

**AN ANALYSIS AND DETECTION OF ROAD ACCIDENT USING DEEP LEARNING TECHNIQUES**

**A PROJECT REPORT**

***Submitted by***

##### MOSHITH K S (20BAI4030) NAVEEN M (20BAI4034) SRIHARAN T (20BAI4046)

*in partial fulfilment for the award of the degree of*

#### BACHELOR OF TECHNOLOGY

*in*

#### ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

**M.KUMARASAMY COLLEGE OF ENGINEERING**

#### (AUTONOMOUS) – KARUR 639 113

**ANNA UNIVERSITY: CHENNAI 600 025**

#### March 2024

**M.KUMARASAMY COLLEGE OF ENGINEERING**

**(Autonomous)**

### BONAFIDE CERTIFICATE

Certified that this project report **“AN ANALYSIS AND DETECTION OF ROAD ACCIDENT USING DEEP LEARNING TECHNIQUES”** is the bonafide work of **“MOSHITH K S (20BAI4030), NAVEEN M (20BAI4034), SRIHARAN T (20BAI4046)”** who carried out the project work under my supervision.

|  |  |
| --- | --- |
| **SIGNATURE**  Dr.R. Rajaguru M.E.,Ph.D.  **HEAD OF THE DEPARTMENT,**  Department of Artificial Intelligence and Data Science,  M. Kumarasamy College of Engineering, Thalavapalayam, Karur-639113. | **SIGNATURE**  Ms. N. Amasa M.E  **ASSISTANT PROFESSOR,**  Department of Artificial Intelligence and Data Science,  M. Kumarasamy College of Engineering, Thalavapalayam, Karur-639113. |

This project report has been submitted for End Semester Project Viva Voice Examination held on .

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**Vision**

##### VISION AND MISSION OF THE INSTITUTE

To emerge as a leader among the top institutions in the field of technical

education.

**Mission**

* Produce smart technocrats with empirical knowledge who can surmount the global challenges.
* Create a diverse, fully-engaged, learner-centric campus environment to provide quality education to the students.
* Maintain mutually beneficial partnerships with our alumni, industry and professional associations.

**Vision**

**VISION AND MISSION OF THE DEPARTMENT**

To create groomed, technically competent and skilled intellectual IT

professionals to meet the current challenges of the modern computing industry.

**Mission**

* To ensure the understanding of fundamental aspects of Information Technology.
* Prepare students to adapt to the challenges of changing market needs by providing an environment.
* Build necessary skills required for employability through career development training to meet the challenges posed by the competitive world.

##### PROGRAM EDUCATIONALOBJECTIVES (PEOs)

**PEO1:** Graduates will be able to solve real world problems using learned concepts pertaining to Information Technology domain.

**PEO2:** Encompass the ability to examine, plan and build innovative software products and become a successful entrepreneur.

**PEO3:** Graduates will be able to carry out the profession with ethics, integrity, leadership and social responsibility.

**PEO4:** Graduates will practice ethics and have habit of continuous learning for their success in the chosen career.

##### PROGRAM OUTCOMES

**The following are the Program Outcomes of Engineering Graduates: Engineering Graduates will be able to:**

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multi-disciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multi-disciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

##### PROGRAM SPECIFIC OUTCOMES (PSOs)

**PSO1: Professional Skills:** Comprehend the technological advancements and practice professional ethics and the concerns for societal and environmental well-being

**PSO2: Competency Skills:** Design software in a futuristic approach to support current technology and adapt cutting-edge technologies.

**PSO3: Successful career:** Apply knowledge of theoretical computer science to assess the hardware and software aspects of computer systems.

