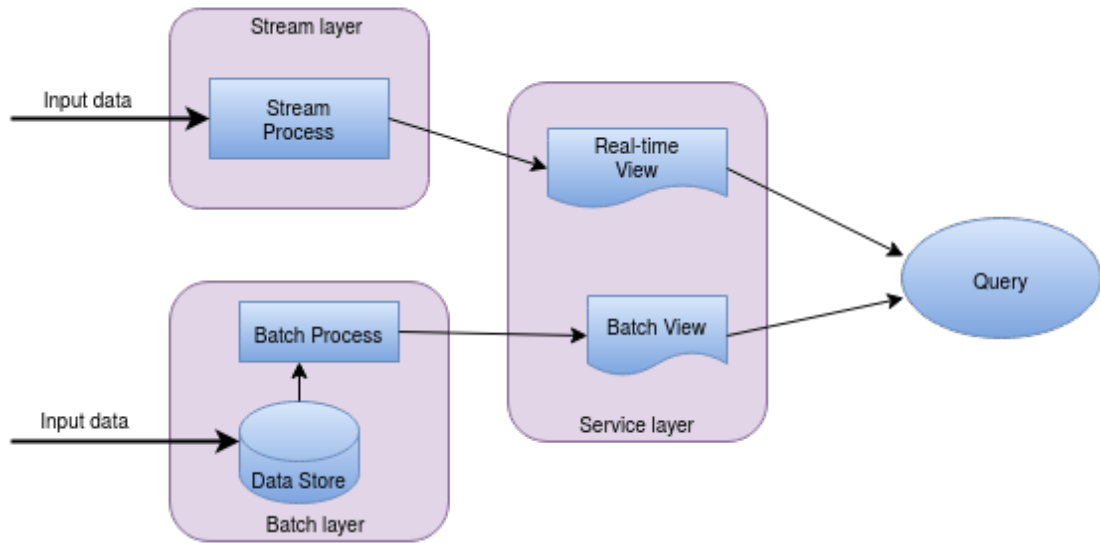


Figure 3.1: Lambda Architecture

the streaming layer quickly. Figure 3.2 shows the flow of Kappa architecture. In this, all data move to the service layer stream. The architecture is composed of mainly two layers, stream layer, and service layer. The stream processing layer runs stream jobs for real-time data processing. The service layer has a similar purpose as in Lambda architecture, where it needs to consider the real-time view from the streaming layer.

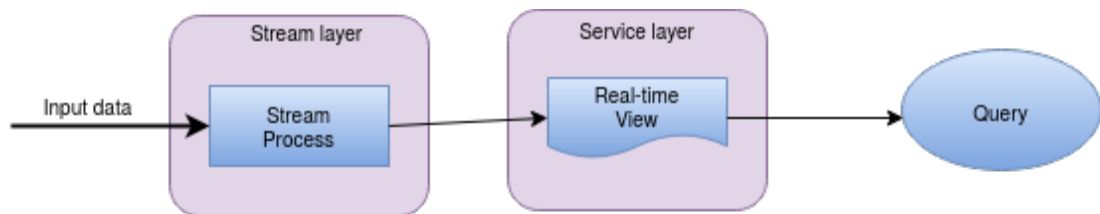


Figure 3.2: Kappa Architecture

In the proposed work, the real-time data from multiple sources are analyzed to discover valuable insights for making real-time decisions and predictions. The data processed at the moment is stored for further use in predictive models in later stages. The proposed framework is designed based on Lambda architecture. The data is ingested into the analytical system immediately after generating at the particular source and preprocesses to make it ready for further analytics. The data from identified sources streamed through the real-time layer, which processes the data and then passes it into the serving layer. The real-time queries are executed using the real-time views of the serving layer. The data stored for further use in a data store is executed using the

batch views along with the real-time views in the data-driven models.

3.2 PROPOSED DESIGN

The proposed design for real-time analytics using multiple data sources is as shown in Figure 3.3. The real-time data from identified data sources are collected and processed for a specific data-driven solution. When the input data required for the analysis are identified at different sources, it is essential to use all available data during analysis to increase the accuracy or performance of the data-driven solution. Here raw data from multiple sources in real-time are used as input data for the specific data-driven solution. The data is ingested from a particular source as soon as it is generated at the source. The data ingestion phase consists of different data ingestion processors for each input data source used. Each data ingestion processor is comprised of a real-time data ingestion mechanism for the specific data source. The data ingestion processor is configured with an initial preprocessing mechanism for filtering data of interest for the desired analytical solution.

The data ingested and filtered at each source is passed through a data blending mechanism. The purpose of the data blending mechanism is to integrate the data from different sources into a single common dataset for further analysis. The data blending phase consists of separate adapters for each source, which reads the input from respective data ingestion processors. Each adapter is a real-time task that can read the data immediately when filtered out from the respective processor. Data blending is performed to extract the common data of interest from each source and append it to a single dataset. The blended data is used in the next stage for analysis to find meaningful patterns in real-time. The blended data results in making data-driven decisions such as emergency alerts of crime incidents, identifying crime hotspots, and predicting possible occurrences of crimes in the city.

The different phases in the entire process include real-time data ingestion, data preprocessing, real-time analytics, and visualization results. A real-time data processing system must be a powerful computing system along with quicker response without any delay. Each phase in the process, starting from data ingestion to data

visualization, must perform the task immediately once the input data is received and send the output data to the next phase. The working of different stages in the proposed design is explained in detail in following sections.

3.2.1 Data ingestion

The more and more digitization in day-to-day life creates ample scope for gathering data from different sources. The aim of the proposed design is to use multiple data sources in collecting real-time information for the desired analytical solution. It is essential to identify the different sources where related data for the desired solution is get generated. The data format and structure may be different from source to source. In the proposed design, the data collection mechanism includes different data ingestion processors which ingest the data in real-time to the analytics system. The data collection mechanism from each of the identified data sources uses a separate data ingestion processor. Each data ingestion processor is comprised of the mechanism of ingesting the data from respective sources as and when data gets generated and responsible for the initial stage of preprocessing by filtering only the crime-related data. The data out from the ingestion mechanism is passed into the next stage of preprocessing.

3.2.2 Data preprocessing

The real-world data collected is incomplete, inconsistent, noisy, and needs to be cleaned before used for analytics. Data preprocessing is the initial stage in data analytics for making sure that data is ready to be analyzed. Analytical results depend on the quality of input data used. The majority of the analytical process comprises preprocessing the data. In the proposed design, we used Apache Flink jobs for preprocessing the data in real-time. Data preprocessing involves different operations such as removing or adding, or enhancing attributes from input data, filtering data to discard unwanted data, combining multiple data, or splitting input data. Flink jobs written for preprocessing mechanisms are responsible for doing the operations immediately once new data ingested into the system. These preprocessing tasks clean the data and store it on Hadoop as analytics-ready data.

3. Real-time Big data Analytics framework

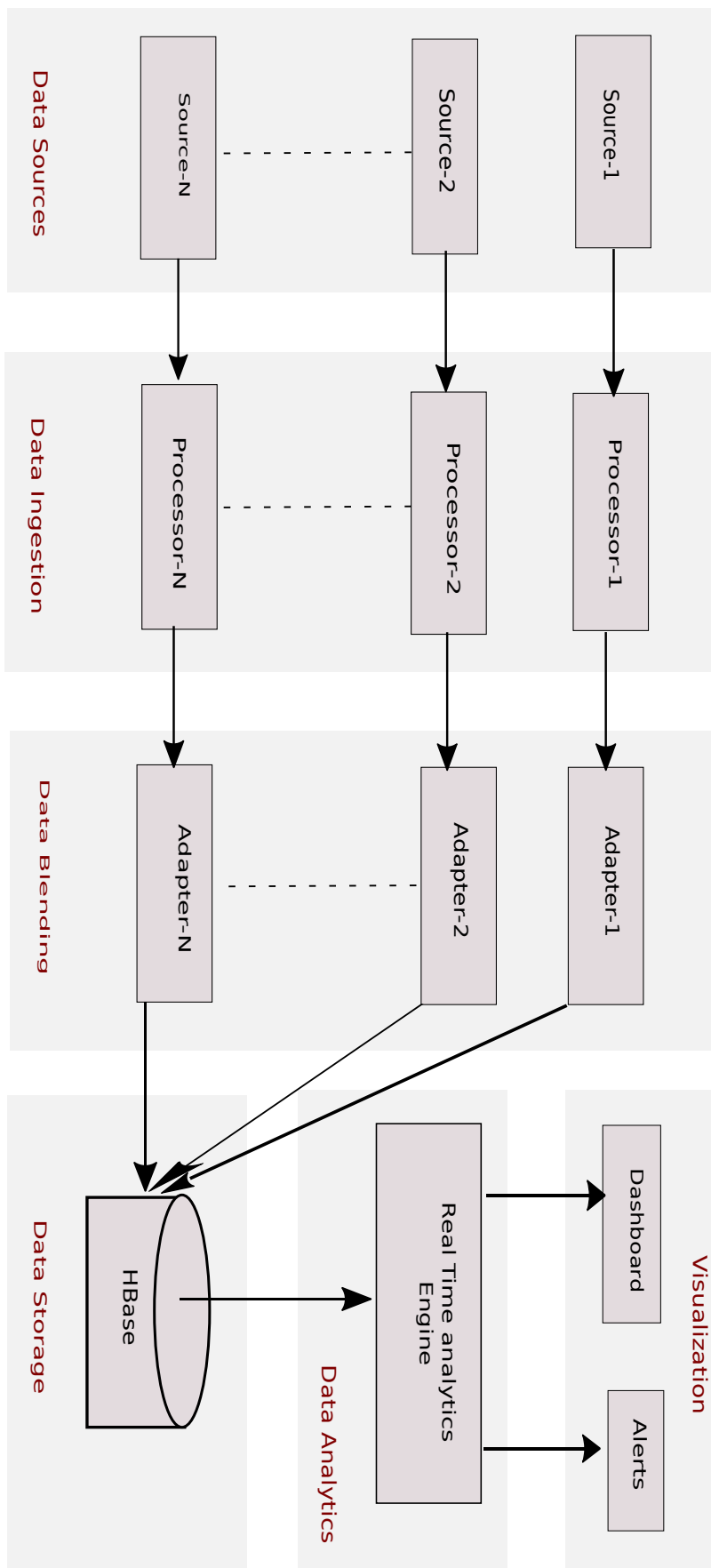


Figure 3.3: Proposed design for real-time big data analytics

The data streamed from each of the data ingestion processors is processed through separate data blending adapters. The data blending adapters comprise a mechanism to integrate the required information from the different sources as a single common dataset for the desired analysis in further stages. The main goal of the proposed framework is to use multi-source data to accomplish the data blending mechanism used in the data preprocessing stage. Each adapter is a real-time process that reads the data stream immediately after receiving the respective data ingestion processor. The adapter's main functionality is to preprocess the data stream ingested from the corresponding data source, where required information is extracted and updated into common data storage. The different adapters used for the different sources assimilate the common information from the input data stream and update it on a single common dataset used for further analysis.

3.2.3 Data storage

In the proposed design, Apache Hadoop is used for data storage. The preprocessed data from the data blending adapter is saved into the data store. In the proposed mechanism, real-time analytics need to access the data from the storage in real-time. Real-time analytics tools such as Apache Flink do not involve any storage mechanism but support reading and writing data from a different storage system. Apache Hadoop is a superior choice of data storage at a low cost. The blended data from the multiple data sources are updated on the HBase table on top of the Hadoop storage. Here HBase supports real-time read and writes access to the data. The real-time data is used along with the historical data for the analytics. The data used in real-time is further used as historical data in future analysis. Hence the data is continuously updated in the Hadoop storage and accessed for further analytics process.

3.2.4 Real-time analytics and Visualization

The preprocessed data is analyzed further to discover valuable insights. The data ready to be analyzed is passed into a real-time analytics engine for finding data-driven solutions such as predicting crimes and decision making. Apache Flink is used as a

real-time analytics engine in the proposed design. Flink can process the data in real-time to build the analytical model for the required data-driven solution. Here data is analyzed for real-time predictions for crime monitoring and making appropriate decisions to prevent the crime to assist the police authorities. Data visualization represents analytical results in the dashboard is to understand by the police authorities to make decisions for further actions. The crime report, crime statistics, crime hotspot predictions are visualized in the graphical representation. It could encompass alerts, emergency notifications, etc.

The blended data updated on the datastore is further analyzed to find valuable insights. The real-time data, along with the past data, are used to create an analytical model for real-time crime prediction. Apache Flink is efficient in building real-time analytical models for using real-time data. The real-time analytical process is designed for real-time crime monitoring and making real-time decisions for preventing crimes. In the proposed work, real-time analytics process is designed for generating an emergency alert system and a crime prediction model using machine learning. The process is updated continuously using real-time data and historical data stored.

3.3 TOOLS AND EVALUATION METRICS

Since the subsequent chapters include experimental evaluation and results, we would like to introduce different tools and technologies used in the experimental work and metrics used to evaluate the performance of the proposed framework. We considered Apache NiFi for managing data flow in the data ingestion mechanism. Apache Hadoop is used for data storage, and Apache Flink is used as a real-time analytics tool.

Apache NiFi

Apache NiFi is an open-source data integration and distribution framework that enables the automation of data flow between systems and services. It was developed by the National Security Agency (NSA) and is now a project of the Apache Software Foundation. NiFi provides a user-friendly interface to create, configure, and monitor data flows, called "data pipelines," using a drag-and-drop interface. These data pipelines

can be used to move, manipulate, and process data in real-time across a variety of data sources and destinations. Some of the key features of NiFi include:

- **Flow-based programming:** NiFi's data pipelines are built using a flow-based programming model, where data is represented as "flows" that move through the pipeline. This makes it easy to create, modify, and scale data pipelines as needed.
- **Built-in processors:** NiFi includes a variety of built-in processors that can be used to perform common data integration tasks, such as routing, filtering, transforming, and aggregating data.
- **Extensibility:** NiFi can be easily extended through the development of custom processors and plugins. This makes it possible to integrate with new data sources and destinations, as well as add custom functionality to data pipelines.
- **Security:** NiFi provides strong security features, including encryption, access control, and auditing. This makes it suitable for use in enterprise environments where data security is a top priority.
- **Monitoring and reporting:** NiFi provides real-time monitoring and reporting of data flows, making it easy to identify and address issues as they arise

Apache NiFi is a powerful data integration and distribution framework that provides a user-friendly interface, strong security features, and extensibility. Its flow-based programming model, built-in processors, and real-time monitoring make it a popular choice for managing data flows in a variety of use cases. NiFi has a wide range of use cases, including:

- **Data ingestion:** NiFi can be used to ingest data from a variety of sources, such as sensors, log files, databases, and APIs.
- **Data processing:** NiFi can be used to process and transform data in real-time, such as converting data formats, enriching data, and performing calculations.
- **Data distribution:** NiFi can be used to distribute data to multiple destinations, such as databases, data warehouses, and other systems.

- IoT data management: NiFi can be used to manage and process data from IoT devices, such as sensors and actuators.

Apache Hadoop

Apache NiFi and Hadoop are two complementary technologies in the big data ecosystem that can be used together to build robust data processing pipelines. When used together, they can provide a powerful solution for big data processing and analysis, enabling real-time monitoring and analysis of systems. Hadoop is an open-source distributed computing framework that provides a scalable and fault-tolerant way to store and process large volumes of data. It consists of several modules, including the Hadoop Distributed File System (HDFS) for storage and Hadoop MapReduce for processing. Hadoop has a rich ecosystem of tools and technologies that work together with Hadoop, such as Hive, Pig, Spark, and HBase, among others.

HBase is an open-source, distributed, column-oriented NoSQL database that is built on top of Apache Hadoop. It was developed by the Apache Software Foundation and is designed to provide a scalable and fault-tolerant way of storing and processing large volumes of structured data. HBase is optimized for real-time data processing and can handle high write throughput. NiFi can be used for ingesting data from various sources, processing, and transforming data, and routing data to various destinations. HBase provides a distributed, scalable, and fault-tolerant way to store and retrieve data. When used together, NiFi and HBase can provide a powerful solution for big data processing and storage. NiFi can be used to ingest data from various sources, such as social media platforms, sensors, and log files, and transform the data as needed. NiFi can then route the data to HBase for storage and retrieval. This allows for real-time data processing and analysis and enables the creation of data-driven applications and systems.

Apache Flink

Apache Flink is an open-source, distributed stream processing framework that is designed to perform real-time analytics on large, fast-moving data streams. It was developed by the Apache Software Foundation and is built on top of the Hadoop ecosystem.

Flink provides a powerful, flexible, and fault-tolerant platform for processing streaming data in real-time. It supports both batch processing and stream processing, allowing developers to easily write data processing pipelines that can handle both real-time and batch workloads. Flink also supports a wide variety of data sources, including Apache NiFi, Apache Kafka, HDFS, and Amazon S3.

One of the key features of Flink is its support for stateful stream processing. This means that Flink can maintain and update state information across multiple events in a stream, which is critical for many real-time use cases. Flink also supports advanced windowing semantics, allowing users to define and operate on time-based windows of data in a stream. In addition to its powerful streaming capabilities, Flink also provides support for machine learning and graph processing through its FlinkML and Gelly libraries, respectively. This makes it a versatile and comprehensive tool for data processing, analysis, and machine learning. Apache Flink is a powerful and flexible open-source stream processing framework that provides support for both real-time and batch processing, stateful stream processing, advanced windowing semantics, and machine learning and graph processing libraries. Flink is well-suited for a wide variety of real-time use cases, such as fraud detection, network monitoring, and IoT data processing, and provides a powerful platform for building data processing applications at scale.

Evaluation Metrics:

In machine learning, accuracy is a metric used to evaluate the performance of a classification model. It is defined as the proportion of correctly classified instances over the total number of instances in the dataset. In cases where the class distribution is imbalanced, it is often more useful to use other evaluation metrics, such as precision, recall, and F1 score. These metrics take into account the true positive, false positive, true negative, and false negative rates of the model, and provide a more nuanced assessment of its performance.