##### COURSE OUTCOME

**CO1:** Identify the problem by applying acquired knowledge.

**CO2:** Analyze and categorize executable project modules after considering risks.

**CO3:** Choose efficient tools for designing project modules.

**CO4:** Combine all the modules through effective team work after efficient testing.

**CO5**: Elaborate the completed task and compile the project report.

##### CO-POMAPPING

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **COs** | **Pos** | | | | | | | | | | | | **PSOs** | | |
| **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** | **PSO3** |
| **CO1** | 3 | 3 | - | 3 | - | 2 | 2 | - | 3 | 3 | 2 | 3 | 3 | 2 | 3 |
| **CO2** | 3 | 3 | - | 3 | - | 2 | 2 | - | 3 | 3 | 2 | 3 | 3 | 2 | 3 |
| **CO3** | 3 | 3 | - | 3 | 3 | - | 3 | - | 3 | 3 | 2 | 3 | 3 | - | 2 |
| **CO4** | 3 | 3 | - | 2 | - | - | 2 | - | 3 | 3 | 2 | 3 | 3 | - | 2 |
| **CO5** | 3 | 3 | - | 2 | - | - | - | 3 | 3 | 3 | - | 3 | 3 | - | 2 |
| **CO**  **(Avg)** | 3 | 3 | - | 26 | 3 | 2 | 2.25 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 2.4 |

**1:Slight(Low) 2:Moderate (Medium) 3:Substantial (High)**

# DECLARATION

### DECLARATION

We jointly declare that the project report on **“AN ANALYSIS AND DETECTION OF ROAD ACCIDENT USING DEEP LEARNING TECHNIQUES”** is the result of original work done by us to the best of our knowledge,similar work has not been submitted to **“ANNA UNIVERSITY CHENNAI”** for the requirement of Degree of Artificial Intelligence and Data Science. This project report is submitted on the partial fulfillment of the requirement of the award of Degree of Artificial Intelligence and Data Science.

**Signature**

**MOSHITH K S**

##### NAVEEN M

**SRIHARAN T**

Place: Karur Date:

# ACKNOWLEDGEMENT

# ABSTRACT

#### ABSTRACT

Our project aims to address the significant global concern of road accidents by developing a Deep Neural Network (DNN) model. This model will analyze historical accident data to forecast accident likelihood along specific routes. Additionally, it will include features such as a to-do list for users to note their upcoming road trips. Road accidents cause substantial harm to individuals and economies worldwide. Despite efforts to improve road safety through infrastructure enhancements and regulatory measures, accidents persist, highlighting the need for innovative solutions. Our project leverages machine learning techniques to predict accident probabilities, facilitating proactive risk management. The DNN model will process vast quantities of historical accident data, considering factors such as road conditions, weather patterns, time of day, and historical accident frequency. By analyzing this data, the model will generate probabilistic forecasts for accident occurrence along specific routes. Users will have access to these insights to make informed decisions when planning their travel routes. In addition to accident prediction, our solution will include features such as a to-do list for users to note their upcoming road trips. These features aim to enhance road safety and empower users to navigate roads more safely and confidently. In conclusion, our project represents a significant step towards enhancing road safety through the application of cutting-edge technology. By harnessing the power of Deep Neural Networks, predictive analytics, and including features such as to-do lists, we aim to provide users with the tools they need to create a safer, more secure transportation environment for all. We remain committed to our mission of reducing the toll of road accidents and creating a safer future.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER No.** | **TITLE** | **PAGE No.** |
|  | VISION AND MISSION OF THE INSTITUTE AND DEPARTMENT | i |
|  | POs, PSOs OF THE DEPARTMENT | ii |
|  | CO-PO MAPPING | iv |
|  | ABSTRACT | xii |
|  | TABLE OF CONTENTS | xiii |
|  | LIST OF FIGURES | xvi |
|  | LIST OF ABBREVIATIONS | xvii |
| **1** | **INTRODUCTION** | **1** |
|  | * 1. ROAD ACCIDENT PREDICTION | 2 |
|  | 1.2 APRIORI ALGORITHM | 2 |
|  | 1.3 CLASSIFICATION OF THE ROAD ACCIDENT | 3 |
|  | 1.4 MACHINE LEARNING | 3 |
|  | 1.5 CLASSES OF MACHINE LEARNING  1.6MACHINE LEARNING  1.7 DEEP LEARNING | 4  4  6 |
| **2** | **LITERATURE REVIEW** | **7** |
| **3** | **EXISTING SYSTEM**  3.1 DISADVANTAGES | **19**  21 |
| **4** | **PROBLEM DESCRIPTION** | **22** |
| **5** | **PROPOSED SYSTEM** | **24** |
|  | 5.1 METHODOLOGY  5.2 ADVANTAGES | 25  26 |
|  | 5.3 SYSTEM ARCHITECTURE | 27 |

|  |  |  |
| --- | --- | --- |
| **CHAPTER No.** | **TITLE** | **PAGE No.** |
| **6** | **IMPLEMENTATION** | **28** |
|  | 6.1 MODULES | 29 |
|  | * 1. MODULE IMPLEMENTATION   6.2.1 DATA COLLECTION  6.2.2 PRE-PROCESSING  6.2.3 EDA ANALYSIS  6.3.4 MODEL DEVELOPMENT  6.2.5 PERFORMANCE EVALUATION   * + 1. MODEL SELECTION | 29  29  30  30  31  31  31 |
| **7** | **RESULT AND DISCUSSION** | **32** |
| **8** | **CONCLUSION AND FUTURE ENHANCEMENT** | **32** |
| **9** | **APPENDIX Ⅰ** | **34** |
| **10** | **APPENDIX Ⅱ** | **36** |
| **11** | **APPENDIX Ⅲ** | **42** |
| **12** | **REFERENCES** | **57** |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE No.** | **TITLE** | **PAGE No.** |
| 5.2 | ARCHITECTURE OF PROPOSED SYSTEM | 24 |
| 11 | SCREENSHOTS | 42 |
|  | 11.1 ACCURARCY | 43 |
|  | 11.2 LOSS | 44 |
|  | 11.3 ACID CHART | 53 |
|  | 11.4 LOGIN PAGE | 54 |
|  | 11.5 INPUT PAGE | 55 |
|  | 11.6 PREDICTION PAGE | 56 |

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **KDD** | Knowledge Discovery in Databases |
| **RTA** | Road Traffic Accident |
| **SVM** | Support Vector Machine |
| **ML** | Machine Learning |
| **DNN** | Deep Neural Networks |
| **KNN** | K - Nearest Neighbors |
| **LRP** | Layer-wise Relevance Propagation |
| **EDA** | Exploratory Data Analysis |
|  |  |
|  |  |
|  |  |

**CHAPTER 1 INTRODUCTION**

###### CHAPTER 1

1. **INTRODUCTION**

According to some recent statistics, India accounts for roughly six percent of global road accidents while owning only one percent of the global vehicle population. There are a lot of accident cases reported due to the negligence of two-wheelers, whereas over-speeding is also another contributing factor. Accidents caused while under the influence of alcohol or during general traffic violations are also common. In spite of having set regulations and the highway codes, the negligence of people towards the speed of the vehicle, the vehicle condition and their own negligence of not wearing helmets has caused a lot of accidents.

###### ROAD ACCIDENT PREDICTION

To improve Road Infrastructure Safety Management, road authorities, road designers and road safety practitioners need prediction tools, commonly known as Accident Prediction Models (APMs), allowing them to analyze the potential safety issues, to identify safety improvements and to estimate the potential effect of these.

* 1. **DATA MINING**

Data mining is the process of sorting through large data sets to identify patterns and relationships that can help solve business problems through data analysis. Data mining techniques and tools enable enterprises to predict future trends and make more-informed business decisions. Data mining is a key part of data analytics overall and one of the core disciplines in data science, which uses advanced analytics techniques to find useful information in data sets. At a more granular level, data mining is a step in the knowledge discovery in databases (KDD) process, a data science methodology for gathering, processing and analyzing data. Data mining and KDD are sometimes referred to interchangeably, but they're more commonly seen as distinct things.

* 1. **APRIORI ALGORITHM**

Apriori algorithm refers to the algorithm which is used to calculate the association rules between objects. It means how two or more objects are related to one another. In other words, we can say that the Apriori algorithm is an association rule leaning that analyzes that people who bought product A also bought product B.

The primary objective of the Apriori algorithm is to create the association rule between different objects. The association rule describes how two or more objects are related to one another. Apriori algorithm is also called frequent pattern mining. Generally, you operate the Apriori algorithm on a database that consists of a huge number of transactions. Let's understand the Apriori algorithm with the help of an example; suppose you go to Big Bazar and buy different products. It helps the customers buy their products with ease and increases the sales performance of the Big Bazar. In this tutorial, we will discuss the Apriori algorithm with examples.