- *True Positive Rate (TPR)* : also known as sensitivity or recall, is a metric used to evaluate the performance of a binary classification model. It is defined as the

proportion of positive instances that are correctly identified by the model.

$$TPR = \frac{TP}{TP + FN} * 100 \quad (3.1)$$

where TP is the number of true positive predictions (instances correctly predicted as positive), and FN is the number of false negative predictions (instances incorrectly predicted as negative)

- *True Negative Rate (TNR)* : also known as specificity, is a metric used to evaluate the performance of a binary classification model. It is defined as the proportion of negative instances (or examples) that are correctly identified by the model.

$$TNR = \frac{TN}{TN + FP} * 100 \quad (3.2)$$

where TN is the number of true negative predictions (instances correctly predicted as negative), and FP is the number of false positive predictions (instances incorrectly predicted as positive).

- *False Positive Rate (FPR)* : is a metric used to evaluate the performance of a binary classification model. It is defined as the proportion of negative instances (or examples) that are incorrectly identified as positive by the model.

$$FPR = \frac{FP}{FP + TN} * 100 = 1 - Specificity \quad (3.3)$$

where FP is the number of false positive predictions (instances incorrectly predicted as positive), and TN is the number of true negative predictions (instances correctly predicted as negative).

- *False Negative Rate (FNR)* : is a metric used to evaluate the performance of a binary classification model. It is defined as the proportion of positive instances (or examples) that are incorrectly identified as negative by the model.

$$FNR = \frac{FN}{FN + TP} * 100 = 1 - Recall \quad (3.4)$$

where FN is the number of false negative predictions (instances incorrectly predicted as negative), and TP is the number of true positive predictions (instances correctly predicted as positive).

- *Accuracy (Acc)* : is a common metric used to evaluate the performance of a machine learning model. It measures the proportion of correctly classified instances (or examples) out of the total number of instances.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} * 100 \quad (3.5)$$

where TP is the number of true positive predictions (instances correctly predicted as positive), TN is the number of true negative predictions (instances correctly predicted as negative), FP is the number of false positive predictions (instances incorrectly predicted as positive), and FN is the number of false negative predictions (instances incorrectly predicted as negative).

- *Precision*: is a metric used to evaluate the performance of a machine learning model, particularly in binary classification problems. It measures the proportion of true positive predictions (instances correctly predicted as positive) out of the total number of positive predictions made by the model.

$$Precision = \frac{TP}{TP + FP} * 100 \quad (3.6)$$

- *Recall*: also known as sensitivity or true positive rate, is a metric used to evaluate the performance of a machine learning model, particularly in binary classification problems. It measures the proportion of true positive predictions (instances correctly predicted as positive) out of the total number of actual positive instances in the dataset.

$$Recall = \frac{TP}{TP + FN} * 100 \quad (3.7)$$

3. Real-time Big data Analytics framework

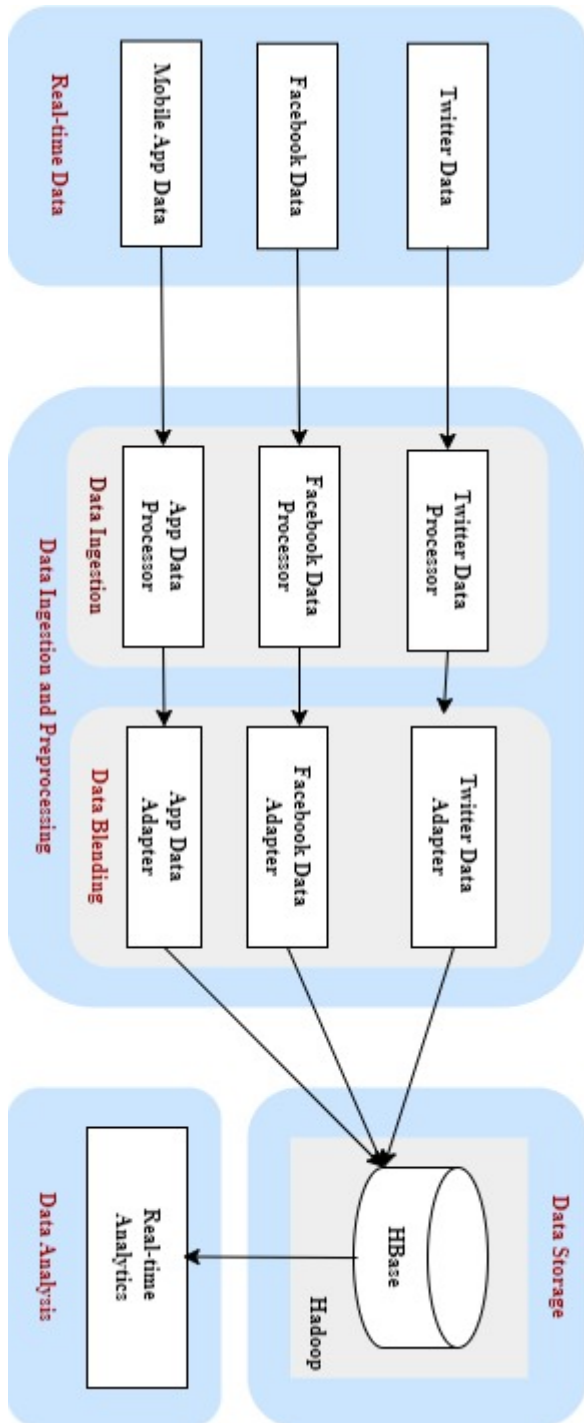


Figure 3.4: Experimental setup for real-time big data analytics using multiple data sources

3.4 EXPERIMENTAL EVALUATION

The complete experimental setup includes three important phases: data ingestion mechanism, data preprocessing and storage, and real-time analytics engine. Apache Flink, one of the big data analytics tools for real-time analytics, is used for experimental work and Apache NiFi to configure the real-time data ingestion mechanism. Apache Hadoop is used for the storage system in this experimental work. Figure 3.4 illustrates the workflow of the real-time analytics process using multiple data sources. Three different data sources were identified as input data for experimental work, where data collected in real-time. The data from each source is ingested and filtered by respective data ingestion processors and then passed into corresponding adapters for the data blending mechanism. Each adapter is comprised of a mechanism to read the data from the respective processor whenever new data arrives at the processor. When the processor passes the new data to the adapter, a real-time job is executed to preprocess the data with a data blending mechanism and store it on the HBase table on top of Hadoop. Here HBase supports real-time read/write access to the data. The preprocessed data stored on the HBase table is blended data from multiple sources that can be used to process a real-time analytical solution further to make desired data-driven decisions.

The critical approach in the proposed work is the data blending mechanism for preprocessing the data. The data from multiple sources prepared ready for further analytics process. Data preprocessing is a critical step in the analytics process as it takes the maximum time of the entire process. The quality of the analytical result purely depends on the quality of the data used. Preprocessing of the input data using appropriate preprocessing mechanisms is necessary for better results. In the proposed work, analytics is to be performed in real-time, where it is a challenging task to preprocess the data as the data arrives continuously at the data collection end. Preprocessing is to be done whenever new data ingested into the system. In the proposed mechanism, the data from multiple sources are used as input, whereas kinds of literature referred to are targeting the single source of data. When data from multiple data sources are used in the analytics, each source may consist of data in

different formats and structures. The proposed design mainly consists of three components: processors for data ingestion, adapters for data blending mechanism, and real-time analytics engine for final data-driven solutions. Data ingestion processors are responsible for data collection in real-time and the necessary filtering of expected data from the data sources. The adapters for the data blending mechanism are used to preprocess the data to make it ready for analytics and append it into blended data. The purpose of a real-time analytics engine is to analyze the incoming data streams sent from the adapters to process it for the desired data-driven solution further.

3.4.1 Data Ingestion Processor

The input data get collected from all the identified data sources in real-time. For each data source used, separate data ingestion mechanisms are configured to ingest the data into the analytics system. Separate data ingestion processors are written using the Apache NiFi tool for all three data sources. The purpose of each data ingestion processor is to read the data stream from the particular source immediately once data get generated at the source. For the experimental work, real-time data from Twitter, Facebook posts, and citizen complaint data from the mobile application are used as input data sources. A separate data ingestion processor is written for each of the data sources, where each processor performs the task of real-time data ingestion along with the initial stage preprocessing of data. The incoming data is filtered in the initial stage of preprocessing to extract only the data related to crime.

For example, considering Twitter data input, the data ingestion processor is configured to ingest real-time tweets using Twitter API. The proposed mechanism is implemented by creating a knowledge base, which helps in streaming only the tweets consisting of words related to the crime are considered necessary data for our analytics process, and other tweets are discarded. The workflow of each processor is as shown in figure-x. Each processor is configured for the respective input data source with the data ingestion mechanism of real-time data. In data ingestion processors, data filtering is done to refine the data to select only the crime-related data of the specific city and discard any other unrelated data streams. If the values of location in the incoming data

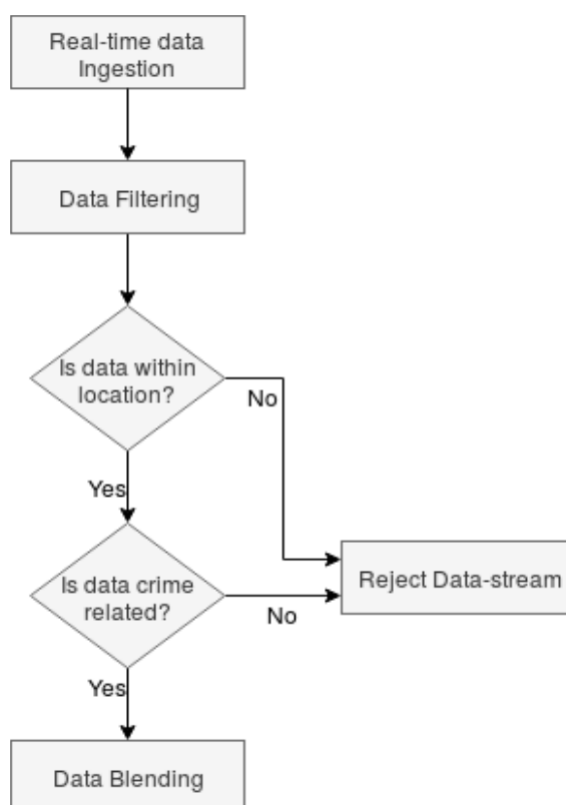


Figure 3.5: Workflow of data ingestion processor

match the city location values, then the data is considered for the further process; otherwise, data is discarded directly. Further, the actual content of the accepted data is verified for having any information related to crime.

The incoming data streams were filtered based on city location values and further verified for whether contents of the incoming data related to crime or not. The knowledge base consists of crime-related words and phrases to compare it with the incoming data to find out whether any crime-related information is present in the incoming data. For experimental work, 565 words and phrases related to different categories of crimes are used in the knowledge base with the help of Cambridge and Macmillan dictionaries. The contents of the knowledge base are used to verify the crime-related information in the contents of the incoming data stream. If any matching information present in the incoming data, then the data stream is passed to the next stage for preprocessing. The outputs of the processors are passed through respective adapters for the data blending mechanism.

3.4.2 Data blending mechanism

Real-time data ingested from each source by respective data ingestion processors are passed to respective adapters. Each adapter process the incoming data from the respective processor with the data blending mechanism. These adapters are the real-time jobs written using Apache Flink as the real-time processing tool. With its streaming architecture, Apache Flink helps to process the events in real-time with consistently high speed with low latency. In this experiment, all three data sources used for the analytics streamed from the data ingestion mechanism are in javascript object notation (JSON) format, but the structure of the data is different in each source. Data blending mechanism is the process of combining the data from multiple sources into a single dataset. Data blending is a different mechanism than the data integration process. Data blending is about working with multiple data sources by preparing them and joining them together for a specific use case, whereas data integration typically stores as a single source in the data warehouse for a user to access.

The proposed data blending mechanism is implemented with adapters to process data streams from the respective data ingestion processors. The adapters are written as Flink jobs that can read new data from the respective processor as and when it arrives. A blending mechanism combines the data received from the different processors and stores it on specific data storage for further use. Here, we use HBase to store the blended data received from the adapters. The data stored is further used by the real-time analytics engine for a desired data-driven solution.

The working of the Twitter data adapter is as shown in Algorithm [3.1](#). Here, Twitter data adapter can read the data stream from the respective data ingestion processor immediately once it is available. The adapter for the Twitter data source is written as a Flink job that reads each new input JSON file from the output of the Twitter data ingestion processor. This JSON file is parsed to filter the target fields, which are the valuable information stored on blended data for further analysis. In the JSON file from the Twitter data source, the values from target fields such as created-at, name, location, and text are considered for the analytics at the next stage. This information from each of the incoming data streams is appended as a new row on the HBase table. An additional

information source-id is stored as '1' for all the new appended rows from the Twitter adapter. The source-id is to be used in the further process to find the identity of the data source.

Algorithm 3.1: Twitter data adapter

Input : Data stream from Twitter data ingestion processor
Output: Blended data using multiple data sources

- 1 Parse the datastream to select target fields (*created-at, name, location, text*)
- 2 For each target field
- 3 Append the values of target fields as new row on HBase table as
- 4 Time \leftarrow valueof(*created-at*)
- 5 User \leftarrow valueof(*name*)
- 6 Location \leftarrow valueof(*location*)
- 7 Contents \leftarrow valueof(*text*)
- 8 Source-id \leftarrow 1
- 9 Repeat from step-1 for new data stream

Like the Twitter data adapter, the data blending adapters for the other two data sources are used. The working of the adapters for the other data sources used in the experiment is also similar to the Twitter data adapter. The structure of incoming data is different, with different attribute names in each data source. The working of the Facebook data adapter is as shown in Algorithm [3.2](#).

Algorithm 3.2: Facebook data adapter

Input : Data stream from Facebook data ingestion processor
Output: Blended data using multiple data sources

- 1 Parse the datastream to select target fields (*created-time, id, location, message*)
- 2 For each target field
- 3 Append the values of target fields as new row on HBase table as
- 4 Time \leftarrow valueof(*created-time*)
- 5 User \leftarrow valueof(*id*)
- 6 Location \leftarrow valueof(*location*)
- 7 Contents \leftarrow valueof(*message*)
- 8 Source-id \leftarrow 2
- 9 Repeat from step-1 for new data stream

The working of the Facebook data adapter is similar to the adapter for Twitter data, but the target fields selected are created-time, id, location, and message. The values of these target attributes in the input data are appended on the blended table. Here, the

source-id is stored as '2' for all new rows appended on blended data. Similarly, for the third data source used, an application data adapter is used where the target attributes such as created-time, complaint-id, incident-location, and description are selected for further process. For this data source, source-id as '3' is assigned for all new rows appended on the blended table. Similarly, one can add any other data source available for the analysis. Algorithm 3.3 shows the working of the application data adapter.

Algorithm 3.3: App-data adapter

Input : Data stream from App-data ingestion processor

Output: Blended data using multiple data sources

- 1 Parse the datastream to select target fields (*created_time*, *id*, *location*, *message*)
 - 2 For each target field
 - 3 Append the values of target fields as new row on HBase table as
 - 4 Time \leftarrow valueof(*created_time*)
 - 5 User \leftarrow valueof(*id*)
 - 6 Location \leftarrow valueof(*location*)
 - 7 Contents \leftarrow valueof(*message*)
 - 8 Source-id \leftarrow 3
 - 9 Repeat from step-1 for new data stream
-

The blended data updated on the HBase table is used for further analysis in the desired data-driven solution. The real-time analytics engine configured for the specific data-driven solution can read each new row of data updated on a blended table. Apache Flink is used to designing the real-time analytics engine that can read each new data entry immediately when updated on the HBase table. This proposed system helps in the real-time analysis of desired data-driven solutions using multiple data sources. The proposed system is flexible enough to add additional data sources as input by configuring a data ingestion processor and a data blending adapter. Data analysts can easily add any possible data sources of interest as input data for the specific analytical solution to improve the performance of the system.

Performance cost of the proposed design is depending on the configuration and performance of the different tools used in the framework. For this experimental work, 3 node clusters using CPU with 4 cores and physical memory of 16 GB are used. Performance cost is also depending on the volume of the data to be ingested for the analysis and complexity of the queries to be analyzed. At the initial stage the

performance evaluation of the proposed design is done for a standard word count problem with 1 GB and 2 GB of data load. The average execution time for this is 53 seconds. This is better performing as compared to the analysis using traditional big data processing system like Hadoop where it takes an average of 5 minutes and 37 seconds. Performance can be increased by configuring with cluster of more nodes with higher configuration. In real-time analytics execution time is the key factor to be considered for the better performance of the system.

3.5 DISCUSSION

The primary goal of the proposed real-time big data analytics framework is to consider multiple data sources that generate real-time data that can be useful in developing data-driven solutions for public safety. Adding multiple data sources in the analytics system increases the number of valuable data insights for the data-driven solution. The proposed real-time big data analytics framework with a data blending mechanism helps the data analysts to collect more input data in real-time. Figure 3.6 shows the sample observations for a specific period with the blending mechanism. It represents the comparison of the data appended on the blended table from each data source. The values for each data source used in the experiment are calculated using source-id in the blended table. The total number of data updated on an hourly basis is observed for each source-id. The consolidated data appended on an hourly basis is considered data from multiple sources. The x-axis represents each hour of execution of the experiment, and the y-axis represents the number of data rows appended on the blended table related to the respective source.