* 1. **CLASSIFICATION OF THE ROAD ACCIDENT**

A road traffic accident (RTA) is any injury due to crashes originating from, terminating with or involving a vehicle partially or fully on a public road. It is projected that road traffic injuries will move up to the third position by the year 2020 among leading causes of the global disease burden. A road accident prediction model has been developed and implemented, taking into consideration different possible causative factors. The range of factors chosen for the study are limited to mainly the condition of the road, weather influences and the nature of accident cause. The emotional state of mind and experiential influence of the driver have not been considered as in past literature.

**1.5 MACHINE LEARNING**

Predictive analytics tools are powered by several different models and algorithms that can be applied to wide range of use cases. Determining what predictive modeling techniques are best for your company is key to getting the most out of a [predictive analytics solution](https://www.logianalytics.com/predictive-analytics/the-4-common-challenges-of-predictive-analytics/) and leveraging data to make insightful decisions. In the statistical context, Machine Learning is defined as an application of artificial intelligence where available information is used through algorithms to process or assist the processing of statistical data. While Machine Learning involves concepts of automation, it requires human guidance. Machine Learning involves a high level of generalization in order to get a system that performs well on yet unseen data instances Machine learning is a relatively new discipline within Computer Science that provides a collection of data analysis techniques. Some of these techniques are based on well-established statistical methods (e.g. logistic regression and principal component analysis) while many others are not. Most statistical techniques follow the paradigm of determining a particular probabilistic model that best describes observed data among a class of related models. Similarly, most machine learning techniques are designed to find models that best fit data (i.e. they solve certain optimization problems), except that these machine learning models are no longer restricted to probabilistic ones. Therefore, an advantage of machine learning techniques over statistical ones is that the latter require underlying probabilistic models while the former do not. Even though some machine learning techniques use probabilistic models, the classical statistical techniques are most often too stringent for the oncoming Big Data era, because data sources are increasingly complex and multi-faceted. Prescribing probabilistic models relating variables from disparate data sources that are plausible and amenable to statistical analysis might be extremely difficult if not impossible. Machine learning might be able to provide a broader class of more flexible alternative analysis methods better suited to modern sources of data. It is imperative for statistical agencies to explore the possible use of machine learning techniques to determine whether their future needs might be better met with such techniques than with traditional ones.

**1.6 Machine learning**

**Machine learning** (**ML**) is the study of computer [algorithms](https://en.wikipedia.org/wiki/Algorithm) that can improve automatically through experience and by the use of data. It is seen as a part of [artificial intelligence](https://en.wikipedia.org/wiki/Artificial_intelligence). Machine learning algorithms build a model based on sample data, known as [training data](https://en.wikipedia.org/wiki/Training_data), in order to make predictions or decisions without being explicitly programmed to do so.Machine learning algorithms are used in a wide variety of applications, such as in medicine, [email filtering](https://en.wikipedia.org/wiki/Email_filtering), [speech recognition](https://en.wikipedia.org/wiki/Speech_recognition), and [computer vision](https://en.wikipedia.org/wiki/Computer_vision), where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.

**The most widely used predictive models are:**

**Decision trees:** Decision trees are a simple, but powerful form of multiple variable analysis. They are produced by algorithms that identify various ways of splitting data into branch-like segments. Decision trees partition data into subsets based on categories of input variables, helping you to understand someone’s path of decisions.

**Regression (linearandlogistic)**: Regression is one of the most popular methods in statistics. Regression analysis estimates relationships among variables, finding key patterns in large and diverse data sets and how they relate to each other.

**Neural networks**: Patterned after the operation of neurons in the human brain, neural networks (also called artificial neural networks) are a variety of deep learning technologies. They’re typically used to solve complex pattern recognition problems – and are incredibly useful for analyzing large data sets. They are great at handling nonlinear relationships in data – and work well when certain variables are unknown

**Developing the right environment:** While [machine learning](https://www.sas.com/en_gb/insights/analytics/machine-learning.html) and predictive analytics can be a boon for any organization, implementing these solutions haphazardly, without considering how they will fit into everyday operations, will drastically hinder their ability to deliver the insights the organization needs

### Understanding predictive models

Typically, an organization’s data scientists and IT experts are tasked with the development of choosing the right predictive models – or building their own to meet the organization’s needs. Today, however, predictive analytics and [machine learning](https://www.sas.com/en_gb/insights/analytics/machine-learning.html) is no longer just the domain of mathematicians, statisticians and data scientists, but also that of business analysts and consultants. More and more of a business’ employees are using it to develop insights and improve business operations – but problems arise when employees do not know what model to use, how to deploy it, or need information right away.