Similarly, Figure 3.7 shows a sample observation for the number of crime data of different categories from different sources for a particular period. It shows the number of data samples of different categories of crime data such as accident data, fire & gang war, murder & rape, and robbery from all three data sources used in the experiment. When the target data for an analytical solution is spread across different data sources, it is important to collect all possible data samples from different data sources. When data is used from multiple sources in the experiments, analysts can benefit from processing

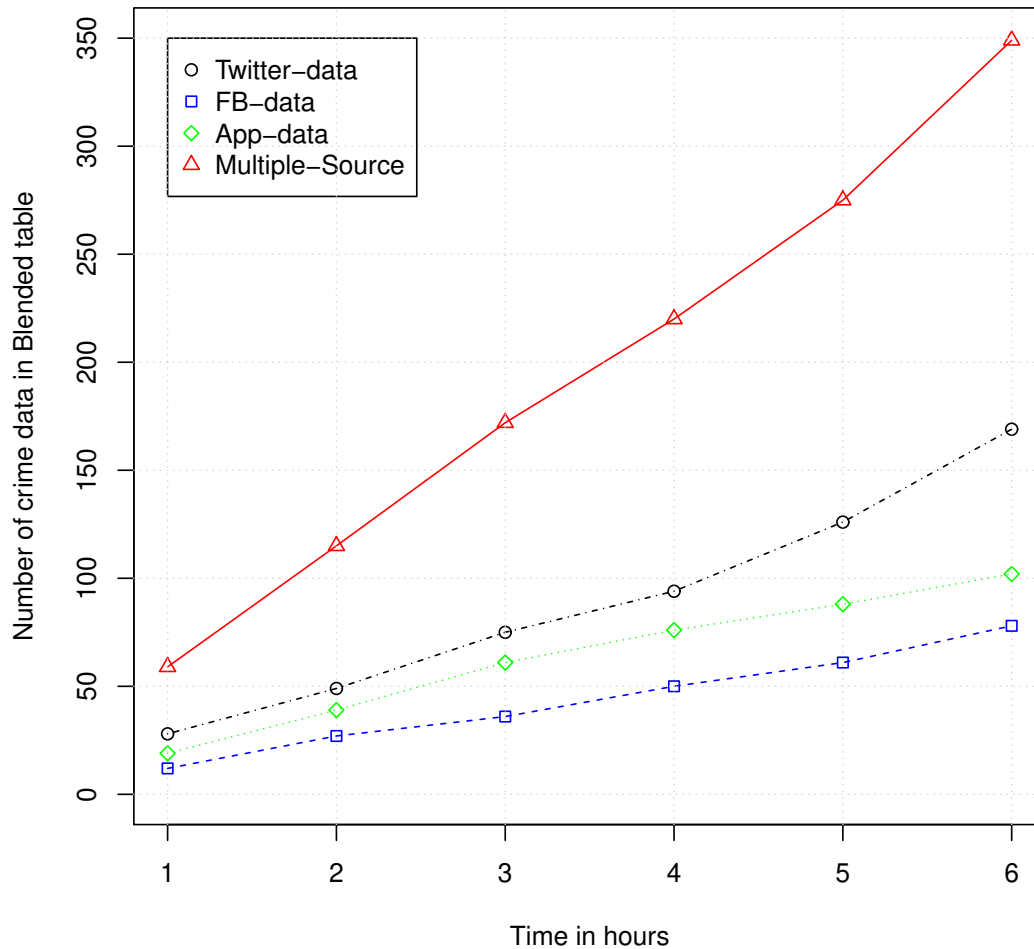


Figure 3.6: Sample observation of data blending of real-time data from multiple sources

more data to achieve better analytical results.

Contribution and Limitations of the proposed design

A real-time big data analytics framework is proposed with data blending approach for multiple data sources. In real-time analytics where data spread across multiple data sources, it is essential to quickly collect data from all disparate sources for rapid analysis. The proposed data blending mechanism with data ingestion mechanism helps in achieving this. The proposed framework is also flexible to add any additional data sources by adding respective ingestion processor and data blending adapter in the

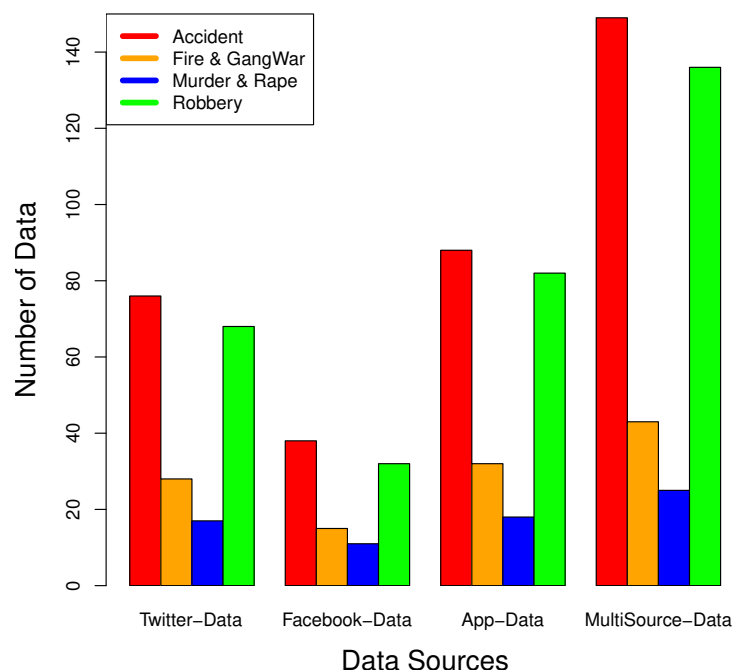


Figure 3.7: Sample observation of data blending of real-time data of different categories of crime using multiple data sources

existing framework

The limitations in this work is, for evaluation of the proposed design experimental setup is made with 3 node clusters using CPU with 4 cores and physical memory of 16 GB. Running the experiment on many clusters with huge volume of data was not performed for the evaluation. In many frameworks as observed the performance varies with increasing clusters and size of the data, this could not be recorded in this work. The experimental work is limited to use three different data sources generating text data. The framework is flexible to use more number of data sources. Also, data sources generating non text data are not considered in the experimental evaluation.

3.6 SUMMARY

In this chapter, we have proposed a real-time big data analytics framework using multiple data sources. Real-time data analytics is most effective in many data-driven solutions such as public safety for quicker response and actions. The public safety

3. Real-time Big data Analytics framework

system must be capable of making timely decisions and predictions for detecting and preventing crimes. It is crucial to perform the entire analytics process as quickly as possible by considering all possible data generated within the city. In the data analytics process, most of the time is spent preprocessing the data to prepare it as analytics-ready. In real-time analytics, whenever new data arrives in the data collection phase, it must be preprocessed and analyzed for a desired analytical solution without much delay in the entire process. Collecting the maximum data for the analysis helps in achieving better outcomes. Hence, it is essential to use multiple data sources for input data in analytics to find much better and more accurate outcomes.

The real-time analytics framework with the data blending approach proposed in this work is appropriate to preprocess the data from multiple sources in real-time. The blended data is used in further analysis to find data-driven solutions such as crime detection and monitoring. The proposed framework is flexible to add any new data sources for the analysis at any time with minimal configuration. In the next chapters, the proposed framework is used to design a real-time emergency alert system and a crime prediction model.

CHAPTER 4

REAL-TIME EMERGENCY EVENT DETECTION SYSTEM

Public safety is an essential service offered in smart city projects to provide better safety and security for individuals and city infrastructure. The advancement in Information Technology and the Internet of Things created much scope for using smart applications in the city to enhance the quality of service, leading to a better life in cities. This digitization generates vast data from distinct sources like social media, IoT, sensors, and any user-generated content from smart applications. The data generated within the city are analyzed to discover valuable insights for producing better data-driven decisions and predictions, which are more crucial for efficient city administration. For example, an emergency event detection system for monitoring crime incidents in real-time using smart city data helps the city administration provide better services in the city.

Since the data generated at various sources, the crime monitoring system must use data from multiple sources to build a better system. The real-time big data analytics system proposed in Chapter-3 can be used to design a real-time emergency events detection system to help city administrators in taking quick actions for the safety of people and city infrastructure.

4.1 INTRODUCTION

Improving the public safety system has a significant impact on the success of any smart city. A smart policing system for public safety is an essential aspect of all smart city projects. Technological revolution and advancement in analytics facilitate the design of intelligent solutions for public safety. Data-driven solutions play a pivotal role in smart solutions for public safety. Many such solutions are adopted in different cities, such as traffic and parking monitoring, predicting crime hotspots, monitoring emergency incidents such as natural crises, and many others. Many such solutions for these services involve manual processing where there can be scope for improvising the system by adopting better analytical solutions. The way digitization entered the daily lifestyle causes different sources to produce a massive amount of user-generated content. The smart police system has an exquisite gain in analyzing this data for valuable insights in making data-driven decisions. Many such attempts are being made with the use of social media analytics by some city police departments. The analytical solutions are more effective and beneficial to city authorities whenever appropriate decisions or actions are taken at the right time. Real-time data analytics is plenty useful to achieve this.

In the public safety system, the actions taken or decisions made by the law enforcement system in the initial minutes of an emergency incident are critical. The immediate actions during the crime incidents such as vehicle accidents, robbery, murder, and fire incidents help the police department manage the incidents better. Designing the smart systems for alerts at the right time by law enforcement authorities is beneficial in taking the right action at the right time. The different smart applications, social media, and smart devices are widely used in the smart policing system for monitoring such crime incidents. With technological advancement and digitization, the necessary data get generated within the city from different sources. Analyzing the data generated from all possible sources for a data-driven solution helps design a better alert system. It is challenging for the smart policing system to collect valid data from different sources and to analyze it for appropriate actions at the right time.

In this chapter, a real-time-based emergency event detection system is proposed for public safety in a smart policing system. The main objective of the proposed system is to use the input data from multiple sources. The real-time big data analytics framework proposed in Chapter-3 is used in designing the proposed emergency event detection system. The input data from the identified data sources are ingested into the analytics system using a data ingestion mechanism in the proposed framework. The data blending adapters help create a common dataset from all the sources that can be further used to find the emergency incident. The real-time analytics engine consists of an event processor with a machine learning model for emergency event classification.

4.2 PUBLIC SAFETY IN SMART CITY

A smart city as a safe city is the integration of technology to enhance the effectiveness of the process to handle the crime with quicker response to emergencies within the city to create a healthy environment for citizens (Hartama et al. 2017). It is challenging for the city administrations to provide safety to the citizens and infrastructure due to the rapid increase in the urban population (Isafiade and Bagula 2017). Public safety to provide safety and security to the residents and city infrastructure is a fundamental operation of any smart city. The emergency incidents management system used by the city administrators is crucial in improving the quality of life within the smart city (Alazawi et al. 2014). An emergency incident may happen in any location within the city in an unpredictable way. In recent days, the police departments have adopted many solutions to manage emergency situations in a better way. With the technical advancement in the Internet of Things (IoT) and sensor devices, such technological solutions are more popular in monitoring emergency incidents. Sensor-based solutions are a recent trend in the smart city for generating emergency incident alerts (Costa and de Oliveira 2020). Most city administrators widely use surveillance cameras that provide visual data to monitor emergency events (Costa 2020). With these sensors data, analysts can use the data from social media and mobile applications that generate huge amounts of user-generated content.

Social media and mobile application users are rapidly increasing in recent days,

4. Real-time emergency event detection system

creating an opportunity for researchers to think of its crowdsourcing ability for emergency event management ((Landwehr et al. 2016), (Gao et al. 2011)). When emergency incidents happen in the surroundings, people respond and share their views, concerns on social media and related applications (Huang and Li 2016). The law enforcement authorities can observe such user-generated content to monitor the happening within the city to monitor the emergency incidents quickly and effectively. There are many such solutions proved that social media contents are one of the major information source for detection and prevention of several emergency incidents such as disaster management during earthquakes (Bai and Yu 2016), flood ((Cresci et al. 2015), (Cervone et al. 2016)), nuclear disaster (Acar and Muraki 2011), tsunamis (Ai et al. 2016), and wildfires (Vieweg et al. 2010). The types of emergency incidents are unknown in advance to detect and characterize the emergency-related incidents (Atefeh and Khreich 2015).

The use of social media and mobile applications by people creates a huge amount of data where a minimal amount of data can be relevant to emergency incidents and contains valuable information. Many previous works focus on extracting valuable information from social media data to detect emergency incidents. These works on detecting emergency incidents mainly use text classification techniques. (Huang et al. 2021) proposed a method to use Twitter data to build a classifier model trained for 26 different types of emergency incidents. The author used a technique to train the classifier model on certain categories of emergency incidents and tested it with other types of emergency incidents. An event detection system by using Twitter data is proposed by (Li et al. 2012) to detect car accidents. The tweets are analyzed to classify and aggregate using machine learning models for detecting earthquakes is proposed by ((Imran et al. 2013), (Caragea et al. 2011)). (Choi and Bae 2015) proposes a real-time monitoring system for disaster management using social big data. This system uses social media data, particularly Twitter data, and analyses tweets in real-time for any disaster-related information.

Most of the previous works on emergency event detection systems using social media data are focuses on a single type of crime incident. The public safety system

must focus on all types of emergency incidents to be monitored at the right time. The residents in the city may use different types of applications or social media platforms to share their views on the incidents happening in their surroundings. The city administration should focus on all possible sources to extract valuable information that can help to monitor emergency incidents. Law enforcement agencies are targeting on video or sensor devices installed within the city to monitor emergency events and look at the possible user-generated contents available within the city. There is much scope for the system that can collect the data from all possible sources and analyze it to detect emergency incidents. This chapter proposes a real-time emergency event detection system for public safety using data from multiple sources.

4.3 PROPOSED WORK

The objective of the proposed work is to design an emergency event detection system using multiple data sources. The framework discussed in Chapter-3 can be used to design the proposed system for emergency event detection system. It supports collecting the real-time data from the identified data sources in real-time and supports blending the data from all sources. The common dataset updated from the data blending mechanism can be used to analyze it to find a required data-driven solution. Here an event processor is designed to detect emergency events using the common dataset updated from the data blending mechanism. The workflow of the proposed design of a real-time emergency event detection system using multiple data sources is as shown in Figure 4.1.

The proposed design aims to target input data from multiple sources. The data ingestion mechanism and data blending mechanism support ingesting and preparing analytics-ready data, as discussed in Chapter 3. The preprocessed data are updated on a blended table created using HBase on top of the Hadoop storage system. The blended data is further passed into a real-time event processor responsible for classifying the input data stream as an emergency incident. The event processor is designed using a machine learning model with a binary classification technique to classify emergency incidents. It is easy to add any additional input data source of interest into the existing

4. Real-time emergency event detection system

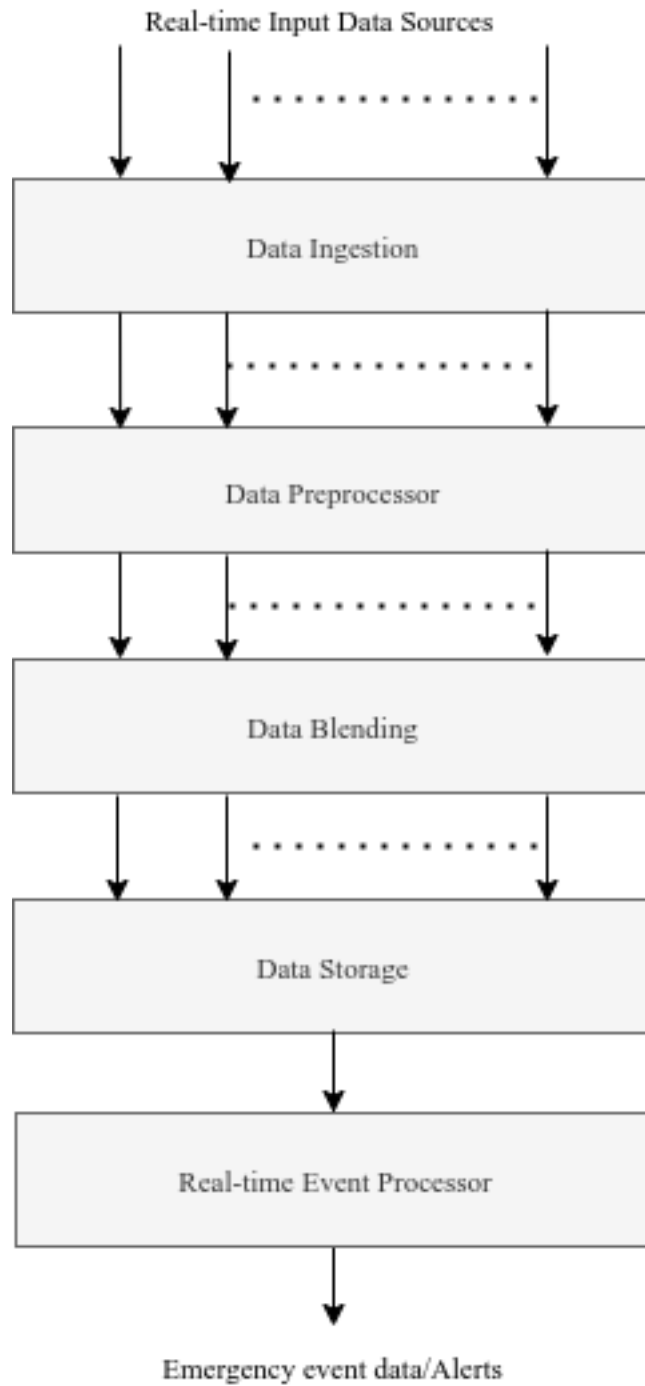


Figure 4.1: Workflow of real-time emergency event detection using multiple data sources

system.

4.4 EXPERIMENTAL EVALUATION

The overview of an experimental setup for real-time emergency event detection using multiple data sources is as shown in Figure 4.2. The proposed experimental setup is based on the real-time big data analytics framework proposed in Chapter-3. The framework supports collecting real-time data from multiple data sources for the analysis. For the experimental work, three different data sources are used that generates crime-related information. Social media data like Twitter data, Facebook posts, and citizen complaint data through mobile applications are the input data sources considered for the experimental work. The data ingestion processors are configured for each data source to ingest the real-time data generated at the particular source. The purpose of the data ingestion processor is to ingest the data from a specific source immediately once data is generated at the source. The ingestion mechanism is also integrated with a data filtering mechanism to ingest only the data related to crime from the given target location. For each data source used, separate data adapters are designed to read the data stream passed from the respective data ingestion processor. The purpose of the adapter is to preprocess the data and update the prepared data on a common dataset from all the sources used. The data blending mechanism used in the adapter helps to update prepared data from all three sources as a common dataset as analytics-ready. An event processor is designed as a real-time analytics process responsible for analyzing the common dataset to detect any emergency incidents. The event processor is a machine learning model to classify the input data as an emergency incident.