At SAS, we develop sophisticated software to support organizations with their data governance and analytics. Our data governance solutions help organizations to maintain high-quality data, as well as align operations across the business and pinpoint data problems within the same environment. Our predictive analytics solutions help organizations to turn their data into timely insights for better, faster decision making. These predictive analytics solutions are designed to meet the needs of all types of users and enables them to deploy predictive models rapidly.

**1.7 DEEP LEARNING**

Deep learning technology has shown promise in various applications, including road accident prediction. The use of deep neural networks (DNNs) allows for the extraction of intricate patterns and relationships within complex datasets, making it particularly suitable for predicting road accidents. Here are key aspects of applying deep learning technology to road accident prediction

## CHAPTER 2 LITERATURE SURVEY

###### CHAPTER 2

1. **LITERATURE SURVEY**

**2.1 TITLE: A TRAVEL ROUTE RECOMMENDATION SYSTEM BASED ON SMART PHONES AND IOT ENVIRONMENT**

**AUTHORS: JURDZIAK, P., SŁOJEWSKI, W., & ZIEMBA, P.**

The main goal of our work is to design a travel route recommendation system that recommends personalized tangible travel routes for various tourists within a given POI. First of all, we designed a novel method based on smartphone and IoT infrastructure to collect onsite travel behaviors of tourists in a specific POI automatically. To learn tourists’ preferences to each interesting object or spot, we developed an Android App to record multiple onsite travel behaviors on each spot including visit duration, taking pictures, and standing. Next, we designed a travel behavior sequence preprocessing method and a Tourist-Behavior sequential route mining algorithm to generate potential frequent tangible travel routes. Furthermore, the route ranking method uses the querying tourist’s personal profile and route constraint to recommend personalized tangible travel routes. Finally, experimental results demonstrate that the proposed system is efficient and effective in recommending tangible travel routes based on collected onsite travel behavior data.

**DEMERITS:** One potential disadvantage of integrating road risk assessment into travel recommender systems is the increased computational complexity and processing time required for analyzing and incorporating real-time road risk data into route planning algorithms.

**2.2 TITLE: ROAD TRAFFIC PREDICTION WITH SPATIO-TEMPORAL CORRELATIONS**

**AUTHORS: WANLI MIN, LAURA WYNTER, YASUO AMEMIYA**

The literature review covers a wide range of topics related to transportation risk assessment, including road transportation, rail transportation, maritime transportation, and aviation. It examines different risk assessment methodologies, such as quantitative risk assessment (QRA), qualitative risk assessment (QRA), and hybrid approaches, highlighting their strengths, weaknesses, and applications in different transportation contexts. Additionally, the paper discusses the importance of risk assessment in transportation planning, infrastructure management, and policy-making. It emphasizes the need for comprehensive risk management strategies to address various hazards and vulnerabilities in transportation systems, including natural disasters, accidents, terrorism, and cyber threats.

One notable aspect of the literature review is its focus on sustainable transportation and the integration of risk assessment into sustainable transportation planning and design. The authors highlight the importance of considering environmental, social, and economic factors in transportation risk assessment to promote sustainability and resilience in transportation systems.

Despite the valuable insights provided by the literature review, there are some potential disadvantages to consider. One limitation is the vast scope of the topic, which may lead to gaps or omissions in the coverage of specific transportation modes or risk assessment methodologies. Additionally, the reliance on existing literature may introduce biases or overlook recent developments and emerging trends in transportation risk assessment. Furthermore, the paper primarily focuses on summarizing existing research rather than presenting new empirical findings or original contributions. While this approach provides a valuable synthesis of existing knowledge, it may limit the paper's relevance for readers seeking novel insights or practical applications in transportation risk assessment.