The experimental setup is similar to the experimental setup for a real-time big data analytics framework using multiple data sources, as explained in Chapter 3. Apache NiFi tool is used for configuring the data ingestion mechanism. The data ingestion processors for all three data sources are configured to ingest the data in real-time. Apache Flink is used as a real-time analytics tool. The data blending adapters for each data source are designed as Flink jobs responsible for preprocessing the data

and consisting of a data blending mechanism. The data blending adapters update the prepared data on a common table created on HBase on top of the Apache Hadoop storage system. The prepared data stored on the blended table is processed through a real-time analytics engine. The real-time analytics engine consists of an event processor with a machine learning model to detect any emergency incidents.

4.4.1 Event Processor

The purpose of the event processor is to process the event streams passed from the adapter to find any emergency incidents. The event processor uses a machine learning-based classification model to generate emergency event alerts from the incoming data stream. A training model is developed by using information about six different categories of crime incidents such as fire incidents, vehicle accidents, robbery, rape, murder, and gang war. The initial training model is created using the data related to these six categories of crime incidents data. This training data set is updated regularly as the model is tested with new incoming data streams. The newly arrived data stream contents from any of the three data adapters are verified for any emergency incidents. When such incidents are detected during the process, an emergency alert gets generated.

As and when the new data stream is passed to the event processor, the content of the data is processed for selecting the topic feature by adopting Latent Dirichlet Allocation (LDA) (Blei et al. 2003). Initially, non-English contents are filtered out using a language detection library, and then stop words are filtered out from the contents. Latent Dirichlet Allocation (LDA) is used to train a topic model that can output the distribution of topics. Then, the classification model developed in the event processor is used to find any emergency incident. This model has experimented with the most popular classification algorithms to choose the better one for the most relevant results.

In this work, the most commonly used classification algorithms in streaming data analytics, such as the Naive Bayes (NB) classifier, Support Vector Machines (SVM) classifier, Logistic Regression (LR), and Random Forest (RF) algorithms are used. NB classifier is a probabilistic classification algorithm based on the application of

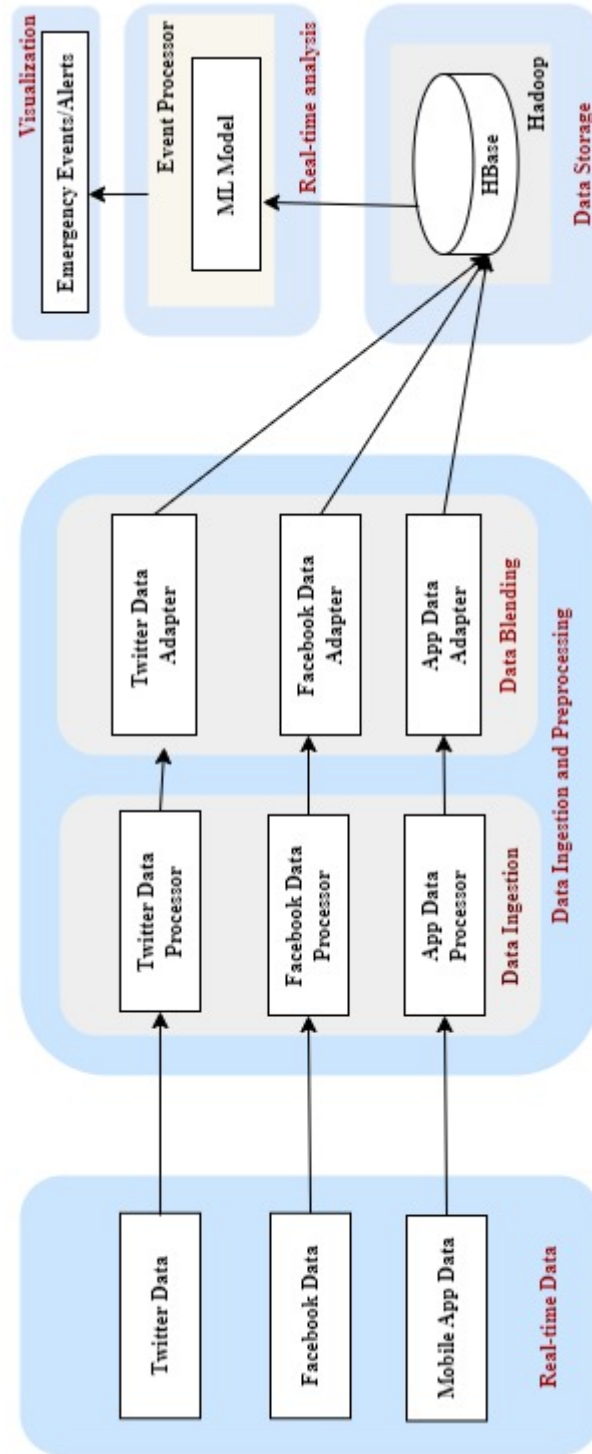
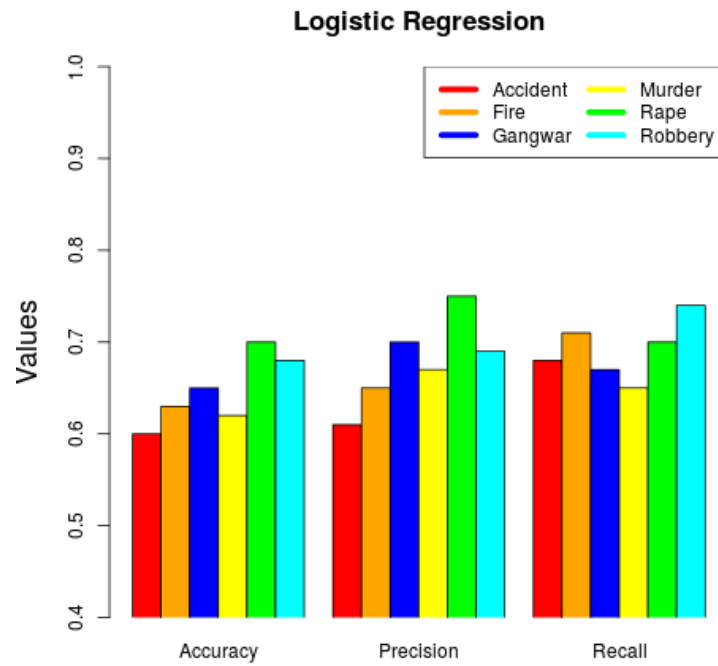


Figure 4.2: Experimental setup for real-time emergency event detection using multiple data sources

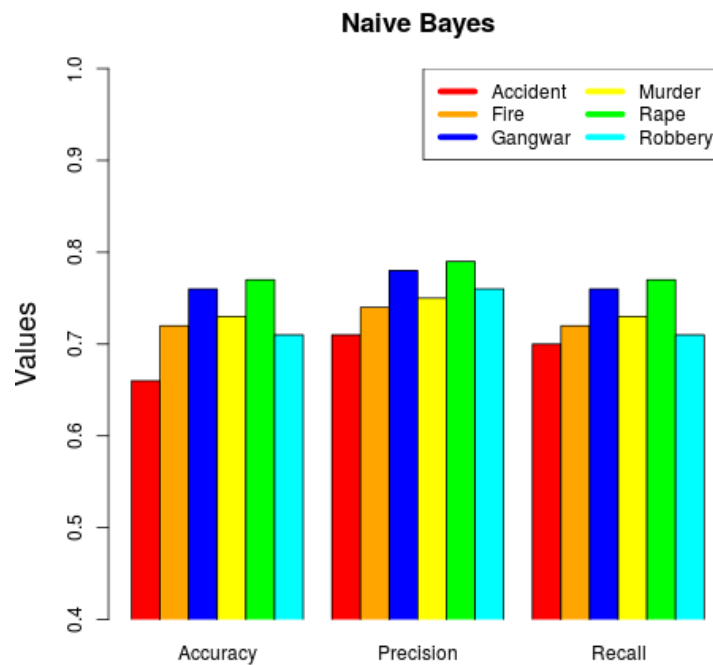
Bayes's theorem (John and Langley 1995). The model assumes that the presence of a specific feature is unrelated to the presence of any other feature. SVM classifier (Cortes and Vapnik 1995) is based on separating hyperplane according to which new samples are classified. Logistic Regression (LR) is a linear classifier that measures the relationship between the dependent variable and independent variables by determining the probabilities using a logistic function (Witten et al. 2011). Random Forest (RF) is based on the forest construction procedure where features as nodes grow like branches of a tree, finally combining all trees to form a Random Forest model (Breiman 2001). To evaluate the performance of the model, frequently used three statistical metrics, accuracy, precision, and recall, are used. The NB classifier gives the most accurate results out of the four different classifiers used. Hence this classifier is used to generate emergency alerts in the proposed system.

4.5 RESULTS AND DISCUSSION

The performance of the four different classifiers used in the experiment for emergency events classification is as shown in Figure 4.3 and 4.4. In this experiment, we target for emergency incidents by considering the six different categories of crimes such as fire incidents, vehicle accidents, robbery, rape, murder, and gang-war. The performance metrics are computed for each category of crime incidents using four different classifiers. Then, the overall measure is calculated as the average of the per class measure. Here NB classifier achieves a higher accuracy of 73%, which is a 3% improvement over RF, 5% improvement over SVM, and 8% improvement over LR classifier.



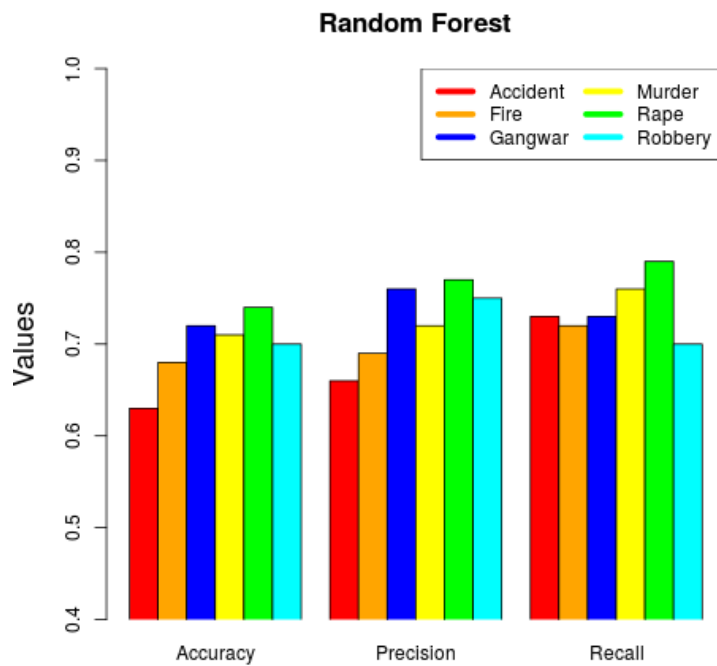
(a) Logistic Regression Classifier



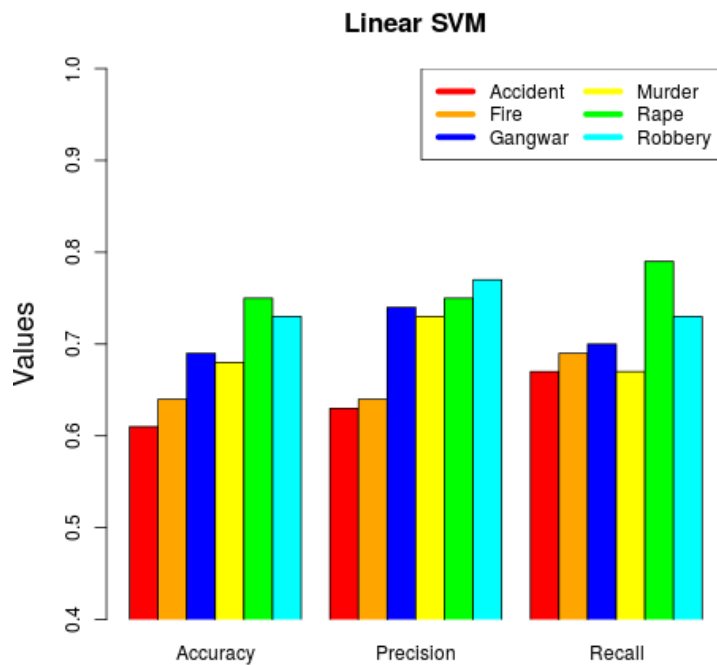
(b) Naive Bayes Classifier

Figure 4.3: Performance Comparison of Classification Algorithms for Emergency Alerts- (a) Logistic Regression and (b) Naive Bayes classifier

4. Real-time emergency event detection system



(a) Random Forest Classifier



(b) Linear SVM Classifier

Figure 4.4: Performance Comparison of Classification Algorithms for Emergency Alerts- (a) Random Forest and (b) Linear SVM

4.6 SUMMARY

A real-time-based emergency alert system to help the public safety solution is implemented using a machine learning-based classification algorithm with the proposed framework. Real-time data analytics is invaluable in many data-driven applications for quick response and actions. Public safety is one of the key services in smart city applications, where timely decisions and predictions are much beneficial for detecting and preventing crimes. In real-time data analytics, it is crucial to perform the entire analytics process as quickly as possible. In the data analytics process, the majority of the time is spent preprocessing the data to make it prepared as analytics-ready. In real-time analytics, whenever new data arrives in the data collection phase, it must be preprocessed and analyzed for a desired analytical solution without much delay in the entire process. Collecting the maximum data for the analysis helps in achieving better outcomes. Hence, it is essential to use multiple data sources for input data in analytics to find much better and more accurate outcomes. The real-time analytics framework with the data blending approach proposed in this work is appropriate to preprocess the data from multiple sources in real-time. A real-time event processing mechanism is proposed for emergency alerts to any such incidents within the city. Analytical solutions such as predictions and data-driven decisions are possibly more accurate when all available data are used instead of a single data source. The proposed mechanism is much more flexible in adding any new data source to be used for the analytics with the existing experimental setup. The experiment is carried out with four different classification algorithms, and the comparison of results shows that Naive Bayes classification performs with an accuracy of 73% which is better in generating emergency alerts.

CHAPTER 5

REAL-TIME ANALYTICS BASED CRIME PREDICTION USING MULTIPLE DATA SOURCES

Early detection and prevention of crime are the most crucial challenges for better safety and security within the city. For a long time, Predictive policing has been used to monitor crimes based on past crime records by identifying the crime hotspots. Recent developments with digitization in various applications generate data continuously on a large scale. It gives much scope for data scientists to analyze the data immediately after it gets generated at the source. Technological developments in streaming data analytics and real-time analytics play a key role in making real-time predictions and decisions. The proposed system is designed to use real-time data from multiple sources for crime hotspot prediction that improves the performance of the prediction model.

In the proposed work, real-time data from multiple sources are used along with historical data to enhance the performance of the prediction model. The proposed real-time crime prediction system is designed using a real-time big data analytics framework proposed in Chapter 3 that incorporates a real-time data ingestion mechanism accompanied by a data blending approach for multiple data sources. The experimental work is carried out with the proposed system using three different data sources for crime prediction. The data ingestion and data blending approach used for designing the system is flexible to add any additional data sources of interest for real-time prediction. The real-time data, along with the historical data, helps in

achieving better performance. It is also tested with different time intervals to update the prediction model.

5.1 INTRODUCTION

Predictive policing has been used for the past few years by many law enforcement agencies for the early detection of crimes. It involves an analytical and statistical approach to forecast possible crimes by using crime data from different sources such as FIR data, social media data, personal communications, etc. Advancement in data analytics and big data is reflected in more analytical solutions introduced in the smart policing system. Crime hotspot analysis is one of the popular methods used by many police departments for monitoring crime. The objective of the crime hotspot prediction system is to predict the geographical locations with crime risk based on crime data patterns. It helps law enforcement authorities to prevent crimes by police patrolling in such areas. Different analytical methods such as data mining, regression models, classification algorithms, Spatio-temporal analysis are used to increase the accuracy of hotspot prediction. Law enforcement agencies use such prediction results to make proper decisions and actions based on the hotspots identified. In certain situations, it may require police presence on the crime incidents, additional attention, quick responses, periodic visits by the police patrolling depending on the types of crimes in that area. It leads to more scope for early prediction of crime hotspots and timely actions to prevent crimes.

Existing crime hotspot prediction systems are built based on collecting historical criminal records from the police departments and generate the prediction. However, the data often collected yearly may be less effective over time in cities because of the significant number of floating populations in urban areas. In this work, an attempt is made to collect the data in real-time, as a large amount of data in smart cities is accessible in a streaming manner. The rapid growth in big data analytics and streaming data analytics made it possible to analyze data immediately after getting generated at the source. Furthermore, more and more technological adaptation in day-to-day life in the city generates a large amount of data continuously. Such data generated from the

users and digital infrastructure can be analyzed instantly to derive useful data patterns. It is also essential to use all available data sources as input to the analytical system. As more and more data sources are used than a single source, it can produce more valuable insights into the prediction system that helps in increasing the accuracy of the system.

In this chapter, a real-time based crime hotspot prediction system is proposed for public safety in a smart policing system. The proposed system collects user-generated content within the smart city from multiple sources as input data. A framework is designed to ingest real-time data related to crime from multiple sources. The data ingested from multiple sources are further processed with a data blending mechanism to integrate it as analytics-ready. The proposed data blending mechanism is scalable to add any number of data sources along with corresponding addition of data ingestion and blending mechanism in the analytics system. A crime hotspot prediction mechanism is designed using this blended data from multiple sources. Here the new data is continuously updated at the source. The data used at present is used as historical data in later stages. The proposed crime hotspot prediction approach is an efficient mechanism by analyzing various possible data sources in real-time. The results show that this approach gives more accurate predictions than the traditional approach, where only the historical data is used from a specific source.