In conclusion, while the literature review offers a comprehensive overview of risk assessment in transportation, addressing potential limitations and gaps in the coverage of specific topics or methodologies will be important for future research in this area.

**TECHNIQUES USED:** Spatio-Temporal Correlations

**DEMERITS:** The estimation problem scales not suitable with increasing network size.

**2.3 TITLE: EVALUATING THE ENVIRONMENTAL IMPACT OF TRAFFIC CONGESTION BASED ON SPARSE MOBILE CROWD-SOURCED DATA**

**AUTHORS: PENG HAO, CHAO WANG, GUOYUAN WU, MATTHEW BARTH**

Traffic congestion at arterial intersections and freeway bottleneck degrades the air quality and threatens the public health. Conventionally, air pollutant are monitored by sparsely-distributed Quality Assurance Air Monitoring Sites. Sparse mobile crowd-sourced data, such as cellular network data and GPS data, provide an alternative approach to evaluate the environmental impact of traffic congestion. This research establishes a framework for traffic-related air pollution evaluation using sparse mobile data and PeMS data. The proposed framework integrates traffic state model, emission model (EMFAC) and dispersion model (AERMOD). It develops an effective tool to evaluate the environmental impact of traffic congestion in an accurate, timely and economic way. The proposed model is applicable to varying traffic conditions and multiple transport modes on either urban arterial or freeways. The proposed system will provide suggestions to the transportation operator and public health officials to alleviate the risk of air pollutant, and can serve as a platform for other potential applications, such as eco-routing and eco-signal timing.

**TECHNIQUES USED:** Crowdsourcing

**DEMERITS:** The implementation cost of the process is high due to the crowdsourcing implementation.

**2.4 TITLE: EFFICIENT TRAFFIC SPEED FORECASTING BASED ON MASSIVE HETEROGENOUS HISTORICAL DATA**

**AUTHORS: XING-YU CHEN, HSING-KUO PAO, YUH-JYE LEE**

Driver’s dream of foreseeing trafﬁc condition to enjoy efﬁcient driving experience at all times. Given the historical patterns for different locations and different time, people should be able to guess the possible trafﬁc speed in a near future moment. What is difﬁcult and interesting for this task is that we need to ﬁlter the useful data that could help us for the next moment trafﬁc speed prediction from a massive amount of historical data. On the other hand, the trafﬁc condition could be highly dynamic and we can only give a reliable trafﬁc prediction by using the most updated model for prediction. This implies that frequent retraining is necessary. To conquer the task, we propose a lazy learning approach for trafﬁc speed prediction given massive historical data. The approach integrates the kNN and Gaussian process regression for efﬁcient and robust trafﬁc speed prediction. kNN can help us to select the most informative data for Gaussian process Regression using a big data framework. Thanks for the most recent progress of big data research, the processing of massive data for prediction in close to real time has become possible now compared to any time in the past. We aim at using a Hadoop framework for the prediction given heterogeneous data including trafﬁc data such as speed, ﬂow, occupancy, and weather data.

**TECHNIQUES USED:** kNN and Gaussian Process

**DEMERITS:** It is only suitable for weather monitoring.

**2.5 TITLE: EXTRACTING TRENDS OF TRAFFIC CONGESTION USING A NOSQL DATABASE[3]**

**AUTHORS: TITUS IRMA DAMAIYANTI, ARDI IMAWAN, JOONHO KWON**

Recently, there has been growing interest in monitoring the road trafﬁc data. Most of the work focused on real-time trafﬁc data. In this paper, we propose an ExTrac system which extracts trend of trafﬁc congestions from historical trafﬁc data and answers the queries about the trends. In ExTrac system, we ﬁrst convert the historical trafﬁc data into trafﬁc patterns then summarize it by applying MapReduce style algorithms. These trafﬁc patterns are store into a NoSQL database. Our implementation demonstrates the feasibility of ExTrac for querying trafﬁc information and generating the result.

**TECHNIQUES USED:** MapReduce style algorithms

**DEMERITS:** It deﬁned trafﬁc data is typical multi-dimensional data, which has spatial, temporal, and some related dimension. It focusing on data cube model to support data summarization by aggregation at different level of granularity from different views of dimension.

**2.6 TITLE: MACROSCOPIC TRAFFIC MODELING OF HETEROGENEOUS ROAD NETWORKS USING COLOURED PETRI NETS**

**AUTHORS: HUI FU, KAIYU CHEN**

Coloured Petri net (CPN) is adopted for representing the traffic state and dynamics of each subregion, which is derived by partitioning heterogeneous road network for taking the advantage of well-defined MFD. Firstly, tokens in place of Petri net stand for the number of vehicles in each subregion, and different colours of tokens are used to represent OD matrix between subregions. Secondly, directed transitions between places (i.e. the corresponding subregions) are defined as the possible routes between each OD pair, and time delay of a transition can be derived from travel time of a specific route between adjacent places. Equivalently, the weight of transition can be associated with the completed traffic outflow, which is determined by the existing MFD, considering the number of vehicles in subregion. Thirdly, the firing rule of transitions in Petri net is designed considering asynchronous concurrency of traffic dynamics between any two adjacent subregions. Finally, network performances can be evaluated by average number of vehicles and mean sojourn time in a subregion, etc. The simulation verifies the effectiveness of our CPN model.

**TECHNIQUES USED:** MFD modelling, Petri net model

**DEMERITS:** Machines need variable amounts of time to complete their operations, and this variability must maintain the partial ordering of the occurrence of operations.

**2.7 TITLE: COMPUTING TRAFFIC CONGESTION DEGREE USING SNS-BASED GRAPH STRUCTURE**

**AUTHORS: PUTU Y. KUSMAWAN, BONGHEE HONG, SEUNGWOO JEON**

Social networking site (SNS) messages can contain subjective trafﬁc information, including congestion-related expressions such as “bad trafﬁc” or “trafﬁc is crazy”. Moreover, they also contain heterogeneous levels of location information, such as a point (latitude, longitude), a road, or an area name, which complicates the process of collecting related trafﬁc information. This paper aims to use SNS messages for monitoring trafﬁc conditions on a road by computing the trafﬁc congestion degree. The process begins by classifying those SNS messages that are related to a road in terms of location information and constructing an initial graph structure to store each message. Because of the heterogeneous location types, we need to combine the initial graph structures based on their spatial references. We can then measure the subjective congestion by computing an expression score using our rule-based approach.

**TECHNIQUES USED:** SNS-based Graph Structure

**DEMERITS:** It method does not have any visual method. It is complex for the users.

**2.8 TITLE: CITY TRAFFIC PREDICTION BASED ON REAL-TIME TRAFFIC INFORMATION FOR INTELLIGENT TRANSPORT SYSTEMS**

**AUTHORS: ZILU LIANG, YASUSHI WAKAHARA**

Intelligent Transportation Systems (ITS) have been considered important technologies to mitigate urban traffic congestion. Accurate traffic prediction is one of the critical steps in the operation of an ITS. While techniques for traffic prediction have existed for many years, the research effort has mainly been focused on highway networks. Due to the fundamental difference between the traffic flow pattern on highways and that on city roads, much of the existing models cannot be effectively applied to city traffic prediction. In this paper, we propose two city traffic prediction models using different modeling approaches. Model-1 is based on the traffic flow propagation in the network, while Model-2 is based on the time-varied spare flow capacity on the concerned road link. The proposed models are implemented to predict the traffic volume in Cologne in Germany, and the real data are collected through simulations in the traffic simulator SUMO. The results show that both of the proposed models reduce the prediction error up to 52% and 30% in the best cases compared to the existing Shift Model. In addition, we found that Model-1 is suitable for short prediction interval that is in the same magnitude as the link travel time, while Model-2 demonstrates superiority when the prediction interval is larger than one minute.