5.2 CRIME HOTSPOT AND PREDICTIVE POLICING

For the past few years, predictive policing has been used by many law enforcement authorities to monitor and prevent crimes. Predictive policing uses analytical techniques to identify likely targets for police intervention and prevent crime or solve past crimes by making statistical predictions, as stated by (Perry et al. 2013). Statistical analysis is widely used to predict crime events and identify likely targets for the police. Crime hotspot prediction is one of the traditional methods used to analyze and visualize crimes across time and space (Hardyns and Rummens 2018). The main goal of predictive policing is to generate future crime trends and patterns and adapt them to the crime prevention process (Gerber 2014). It helps the city administrators to use their security resources effectively (Meijer and Wessels 2019). At the earlier

5. Real-time analytics based crime prediction using multiple data sources

stages, the data collected from the crime records, statistics from the police departments are used to generate future crime patterns. Due to the advancement in Information Communication Technology (ICT), crime-related data generated from various user-generated contents such as social media data and devices such as sensors, video surveillance cameras are widely used in crime prediction. With the technological adaptation in the IoT and Cloud, smart policing for public safety is among the most common services in any smart city project (Davies 2020).

Data-driven approaches with Intelligence-led policing and hotspot predictions are becoming more popular in smart policing. Intelligence-led policing is a process of analyzing crime data to make policing strategies and operations for reducing and prevent crime with appropriate management of policing resources (Ratcliffe 2016). Crime hotspot prediction system finds possible locations with a high crime rate using analytical tools (Hardyns and Rummens 2018). Crime hotspots help city administrators to make appropriate decisions in monitoring and prevent crimes. It helps smart policing system for appropriate management of limited security resources. With the rapid development of digitization in day-to-day life, vast amounts of information are generated continuously within the city. The challenge for the smart policing system is to use all possible information in their crime prediction and monitoring tools to find a better solution.

Crime data analytics to forecast future crimes have been used for a long time by many data analysts. The majority of the crime prediction tools focus on historical crime patterns collected from various sources (Chainey et al. 2008), (Ohyama and Amemiya 2018). The Geographical information system (GIS), environmental data (Mohler et al. 2011), and social media data are widely used for crime prediction by most researchers from the past few years (Andresen 2009), (Williams and Burnap 2015), (Zhuang et al. 2017). Many researchers use statistical techniques including regression (Kennedy and Dugato 2018), kernel density estimation (KDE) (Alves et al. 2018), Latent Dirichlet Allocation (LDA) (Hu et al. 2018) to identify the future crime area by using historical crime records. Advancement in crime hotspot prediction systems adapts machine learning algorithms to improve the performance of the system.

Crime hotspot prediction using a machine learning approach with SVM is proposed by (Gerber 2014). (Liu and Zhu 2017), (Vural and Gök 2017) proposed the Naive Bayes method to build the crime prediction models to forecast the crime locations. (Yu et al. 2014) proposed ensemble learning for crime forecasting using Spatio-temporal data. The advancement in Big data technology and Artificial Intelligence help researchers to develop predictive tools with better performance.

The recent development in the open data repository of crime data provides opportunities to improve the analytical solutions for crime detection and prevention. Streaming data analytics tools are popularly used in various applications for the past few years (Puentes et al. 2020). In recent days, streaming data analytics have been used for crime forecasts in real-time. The majority of real-time analytics in crime prediction uses social media data. An event detection system is proposed in (Hasan et al. 2018) to detect important events in real-time from Twitter data streams. (Zhou et al. 2016), (Fan et al. 2016) proposed similar work for city event detection for London city using Twitter data streams. (Ali et al. 2017) proposed an event detection system designed for real-time data analytics of IoT-enabled communication systems. A real-time monitoring system for disaster management using social big data analytics is proposed in (Seonhwa and Byunggul 2015). (Zhang and Yuan 2015) proposed a predictive model for air quality monitoring by analysis of real-time meteorology data from Beijing city. A prediction system designed to predict future terrorist incidents using real-time news data sources is proposed in (Toure and Gangopadhyay 2016) .

This chapter proposes a real-time crime prediction system that targets both real-time data and historical data. The system uses real-time data from multiple sources to enhance prediction performance. In recent days, some researchers attempt to analyze multi-source data to improve prediction accuracy. The data generated from different social media platforms can be integrated to enhance big data-driven crime prediction models, as stated in (Pina-Garcia and Ramirez-Ramirez 2019). (You et al. 2019) proposed a mechanism for harnessing multi-source data about public sentiments and activities for an informed design that addresses the process from data collection to data visualization. A framework proposed for collecting and analyzing data from social

media and surveillance cameras to describe public safety events is proposed in (Xu et al. 2019). The proposed system uses the blending of data from different sources to improve the prediction model performance.

5.3 PROPOSED WORK

In the proposed work, real-time data used to find the crime-related data patterns to predict the crime hotspots. The input data from identified sources are streamed into the analytics system immediately after it gets generated at the source. Such data is processed when it arrives at the desired analytics system and stored for further use in the crime hotspot predictions. Hence the proposed framework is designed based on the working of Lambda architecture, as explained in chapter 2. Real-time data ingested from identified sources, as and when it gets generated at the respective data sources and then preprocessed to prepare it as analytics-ready. Here input data is streamed through the real-time layer and then passed to the serving layer. The prepared data are simultaneously updated in the data store and used in later stages using the batch layer as batch views along with the real-time views. Real-time views of the serving layer manage to process the real-time queries requested by the user.

The design of the crime hotspot prediction system using real-time data analytics from multiple sources is based on the proposed design for real-time big data analytics for multi-source data explained in chapter 2. The overall process for crime hotspot prediction system consists of different phases as data ingestion from real-time data sources, data preprocessing, data storage, real-time analytics engine for prediction, as shown in Figure 5.1. The functioning of each of the phases during the real-time crime prediction is conferred in the following.

Data Ingestion:

The main aim of the proposed design is to collect real-time data from multiple sources. Here, separate data ingestion processors are designed for each identified data source to ingest real-time data into the analytics system. Each data ingestion processor for a specific data source is configured to ingest the data immediately after it is generated at

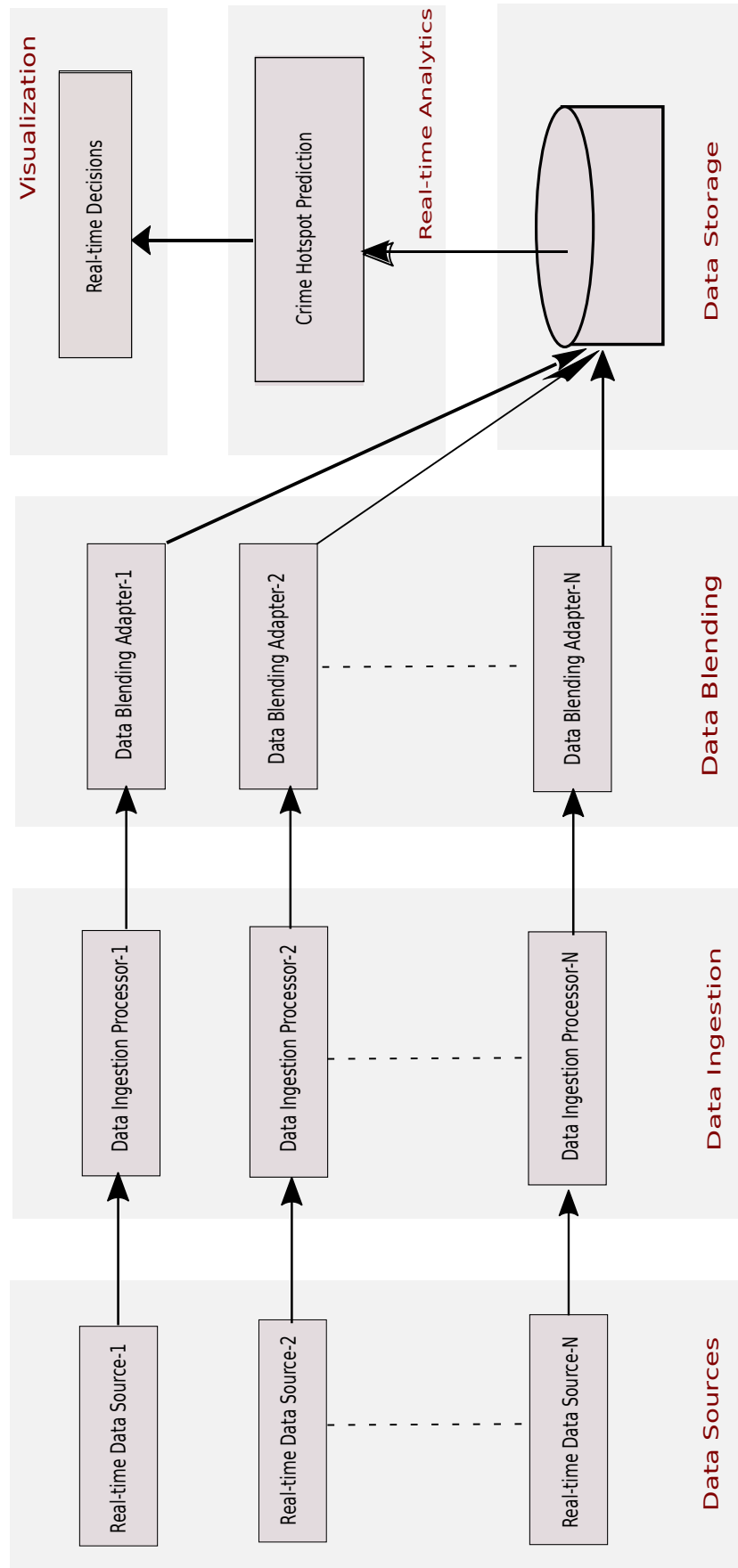


Figure 5.1: Proposed design for real-time crime prediction using multiple data sources

the particular data source. The data ingestion processor is also responsible for filtering only the crime-related data. The filtered data from each of the ingestion processors is passed into the next stage for the preprocessing. Any new data source of interest can be added to the system by adding a respective data ingestion processor.

Data Preprocessor:

The data stream passed from the data ingestion processor is passed into the respective data blending adapters. The purpose of the data blending adapter is to preprocess the noisy data input and also data blending from different sources into a single common dataset. In this phase, noise removal, tokenization, and normalization techniques are used to preprocess the contents of the data from an incoming data stream. The target attributes identified from the particular data sources are filtered and updated as a common dataset on a data store.

Data storage:

The proposed system uses both historical and real-time data. The data used in real-time at the moment are used as historical data in the later stages. Apache Hadoop is used as a storage system, which supports read and write access for Apache Flink used as a real-time analytics tool. An HBase table on top of the Hadoop ecosystem is used as a common dataset from all the input data sources, which is updated continuously from the proposed data blending mechanism.

Real-time crime prediction:

Both real-time data and historical data are used to create an analytical model for real-time crime prediction. Apache Flink is efficient in building real-time analytical models by using real-time data. In the proposed work real-time analytics process is designed for crime prediction using a machine learning model. The process is updated continuously using real-time data and historical data stored.

5.4 EXPERIMENTAL EVALUATION

The detailed workflow of an experimental setup for the real-time analytics for crime hotspot prediction is shown in Figure 5.2. The complete process is based on the different stages, as explained in the previous section. The real-time data sources that generate crime-related information are identified, and used for the experimental evaluation. Three different data sources are considered for analysis, like Twitter data, Facebook posts, and citizen complaints data through the mobile application. For each data source, a specific data ingestion processor is configured to ingest the data in real-time. The three different data sources use three different data ingestion processors to ingest the data from a particular source and filter the data stream if only related to crime incidents of a target location. The data streams passed from each of the processors are processed by respective data adapters. Each adapter comprised a mechanism to read the data stream as and when passed from the respective data ingestion processor. Each adapter is a real-time task to preprocess the data with the proposed data blending mechanism. The prepared data from all three adapters are stored on a common HBase table on top of Hadoop. This data is stored in HDFS to use in later stages as historical data. Here both historical data and real-time data streamed are used in the real-time analysis for crime hotspot prediction.

The primary objective of the proposed work is to use multiple data sources in real-time for crime hotspot prediction. Preparing the data as analytics-ready immediately as it gets ingested into the system is a challenging task in the process. The accuracy of the prediction results purely depends on the quality of the data used. The preprocessing stage is a critical step in the analytics process, where it takes the maximum duration of the entire process. Since data gets collected in real-time, the preprocessing to be done as and when data enters into the system and make it available as analytics-ready immediately. Here, the data from different sources are ingested as input data, containing different formats and structures. The proposed design consists mainly of three different stages: data collection phase with a data ingestion mechanism, data preprocessing phase with a data blending mechanism, and real-time analytics with a prediction model for crime hotspot prediction. The data ingestion mechanism is responsible for ingesting

5. Real-time analytics based crime prediction using multiple data sources

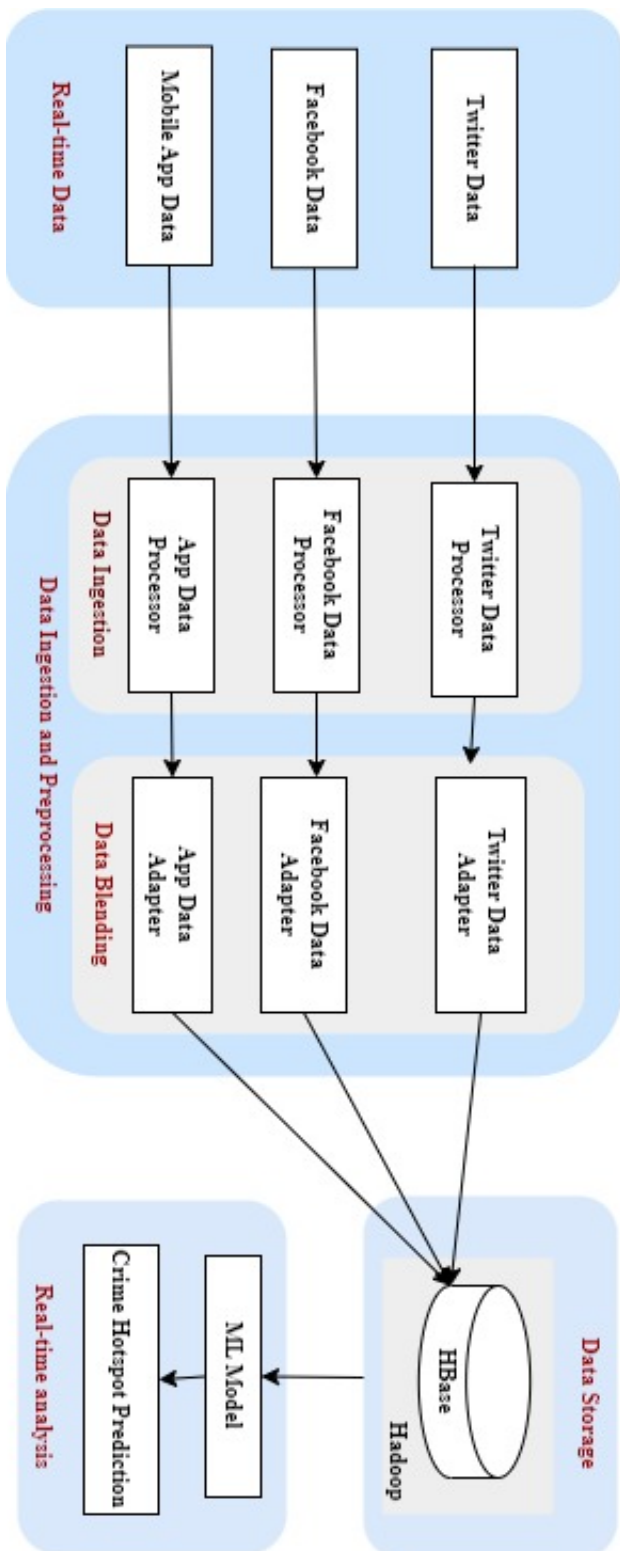


Figure 5.2: Experimental setup for real-time prediction of Crime hotspots from multi-source data

the data from the identified sources in the real-time and initial filtering process to ingest only the crime-related data of a specific location. The data blending mechanism involves preprocessing the data and integrating the data ingested from different sources into a common dataset as analysis ready. The prediction model uses the preprocessed data to predict the crime hotspot using the appropriate machine learning model.

5.4.1 Data Ingestion Processor

For the experimental work, real-time data ingested from three different data sources into the analytics process. Real-time data from Twitter posts, Facebook posts, and citizen-compliant data through mobile applications are used as input data. The proposed design uses separate data ingestion processors for each data source during the data ingestion mechanism. The working of the data ingestion mechanism from a specific data source is as shown in Figure 5.3. Each data ingestion processor responsible for ingesting the data in real-time from a specific data source and perform basic preprocessing of data to filter out the unwanted information. For example, the Twitter data ingestion processor responsible for ingesting the tweets as and when it gets generated at the source and filter to extract only the crime-related tweets of a specified location. Once the data streamed into the system in each of the data ingestion processors, it is verified that location values in the incoming data are within the range of given location values. If the location values match the specified values, the contents of the datastream are verified for having any crime-related information.



Figure 5.3: Work-flow of Data Ingestion Process

In the data ingestion phase, once the input data stream is successfully verified to match the location values, it is further verified for having any crime-related information. A knowledge base is used in the p that consists of the words and phrases related to

different categories of crimes, as explained in Chapter 3. The content of each incoming data stream is verified for any crime-related information present or not with the help of the knowledge base. If the incoming data stream is found to have crime-related content, then it is passed into the next preprocessing stage. The data does not match the specified location range and does not contain any crime-related information discarded directly during the data ingestion mechanism.

5.4.2 Data Blending Adapter

The data blending mechanism consists of separate data blending adapters for each data source. Each data blending adapter reads the data from the respective data ingestion processor and processes it to prepare a common dataset as analytics-ready. Data blending adapters are real-time tasks implemented using Apache Flink as a real-time processing tool. The input data ingested from all three data sources are in javascript object notation (JSON) format, but the structure of the data is different in each case. In general, both structure and format of the data can be different across various data sources. The data blending mechanism aims to integrate the data from multiple sources into a common dataset as analytics-ready.

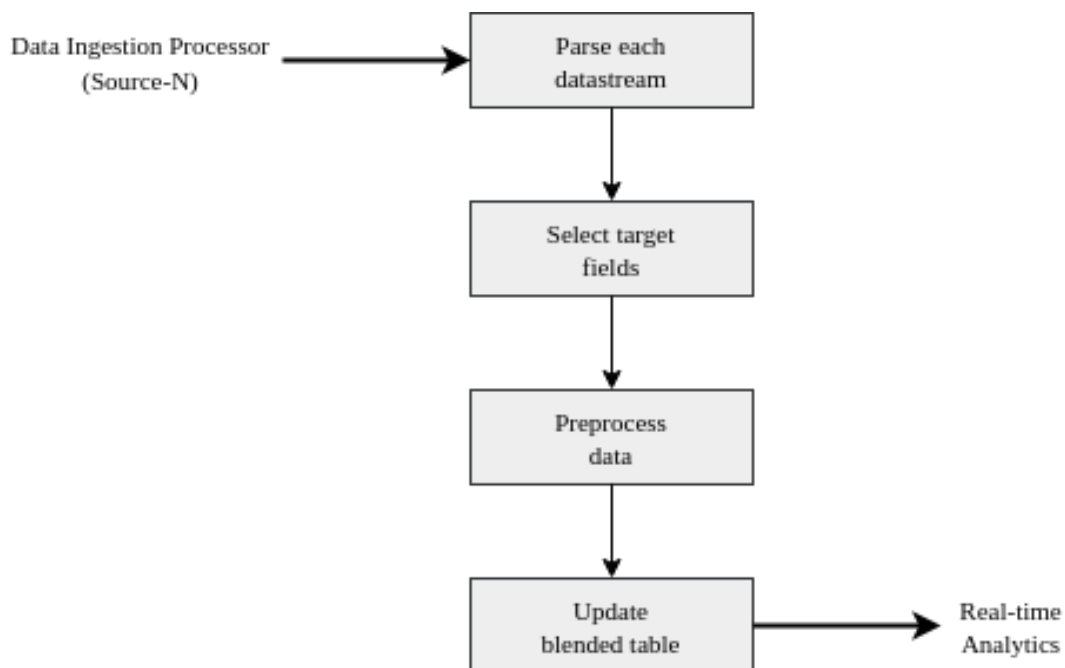


Figure 5.4: Work-flow of Data Blending Adapter

The working of a data blending adapter for each data source is as shown in Figure-5. Here, each adapter is configured to read the data stream passed from the respective data ingestion processor. The data blending adapter for each data source is written as a Flink job, capable of reading any new input data stream from the data ingestion processor immediately once available. The data stream gets parsed to filter the target fields; this information is helpful in the further analytics process. For the experimental work, the target fields selected are the time at data created, name or user information, location values, and text or message contents. The different data sources used here consist of these target information in a different structure. Hence, the data blending adapter for each data source is configured for filtering the target fields concerning the specific data source structure. The information filtered from each of the data sources is updated on a blended table in HBase. Additional information as a source-id number updated in each row of data helps identify the data source to which particular information belongs. The updated data on the blended table is further used in the real-time analytics engine for crime prediction.

5.4.3 Real-time analytics for crime prediction

The blended data updated on the Hbase table is used in further analysis to find a crime prediction solution. In the proposed work, crime prediction is a classification problem to predict a location will be a crime or not. The classification model is developed where the training data must consist of both positive and negative samples. Here positive data represents the crime scene, whereas negative samples represent non-crime scenes. The proposed work target is to use both historical data and real-time data for the prediction. At the initial stage, historical data are used, which represents only positive data samples. Hence, the sampling method proposed by (Gerber 2014) is used to add negative data samples. This method generates evenly spaced locations that do not coincide with the positive samples at a particular sampling granularity as negative samples.

In the proposed work, the crime prediction model is evaluated based on future crime data instead of using a cross-validation method. As stated by (Bogomolov et al. 2014), the cross-validation technique performs the repeated trials on historical data to remove

the fluctuation in the real-time data. Hence a testing period is defined in which the model is assumed to be valid. Here training data is created using the crime data before time t and similarly create testing data using the crime data within $[t, t+p]$ where p is the testing period. The values used for the testing period p are 8-hour, one day, one week, and one month. The prediction model performance is evaluated by using different p values to select an appropriate p value.

As the proposed crime prediction model is a classification problem, the most popularly used classification algorithms, such as Logistic Regression (LR), Naive Bayes (NB) classifier, Random Forest (RF), and Support Vector Machines (SVM) classifier, are used. Logistic Regression (LR) is a linear classifier that measures the relationship between the dependent and independent variables by determining the probabilities using a logistic function (Witten et al. 2011). NB classifier is a probabilistic classification algorithm based on the application of Bayes's theorem (John and Langley 1995). The model assumes that a specific feature's presence is unrelated to the presence of any other feature. Random Forest (RF) is based on the forest construction procedure where features as nodes grow like branches of a tree, finally combining all trees to form a Random Forest model (Breiman 2001). SVM classifier (Cortes and Vapnik 1995) is based on separating hyperplanes according to which new samples are classified. The performance of classification models is measured using three common metrics like accuracy, precision, and recall. Since the main objective of the proposed work on real-time analytics using multiple data sources, we limited the work with these four most commonly used classification algorithms in streaming data. Here NB classifier performs better than other classification algorithms used in the experiment.

5.5 RESULTS AND DISCUSSION

The performance of the proposed crime prediction system is evaluated using historical data and real-time data. The proposed system uses multiple data sources as input for crime prediction to achieve better performance. Three different real-time data sources are used for experimental work, along with the historical data. Figure 5.5 shows the

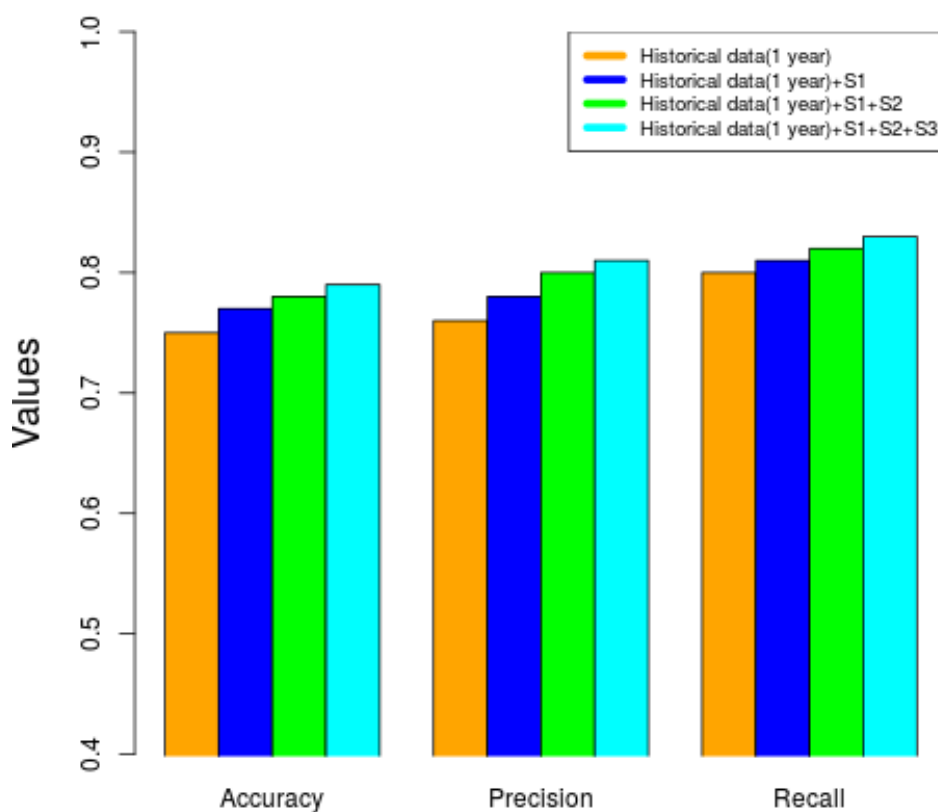


Figure 5.5: Performance Comparison between single source and blending of sources

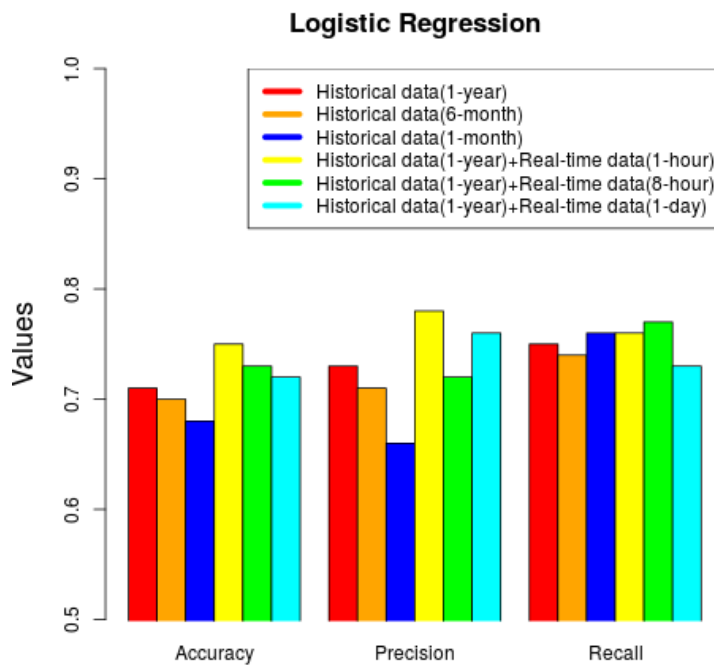
comparison of multiple data sources over a single data source for crime prediction. The results compare crime prediction model performance when different data sources are combined as input data. The prediction performance is compared with performance metrics such as accuracy, precision, and recall for a different set of input data. The performance metrics are compared by considering a different set of input data such as only historical data, historical data along with Twitter data(S1), historical data along with Twitter data and Facebook data(S2), historical data along with Twitter data, Facebook data, and mobile app data(S3). The result shows that using multiple data sources increases the performance of the crime prediction model.

Figure 5.6 and 5.7 shows a evaluation of different performance metrics of proposed crime prediction model with four different classification methods. The

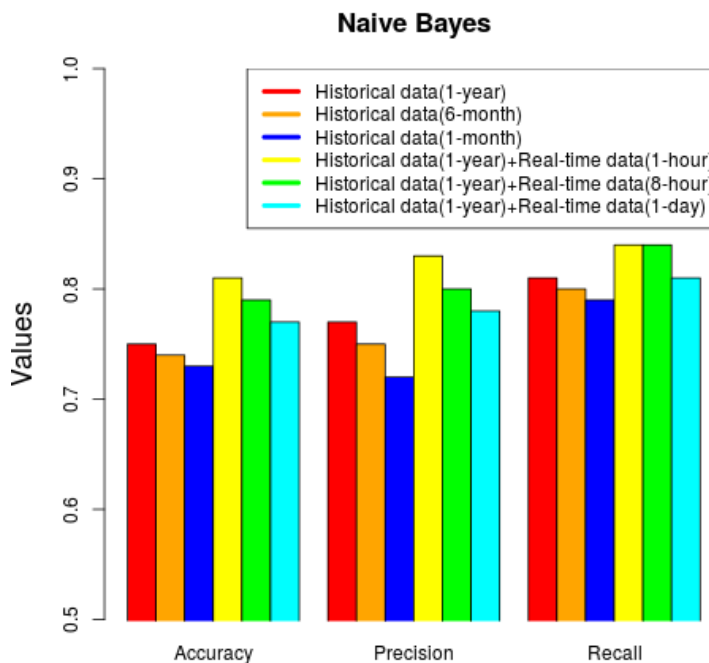
Table 5.1: Comparison of prediction performance of different classifiers with blended data

Data	Logistic Regression			Naive Bayes			SVM			Random Forest		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Historical data(1-year)	0.71	0.73	0.75	0.75	0.77	0.81	0.75	0.74	0.8	0.74	0.74	0.79
Historical data(6-month)	0.70	0.71	0.74	0.74	0.75	0.80	0.74	0.73	0.81	0.72	0.72	0.78
Historical data(1-month)	0.68	0.66	0.76	0.73	0.72	0.79	0.72	0.75	0.75	0.71	0.73	0.77
Historical data(1-year)+Real-time data(1-hour)	0.74	0.78	0.76	0.81	0.83	0.84	0.79	0.81	0.83	0.77	0.78	0.82
Historical data(1-year)+Real-time data(8-hour)	0.73	0.72	0.77	0.79	0.80	0.84	0.76	0.77	0.8	0.76	0.77	0.81
Historical data(1-year)+Real-time data(1-day)	0.72	0.76	0.73	0.77	0.78	0.81	0.76	0.76	0.79	0.75	0.76	0.80

results compare prediction model performances with historical data of different time intervals and historical data combined with real-time data at different time intervals. Table 5.1 shows the comparison of performance metrics for four different classification models using different combination of data. The prediction results indicate that historical data for one year gives better performance than the historical data used for six months and one-month duration. The results clearly show that performance improves with all the four different classification methods used in the experimental work. Hence, historical data for a one-year duration is used along with the real-time data in the proposed crime prediction model. The real-time data is the blended data from all the three data sources used. The model performs better when real-time data is used along with historical data.

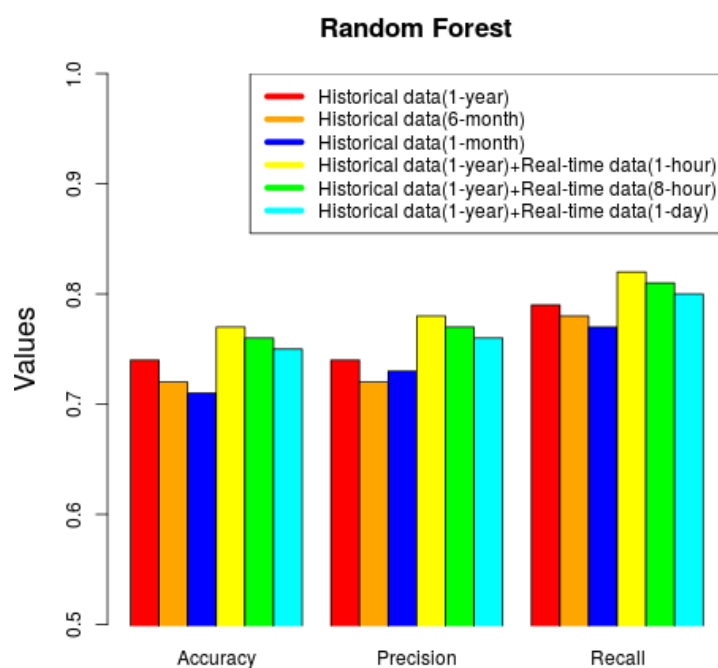


(a) Logistic Regression Classifier

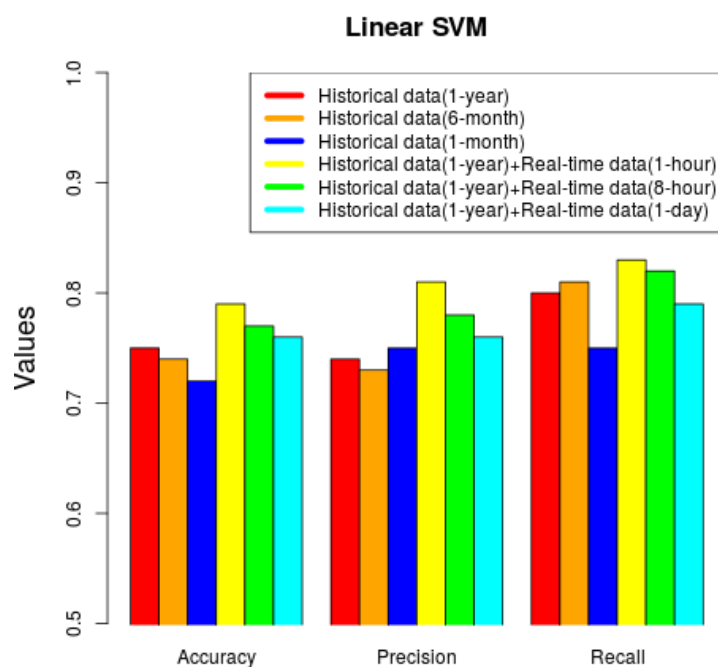


(b) Naive Bayes Classifier

Figure 5.6: Prediction performance of different classification algorithms with blended data (a) Logistic Regression and (b) Naive Bayes classifier

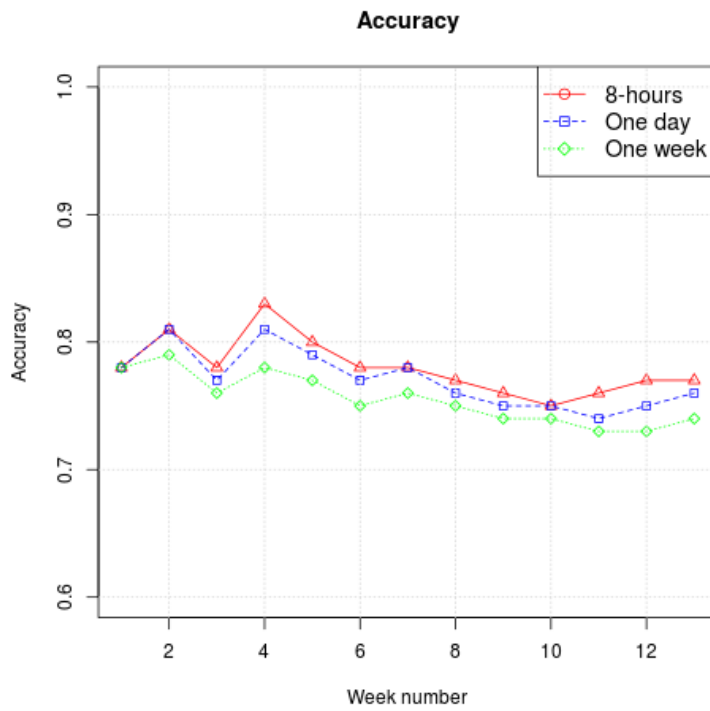


(a) Random Forest Classifier

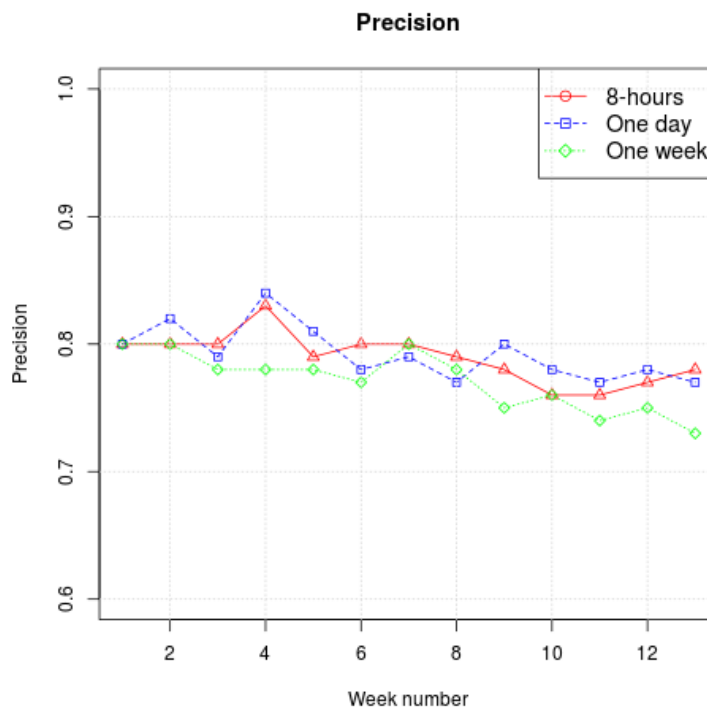


(b) Linear SVM Classifier

Figure 5.7: Prediction performance of different classification algorithms with blended data (a) Random Forest and (b) Linear SVM

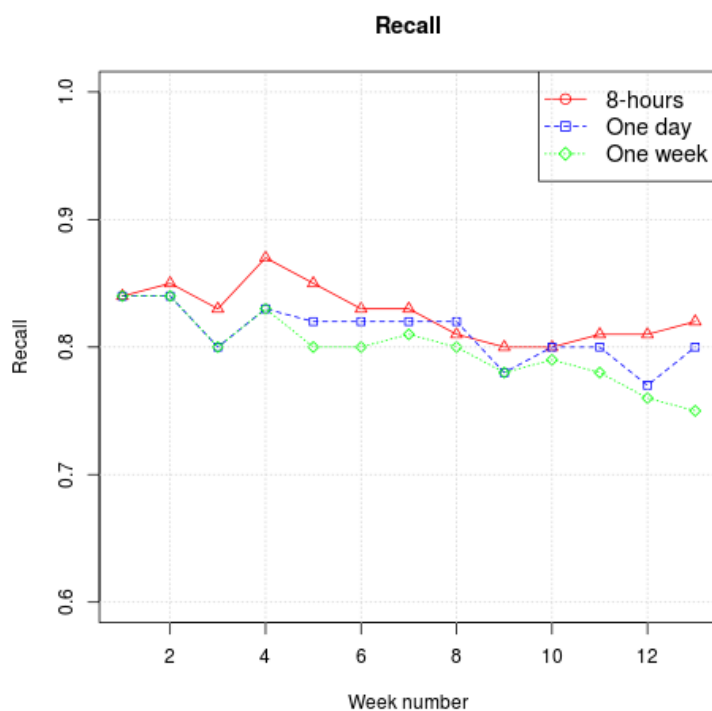


(a) Accuracy

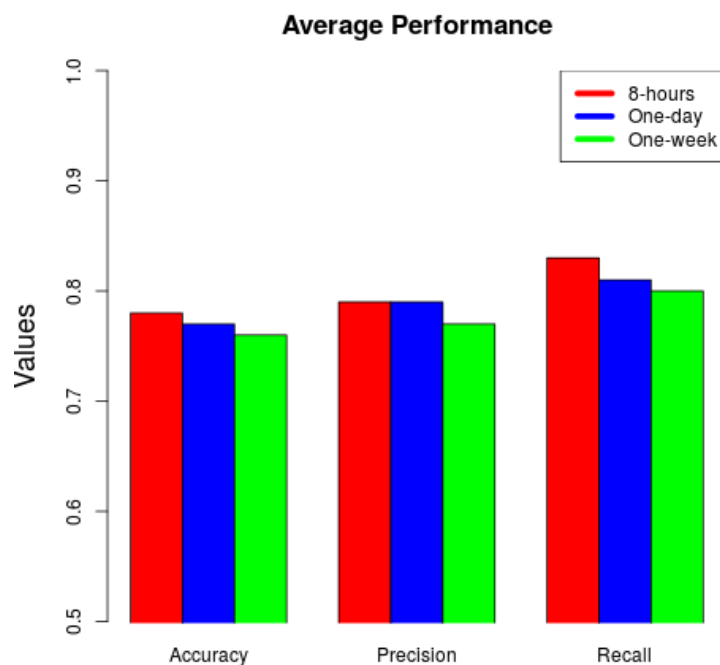


(b) Precision

Figure 5.8: Prediction performance comparison for different testing period- (a) Accuracy and (b) Precision



(a) Recall



(b) Average Performance

Figure 5.9: Prediction performance comparison for different testing period- (a) Recall and (b) Average Performance

The prediction model is tested with real-time data of different testing periods of one hour, eight hours, and one day, along with the historical data. The detailed comparison in the graph shows that real-time data of one-hour duration and the historical data of one-year duration perform better than the other input data combinations in all four different classification methods. Here NB classifier attains an accuracy of 81% and performs better than the other three classification techniques used.

The proposed crime prediction model performance is evaluated over time. During the experimental work, different time intervals(t) such as eight hours, one day, and one week are used to study the performance of the proposed model. Here, a new model is retained for every eight-hour, one day, and one week to compare the effectiveness of the model. For t being eight hours, the model is updated for each eight-hour duration to evaluate it and find the average results for each week. Similarly, for t being one-day, the model is updated every day to evaluate it and find the average results for each week. Finally, for t being one week, update the model every week and evaluate it individually in succeeding weeks. Figure 5.8 and 5.9 shows the performance of the proposed model over time. Here NB classifier is used to evaluate the model that gives better performance in the previous experiment. Figure 5.8(a), (b) and 5.9(a) show model performance metrics such as accuracy, precision, and recall, respectively. For the comparison of different time intervals, average performance over time is shown in Figure 5.9(b). When comparing the results for different t values, the model performance difference increases over time. It is observed that model performances are almost similar in the case of t is eight-hour and one days. Therefore t as one day is set to update the model instead of t as eight-hour to avoid computational expenses.

5.6 SUMMARY

In this chapter, a real-time analytics system using real-time data from multiple sources for crime hotspot prediction is proposed that improves the performance of the prediction model. In the proposed work, real-time data from multiple sources are used along with historical data to enhance the accuracy of the prediction model. This chapter discusses the proposed framework for a real-time crime prediction system that incorporates a real-

time data ingestion mechanism accompanied by a data blending approach for multiple data sources. The experimental work is carried out with the proposed framework using three different data sources for crime prediction. The data ingestion and data blending approach used in the proposed framework is flexible to add any more data sources of interest for real-time prediction. The results show that real-time data, along with the historical data, attains better performance. It is also tested with different time intervals to update the prediction model. Naive Bayes classification with 81% accuracy performs better than the other models used in the experiment.

As data generates continuously within the city, it can be analyzed to make timely decisions. Analyzing only the historical data is not sufficient for monitoring the crime for public safety. Real-time crime prediction is beneficial in the early detection and prevention of crime. It is also essential to use all possible data available from different sources to get more valuable insights into the prediction model. The proposed model is evaluated using both historical data and real-time data. It is observed that the prediction model attains better performance when multiple data sources are used. The experimental work shows that the model performs better when real-time data sources are used with historical data of one-year duration. The prediction model is compared with four different classification methods, which are popularly used with the streaming data. It also experiments with different time durations for prediction model updates with the real-time data sources. The results show that the model can be updated daily for better performance.

CHAPTER 6

CONCLUSIONS AND FUTURE SCOPE

6.1 CONCLUSION

Real-time data analytics plays a crucial role in numerous data-driven applications, enabling quick responses and actions. One significant application domain is public safety in smart city contexts, where timely decisions and predictions are vital for crime detection and prevention. In real-time analytics, the speed at which the entire analytics process is performed is of utmost importance. Data preprocessing, which prepares the data for analytics, typically consumes a substantial amount of time in any data analytics process. Consequently, when data is collected, it must be swiftly preprocessed and analyzed to facilitate desired analytical solutions without significant delays. Utilizing multiple data sources as input data in analytics can lead to improved and more accurate outcomes. The proposed data blending approach in this study is well-suited for real-time preprocessing of data from various sources. It offers flexibility in incorporating new data sources into the analytics framework with the same experimental setup, ensuring scalability and adaptability.

A real-time event processing mechanism is introduced to address emergency situations within the city, allowing for timely alerts and response to such incidents. By leveraging the proposed mechanism, analytical solutions like predictions and data-driven decisions can benefit from the use of all available data sources, rather than

relying solely on a single source. This holistic approach enhances the accuracy and effectiveness of the analytics process, leading to more reliable insights and actionable intelligence. The flexibility of the proposed mechanism enables the seamless integration of additional data sources into the analytics framework. As new data sources emerge or existing sources evolve, they can be effortlessly incorporated, ensuring the longevity and relevance of the analytics solution. This adaptability ensures that the system remains up-to-date and capable of leveraging the latest data for improved real-time analytics. The proposed data blending mechanism and real-time event processing contribute to efficient data preprocessing, accurate predictions, and informed decision-making. By utilizing multiple data sources and accommodating new sources seamlessly, the analytics process becomes more comprehensive and flexible, leading to enhanced outcomes and a proactive approach to public safety in real-time scenarios.

6.2 FUTURE SCOPE

In the future, there are several avenues for expanding and improving upon the proposed framework for real-time analytics using multiple data sources. The proposed framework can be extended to accommodate a larger number of data sources. By incorporating diverse sources, the framework can provide a more comprehensive and holistic view of the data, leading to more accurate insights and predictions. Currently, the proposed work focuses on text data ingested in the JSON format. Future work can explore the integration of data from various sources with different types of data formats and structures. This would require developing mechanisms to handle and preprocess data in formats such as CSV, XML, or relational databases. The data blending mechanism proposed in this work can be adapted and applied to other data-driven applications beyond the specific use case discussed. By making the mechanism more generic and flexible, it can be utilized in a wide range of domains where input data is sourced from multiple heterogeneous sources. The classification model developed for generating emergency event alerts can be further improved for increased accuracy. Future work can involve fine-tuning the model parameters, exploring ensemble methods, or considering advanced techniques such as deep

learning to enhance the performance of the classification model.

The proposed work compares four popular classification algorithms for streaming data. However, future work can involve evaluating the performance of additional algorithms to identify better-performing models. This exploration can help identify algorithms that are more suitable for specific types of data or exhibit improved accuracy in the real-time analytics context. With the proliferation of Internet of Things (IoT) devices, future work could explore the integration of edge computing capabilities with real-time big data analytics. This would involve developing distributed analytics frameworks that can process and analyze data at the edge of the network, closer to the data sources, to reduce latency and improve real-time decision-making.

By addressing these future research directions, the proposed framework can be enhanced to handle a wider range of data sources, formats, and structures. The inclusion of more diverse data, along with advancements in classification algorithms, will contribute to the accuracy and effectiveness of real-time analytics. Moreover, the generalization of the data blending mechanism will enable its application in various data-driven domains, extending the framework's usefulness and relevance.

BIBLIOGRAPHY

- Acar, A. and Muraki, Y. (2011). "Twitter for crisis communication: Lessons learned from japan's tsunami disaster." *Int. J. Web Based Communities*, 7(3), 392–402.
- Ai, F., Comfort, L. K., Dong, Y. and Znati, T. (2016). "A dynamic decision support system based on geographical information and mobile social networks: A model for tsunami risk mitigation in padang, indonesia." *Safety Science*, 90, 62–74.
- Alazawi, Z., Alani, O., Abduljabbar, M., Altowaijri, S. and Mehmood, R. (2014). "A smart disaster management system for future cities." *In Proceedings of the 2014 ACM international workshop on Wireless and mobile technologies for smart cities (WiMobCity '14)*, 1–10.
- Ali, M. I., Ono, N., Kaysar, M., Shamszaman, Z. U., Pham, T.-L., Gao, F., Griffin, K. and Mileo, A. (2017). "Real-time data analytics and event detection for iot-enabled communication systems." *Journal of Web Semantics*, 42, 19–37.
- Alves, L. G., Ribeiro, H. V. and Rodrigues, F. A. (2018). "Crime prediction through urban metrics and statistical learning." *Physica A: Statistical Mechanics and its Applications*, 505, 435 – 443.
- Andresen, M. (2009). "The place of environmental criminology within criminological thought." *Classics in environmental criminology*, 5–28.
- Atefeh, F. and Khreich, W. (2015). "A survey of techniques for event detection in twitter." *Comput. Intell.*, 31(1), 132–164.
- Baboshkin, P. and Uandykova, M. (2021). "Multi-source model of heterogeneous data

BIBLIOGRAPHY

- analysis for oil price forecasting.” *International Journal of Energy Economics and Policy*, 11, 384–391.
- Baccianella, S., Esuli, A. and Sebastiani, F. (2010). “SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining.” In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC’10)*, European Language Resources Association (ELRA), Valletta, Malta, 2200–2204.
- Bai, H. and Yu, G. (2016). “A weibo-based approach to disaster informatics: incidents monitor in post-disaster situation via weibo text negative sentiment analysis.” *Natural Hazards*, 83, 1177–1196.
- Blei, D. M., Ng, A. Y. and Jordan, I. (2003). “Latent dirichlet allocation.” *Journal of Machine Learning Research*, 993–1022.
- Bogomolov, A., Lepri, B., Staiano, J., Oliver, N., Pianesi, F. and Pentland, A. (2014). “Once upon a crime: Towards crime prediction from demographics and mobile data.” In *Proceedings of the 16th International Conference on Multimodal Interaction, ICMI ’14*, Association for Computing Machinery, New York, NY, USA, 427–434.
- Breiman, L. (2001). “Random forests.” *Machine Learning*, 5–32.
- Bueno, R., Fonseca, A., Gutiérrez, Y. and Montoyo, A. (2013). “Ssa-uo: Unsupervised sentiment analysis in twitter.” volume 2, 501–507.
- Caragea, C., McNeese, N., Jaiswal, A., Traylor, G., Kim, H., Mitra, P., Wu, D., Tapia, A., Giles, L., Jansen, B. and Yen, J. (2011). “Classifying text messages for the haiti earthquake.” In *8th International Conference on Information Systems for Crisis Response and Management, ISCRAM*, 1–10.
- Catlett, C., Cesario, E., Talia, D. and Vinci, A. (2019). “Spatio-temporal crime predictions in smart cities: A data-driven approach and experiments.” *Pervasive and Mobile Computing*, 53, 62–74.

- Cervone, G., Sava, E., Huang, Q., Schnebele, E., Harrison, J. and Waters, N. (2016). “Using twitter for tasking remote-sensing data collection and damage assessment: 2013 boulder flood case study.” *International Journal of Remote Sensing*, 37, 100–124.
- Cesario, E., Catlett, C. and Talia, D. (2016). “Forecasting crimes using autoregressive models.” In *2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing*, 795–802.
- Chainey, S., Tompson, L. and Uhlig, S. (2008). “The utility of hotspot mapping for predicting spatial patterns of crime.” *Security Journal*, 21, 4–28.
- Choi, S. and Bae, B. (2015). “The real-time monitoring system of social big data for disaster management.” In Park, J. J. J. H., Stojmenovic, I., Jeong, H. Y. and Yi, G., editors, *Computer Science and its Applications*, Springer Berlin Heidelberg, Berlin, Heidelberg, 809–815.
- Chourabi, H., Nam, T., Walker, S., Gil-Garcia, J. R., Mellouli, S., Nahon, K., Pardo, T. A. and Scholl, H. J. (2012). “Understanding smart cities: An integrative framework.” In *2012 45th Hawaii International Conference on System Sciences*, 2289–2297.
- Congosto, M., Basanta-Val, P. and MARK, L. S.-F. (2017). “T-hoarder: A framework to process twitter data streams.” In *Journal of Network and Computer Applications*, volume 83, 28–39.
- Cortes, C. and Vapnik, V. (1995). “Support-vector networks.” *Machine Learning*, 20(3), 273–297.
- Costa, C., Chatzimilioudis, G., Zeinalipour-Yazti, D. and Mokbel, M. F. (2017). “Towards real-time road traffic analytics using telco big data.” In *Proceedings of the International Workshop on Real-Time Business Intelligence and Analytics, BIRTE '17*, Association for Computing Machinery, New York, USA, 1–5.
- Costa, D. G. (2020). “Visual sensors hardware platforms: A review.” *IEEE Sensors Journal*, 20(8), 4025–4033.

BIBLIOGRAPHY

- Costa, D. G. and de Oliveira, F. P. (2020). “A prioritization approach for optimization of multiple concurrent sensing applications in smart cities.” *Future Generation Computer Systems*, 108, 228–243.
- Cresci, S., Tesconi, M., Cimino, A. and Dell’Orletta, F. (2015). “A linguistically-driven approach to cross-event damage assessment of natural disasters from social media messages.” In *Proceedings of the 24th International Conference on World Wide Web, WWW ’15 Companion*, Association for Computing Machinery, New York, NY, USA, 1195–1200.
- Davies, A. (2020). *IOT, Smart Technologies, Smart Policing: The Impact for Rural Communities*, 25–37.
- Dhapte, A. (2018). “Market research report.” Technical report, Market Research Future.
- Dirks, S., Gurdgiev, C. and Keeling, M. (May 2010). “Smarter cities for smarter growth.” In *IBM Global Business Services Executive Report*.
- Fan, F., Feng, Y., Yao, L. and Zhao, D. (2016). “Adaptive evolutionary filtering in real-time twitter stream.” In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, CIKM ’16*, Association for Computing Machinery, New York, NY, USA, 1079–1088.
- Flink (2017). “Apache flink.” <https://flink.apache.org/>). Accessed on Jan 10, 2017.
- Gao, H., Barbier, G. and Goolsby, R. (2011). “Harnessing the crowdsourcing power of social media for disaster relief.” *IEEE Intelligent Systems*, 26(3), 10–14.
- Garcia, S., Luengo, J. and Herrera, F. (2014). “Data preprocessing in data mining.” *Springer Publishing Company, Incorporated*, 39–57.
- Garcia, S., Ramirez-Gallego, S., Luengo, J., Benitez, J. M. and Herrera, F. (2016). “Big data preprocessing: methods and prospects.” *Big Data Analytics*, 1(1), 1–22.
- Gartner (2020). “Real-time analytics.”)Accessed on July 14, 2020.

- Gerber, M. S. (2014). "Predicting crime using twitter and kernel density estimation." *Decision Support Systems*, 61, 115 – 125.
- Ghosh, D., Chun, S. A., Shafiq, B. and Adam, N. R. (2016). "Big data-based smart city platform: Real-time crime analysis." In *Proceedings of the 17th International Digital Government Research Conference on Digital Government Research*, dg.o '16, ACM, New York, NY, USA, 58–66.
- Giffinger, R. and Gudrun, H. (2010). "Smart cities ranking: an effective instrument for the positioning of the cities?.." *ACE: Architecture, City and Environment*, 4, 7–26.
- Hadoop (2016). "Apache hadoop." <http://hadoop.apache.org/>). Accessed on Feb 19, 2016.
- Hardyns, W. and Rummens, A. (2018). "Predictive policing as a new tool for law enforcement? recent developments and challenges." *European Journal on Criminal Policy and Research*, 24, 201–218.
- Hartama, D., Mawengkang, H., Zarlis, M., Sembiring, R. W., Furqan, M., Abdullah, D. and Rahim, R. (2017). "A research framework of disaster traffic management to smart city." In *2017 Second International Conference on Informatics and Computing (ICIC)*, 1–5.
- Hasan, M., Orgun, M. A. and Schwitte, R. (2018). "Real-time event detection from the twitter data stream using the twitternews+ framework." In *Journal of Information Processing and Management*, volume 56, 1146–1165.
- Hopkin, G. (2023). "The rise of real-time data and event-streaming tech." Technical report, Datacenter.
- Hu, Y., Wang, F., Guin, C. and Zhu, H. (2018). "A spatio-temporal kernel density estimation framework for predictive crime hotspot mapping and evaluation." *Applied Geography*, 99, 89 – 97.

BIBLIOGRAPHY

- Huang, L., Liu, G., Chen, T., Yuan, H., Shi, P. and Miao, Y. (2021). “Similarity-based emergency event detection in social media.” *Journal of Safety Science and Resilience*, 2(1), 11–19.
- Huang, W. and Li, S. (2016). “Understanding human activity patterns based on space-time-semantics.” *ISPRS Journal of Photogrammetry and Remote Sensing*, 121, 1–10.
- Imran, M., Elbassuoni, S., Castillo, C., Diaz, F. and Meier, P. (2013). “Extracting information nuggets from disaster- related messages in social media.” In *ISCRAM 2013*, 1–10.
- Isafiade, O. E. and Bagula, A. B. (2017). “Fostering smart city development in developing nations: A crime series data analytics approach.” In *2017 ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K)*, 1–8.
- Ismail, A. (2016). “Utilizing big data analytics as a solution for smart cities.” In *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, 1–5.
- ISO/IEC-JTC (2015). “Smart cities.” Technical report, Information Technology.
- John, G. H. and Langley, P. (1995). “Estimating continuous distributions in bayesian classifiers.” In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, 338–345.
- Kafka (2017). “Apache kafka.” <https://kafka.apache.org/>). Accessed on Jan 05, 2017.
- Katsifodimos, A. and Schelter, S. (2016). “Apache flink: Stream analytics at scale.” In *2016 IEEE International Conference on Cloud Engineering Workshop (IC2EW)*, 193–193.
- Kennedy, L. and Dugato, M. (2018). “Forecasting crime and understanding its causes. applying risk terrain modeling worldwide.” *European Journal on Criminal Policy and Research*, 24, 1–6.
- Kreps, J. (2014). *Questioning the Lambda Architecture*, O’reilly.

- Landwehr, P. M., Wei, W., Kowalchuck, M. and Carley, K. M. (2016). “Using tweets to support disaster planning, warning and response.” *Safety Science*, 90, 33–47.
- Lau, B. P. L., Marakkalage, S. H., Zhou, Y., Hassan, N. U., Yuen, C., Zhang, M. and Tan, U.-X. (2019). “A survey of data fusion in smart city applications.” *Information Fusion*, 52, 357–374.
- Leonidas, A. (2017). *Understanding Smart Cities - A tool for Smart Government or an Industrial Trick?*, volume 1. 215-262.
- Li, Q., Chen, Y., Jiang, L. L., Li, P. and Chen, H. (2016). “A tensor-based information framework for predicting the stock market.” *ACM Transactions on Information Systems*, 34(2), 1–30.
- Li, R., Lei, K. H., Khadiwala, R. and Chang, K. C.-C. (2012). “Tedas: A twitter-based event detection and analysis system.” In *2012 IEEE 28th International Conference on Data Engineering*, 1273–1276.
- Liu, H. and Zhu, X. (2017). “Joint modeling of multiple crimes: A bayesian spatial approach.” *ISPRS International Journal of Geo-Information*, 6(1), 1–16.
- Mantoro, T., Feriadi, Agani, N., Ayu, M. A. and Jatikusumo, D. (2014). “Location-aware mobile crime information framework for fast tracking response to accidents and crimes in big cities.” In *2014 3rd International Conference on Advanced Computer Science Applications and Technologies*, 192–197.
- Marcu, O. C., Costan, A., Antoniu, G. and Perez-Hernandez, M. S. (2016). “Spark versus flink: Understanding performance in big data analytics frameworks.” In *2016 IEEE International Conference on Cluster Computing (CLUSTER)*, 433–442.
- Marz, N. and Warren, J. (2015). *Big Data: Principles and Best Practices of Scalable Realtime Data Systems*, Manning Publications Co., USA, 1st edition. 328.
- McMillan, D., Engström, A., Lampinen, A. and Brown, B. (2016). “Data and the city.” 2933–2944.

BIBLIOGRAPHY

- Meijer, A. and Wessels, M. (2019). “Predictive policing: Review of benefits and drawbacks.” *International Journal of Public Administration*, 42(12), 1031–1039.
- Miller, G. A. (1995). “Wordnet: A lexical database for english.” *Commun. ACM*, 38(11), 39–41.
- Mohler, G. O., Short, M. B., Brantingham, P. J., Schoenberg, F. P. and Tita, G. E. (2011). “Self-exciting point process modeling of crime.” *Journal of the American Statistical Association*, 106(493), 100–108.
- Nahri, M., Boulmakoul, A., Karim, L. and Lbath, A. (2018). “Iov distributed architecture for real-time traffic data analytics.” *Procedia Computer Science*, 130, 480–487.
- NiFi (2017). “Apache nifi.” <http://nifi.apache.org/>. Accessed on June 05, 2017.
- Nuaimi, E., Neyadi, H., Mohamed, N. and Al-Jaroodi, J. (2015). “Applications of big data to smart cities.” *Journal of Internet Services and Applications*, 6, 1–15.
- Ohyama, T. and Amemiya, M. (2018). “Applying crime prediction techniques to japan: A comparison between risk terrain modeling and other methods.” *European Journal on Criminal Policy and Research*, 24, 469–487.
- Olmezogullari, E. and Ari, I. (2013). “Online association rule mining over fast data.” In *Proceedings of the 2013 IEEE International Congress on Big Data, BIGDATA CONGRESS '13*, IEEE Computer Society, USA, 110–117.
- Parvez, M. R., Mosharraf, T. and Ali, M. E. (2016). “A novel approach to identify spatio-temporal crime pattern in dhaka city.” In *Proceedings of the Eighth International Conference on Information and Communication Technologies and Development, ICTD '16*, ACM, New York, NY, USA, 41:1–41:4.
- Peng, C.-Z., Jiang, Z.-J., Cai, X.-B. and Zhang, Z.-K. (2012). “Real-time analytics processing with mapreduce.” In *2012 International Conference on Machine Learning and Cybernetics*, volume 4, 1308–1311.

- Perry, W. L., McInnis, B., Price, C. C., Smith, S. and Hollywood, J. S. (2013). *Predictive Policing: The Role of Crime Forecasting in Law Enforcement Operations*, RAND Corporation. 136 55-110.
- Pina-Garcia, C. and Ramirez-Ramirez, L. (2019). “Exploring crime patterns in mexico city.” *Journal of Big Data*, 6(65), 1–21.
- Puentes, F., Pérez-Godoy, M., González, P. and Jesus, M. (2020). “An analysis of technological frameworks for data streams.” *Progress in Artificial Intelligence*, 9, 239–261.
- Puiu, D., Barnaghi, P., Tonjes, R., Kumper, D., Ali, M., Mileo, A., Parreira, J. X., Fischer, M., Kolozali, S., Farajidavar, N., Gao, F., Iggena, T., Pham, T. L., Nechifor, C. S., Puschmann, D. and Fernandes, J. (2016). “Citypulse: Large scale data analytics framework for smart cities.” *IEEE Access*, 4, 1086–1108.
- Ramirez-Gallego, S., Krawczyk, B., Garcia, S., Wozniak, M. and Herrera, F. (2017). “A survey on data preprocessing for data stream mining: Current status and future directions.” *Neurocomputing*, 239, 39–57.
- Ratcliffe, J. H. (2016). *Intelligence-Led Policing*, Taylor & Francis Group, 1-40.
- Rathore, Mazhar, M., Awais, A., Anand, P., Jiafu, W. and Daqiang, Z. (2016). “Real-time medical emergency response system: Exploiting iot and big data for public health.” *Journal of Medical Systems*, 40(12), 283 1–10.
- Rathore, M. M., Paul, A., Hong, W.-H., Seo, H., Awan, I. and Saeed, S. (2018). “Exploiting iot and big data analytics: Defining smart digital city using real-time urban data.” *Sustainable Cities and Society*, 40, 600–610.
- Reinsel, D., Gantz, J. and Rydning, J. (2017). “Data age 2025: The evolution of data to life-critical.” Technical report, Seagate.
- Report, M. (2022). “Market research report.” Technical report, Fortune Business Insights.

- Seonhwa, C. and Byunggul, B. (2015). *The Real-Time Monitoring System of Social Big Data for Disaster Management*, 809–815. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Sotsenko, A., Jansen, M., Milrad, M. and Rana, J. (2016). “Using a rich context model for real-time big data analytics in twitter.” In *2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*, 228–233.
- Ta, V.-D., Liu, C.-M. and Nkabinde, G. W. (2016). “Big data stream computing in healthcare real-time analytics.” In *2016 IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, 37–42.
- Taboada, M., Brooke, J., Tofiloski, M., Voll, K. and Stede, M. (2011). “Lexicon-Based Methods for Sentiment Analysis.” *Computational Linguistics*, 37(2), 267–307.
- Thelwall, M., Buckley, K. and Paltoglou, G. (2012). “Sentiment strength detection for the social web.” *J. Am. Soc. Inf. Sci. Technol.*, 63(1), 163–173.
- ToppiReddy, H. K. R., Saini, B. and Mahajan, G. (2018). “Crime prediction & monitoring framework based on spatial analysis.” *Procedia Computer Science*, 132, 696 – 705. International Conference on Computational Intelligence and Data Science.
- Toure, I. and Gangopadhyay, A. (2016). “Real time big data analytics for predicting terrorist incidents.” In *2016 IEEE Symposium on Technologies for Homeland Security (HST)*, 1–6.
- Venturini, L. and Baralis, E. (2016). “A spectral analysis of crimes in san francisco.” In *Proceedings of the 2Nd ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics*, UrbanGIS '16, ACM, New York, NY, USA, 4:1–4:4.
- Vieweg, S., Hughes, A., Starbird, K. and Palen, L. (2010). “Microblogging during two natural hazards events: What twitter may contribute to situational awareness.” volume 2, 1079–1088.

- Vilajosana, I., Llosa, J., Martínez, B., Domingo-Prieto, M., Angles, A. J. and Vilajosana, X. (2013). “Bootstrapping smart cities through a self-sustainable model based on big data flows.” *IEEE Communications Magazine*, 51, 128–134.
- Voskarides, N., Odijk, D., Tsagkias, M., Weerkamp, W. and de Rijke, M. (2014). “Query-dependent contextualization of streaming data.” In de Rijke, M., Kenter, T., de Vries, A. P., Zhai, C., de Jong, F., Radinsky, K. and Hofmann, K., editors, *Advances in Information Retrieval*, Springer International Publishing, Cham, 706–712.
- Vural, M. and Gök, M. (2017). “Criminal prediction using naive bayes theory.” *Neural Computing and Applications*, 28, 2581–2592.
- Wang, F., Hu, L., Zhou, D., Sun, R., Hu, J. and Zhao, K. (2015). “Estimating online vacancies in real-time road traffic monitoring with traffic sensor data stream.” *Ad Hoc Networks*, 35, 3 – 13. Special Issue on Big Data Inspired Data Sensing, Processing and Networking Technologies.
- Wessler, M. (2015). *Data Blending For Dummies*, 11-44, Alteryx.
- Williams, M. and Burnap, P. (2015). “Cyberhate on social media in the aftermath of woolwich: A case study in computational criminology and big data.” *British Journal of Criminology*, 56, 211–238.
- Witten, I., Frank, E. and Hall, M. (2011). “Data mining: Practical machine learning tools and techniques.” *3rd Edition*, Elsevier, 191–399.
- Xu, Z., Mei, L., Lv, Z., Hu, C., Luo, X., Zhang, H. and Liu, Y. (2019). “Multi-modal description of public safety events using surveillance and social media.” *IEEE Transactions on Big Data*, 5(4), 529–539.
- Yang, W., Liu, X., Zhang, L. and Yang, L. T. (2013). “Big data real-time processing based on storm.” In *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, 1784–1787.

- Yao, F. and Wang, Y. (2020). “Towards resilient and smart cities: A real-time urban analytical and geo-visual system for social media streaming data.” *Sustainable Cities and Society*, 63, 1–38.
- You, L., Tuncer, B. and Xing, H. (2019). “Harnessing multi-source data about public sentiments and activities for informed design.” *IEEE Transactions on Knowledge and Data Engineering*, 31(2), 343–356.
- Yu, J., Ding, W., Chen, P. and Morabito, M. (2014). “Crime forecasting using spatio-temporal pattern with ensemble learning.” volume 8444, 174–185.
- Zhang, C. and Yuan, D. (2015). “Fast fine-grained air quality index level prediction using random forest algorithm on cluster computing of spark.” In *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*, 929–934.
- Zhang, S., Zhang, C. and Yang, Q. (2003). “Data preparation for data mining.” *Applied Artificial Intelligence*, 17, 375–381.
- Zhou, Y., De, S. and Moessner, K. (2016). “Real world city event extraction from twitter data streams.” *Procedia Computer Science*, 98, 443–448.
- Zhuang, Y., Almeida, M., Morabito, M. and Ding, W. (2017). “Crime hot spot forecasting: A recurrent model with spatial and temporal information.” In *2017 IEEE International Conference on Big Knowledge (ICBK)*, 143–150.

PUBLICATIONS BASED ON THE RESEARCH WORK

1. Manjunatha and Annappa B. (2020). **Real time big data analytics framework with data blending approach for multiple data sources in Smart city applications.** *Scalable Computing: Practice and Experience*, 611-623. [DOI: [0.12694:/scpe.v21i4.1759](https://doi.org/10.12694/scpe.v21i4.1759)]
2. Manjunatha and Annappa B (2020). **Real-Time Emergency Event Detection System for Public Safety Using Multi-Source Data.** *International Journal of Advanced Science and Technology*, 29:344-351. [DOI: <http://sersc.org/journals/index.php/IJAST/article/view/7167>]
3. Manjunatha and B. Annappa (2018). **Real Time Big Data Analytics in Smart City Applications.** *International Conference on Communication, Computing and Internet of Things (IC3IoT)* [DOI: [10.1109/IC3IoT.2018.8668106](https://doi.org/10.1109/IC3IoT.2018.8668106)]
4. Manjunatha and B. Annappa. **Real-time big data analytics framework for crime prediction using multiple data sources in Smart city.** *KSII Transactions on Internet and Information Systems* [Under review]

BIO-DATA

Name: Manjunatha
Date of Birth: 28/07/1983
Gender: Male
Email Id: manjunatha.msh@gmail.com
Present Address: 2-115 Trasi House Holamage, Hakladi,
Kundapura, Udupi, Karnataka, India -
576235
Educational Qualifications: B.E (CSE) - Sri Jayachamarajendra College
of Engineering Mysore, Karnataka, India
M.Tech (Computer Science Engineering)
- NMAM Institute of Technology Nitte
Karkala, Udupi, India
Areas of Interest: Data Science, Big Data Analytics, Machine
Learning.