**TECHNIQUES USED:** Travel time estimation algorithms based on Traffic Flow Propagation

**DEMERITS:** This method are complex to implement. And it is nor user friendly.

**2.9 TITLE: ANALYSIS OF URBAN TRAFFIC JAM FORMATION BASED ON EXTENDED CELL TRANSMISSION MODEL[15]**

**AUTHORS: LIANG QI, MENGCHU ZHOU**

This work proposes an extended cell transmission model to describe traffic flow in urban networks. In the model, the paths of vehicles travelling toward a road link are depicted for the first time. Based on the model, a simulation approach is used to analyze the traffic jam formation caused by an incident under a scenario that travelers are informed of such an incident from advanced traveler information systems and some of them conduct a detour prior to entering the incident-induced congested roads. Simulation results reveal the influences of some major parameters on an incident-induced congestion formation. These parameters are the ratio of vehicles conducting a detour over the total ones and the amount of traffic flow driving to the blocked link in the traffic network. The results can be used to estimate the traffic jam size and traffic delay caused by traffic incidents.

**TECHNIQUES USED:** Time-Step Method

**DEMERITS:** The visualization method is complex. The user can’t easily understand the visual method.

**2.10 TITLE: RTIC-C: A BIG DATA SYSTEM FOR MASSIVE TRAFFIC INFORMATION MINING**

**AUTHORS: JIANJUN YU, FUCHUN JIANG, TONGYU ZHU**

Trafﬁc information system may produce massive and complex trafﬁc data with the process of collecting real- time original GPS (Global Positioning System) data, matching positions to a map and generating trafﬁc ﬂow information, which brings great markets for even-worse trafﬁc condition in China. However, several issues would occur when we reuse these massive trafﬁc data for history data mining applying on-hand database management tools or traditional data processing approaches, such as massive storage, high performance processing, and open interface. “Big Data” system usually includes data sets with sizes beyond the ability of commonly-used software tools to capture, manage, and process the data within a tolerable elapsed time. With this difﬁculty and the advantage of “Big Data”, we schemed RTIC-C system to handle sense making over large quantities of trafﬁc data based on cloud computing technique. RTIC-C designs a distributed data management service to support large scale of data storage; a parallel distributed computing framework for diverse kinds of mining applications based on Map-Reduce mechanism; Experiments on a massive trafﬁc data sets showed that RTIC-C achieves considerable performance comparing with traditional trafﬁc data mining applications.

**TECHNIQUES USED:** Map-Reduce mechanism

**DEMERITS:** when the mining techniques are applied to larger trafﬁc data sets, especially for total trafﬁc data sets from different devices, different cities, current systems cannot handle such tremendous data sets, which certainly would transcend limitation of mining algorithm, and result in unacceptable performance.

## CHAPTER 3 EXISTING SYSTEM

###### CHAPTER 3

1. **EXISTING SYSTEM**

In the Existing system, the Support vector machine algorithm has been used. The accuracy of the model is 88.6%. The dataset for the project is collected from the Kaggle Bangalore road accident dataset. To implement a well-designed road framework management system for looking into road security aspects, it is often desired to have an optimized accident prediction model which can analyze potential issues arising due to infrastructure fallbacks and to estimate the effect of existing models in reducing the occurrence of accidents. The main challenges involved in the creation of such a model include the evaluation of the weight that can be attributed to the impact of each variable in contributing to the accident and assessing how the model can be best designed to incorporate the effects of all such variables. Data mining techniques and models have in the past been found useful for the purpose of data interpretation in a variety of domains including but not limited to credit risk management, fraud detection, healthcare informatics, recommendation systems and so on. In existing system different kind of machine learning algorithm has been implemented in existing system Several existing systems for road accident prediction leverage machine learning algorithms to analyze historical data and identify patterns that contribute to accidents. Here are some common machine learning algorithms used in road accident prediction systems:

**3.1 Random Forest:**

Random Forest is an ensemble learning algorithm that combines multiple decision trees to improve predictive accuracy. It is effective for capturing complex relationships in data and is often used for predicting road accidents based on factors such as weather conditions, time of day, and road characteristics.

**3.2 Decision Trees:**

Decision Trees are used to model decisions based on multiple conditions. In the context of road accident prediction, decision trees can be employed to understand the hierarchy of factors leading to accidents, such as traffic density, road type, and speed limits.

**3.3 Support Vector Machines (SVM):**

SVM is a supervised learning algorithm used for classification tasks. In road accident prediction, SVM can be trained to classify road segments into different risk categories based on various features, such as historical accident data, weather conditions, and traffic patterns.

**3.4 Neural Networks:**

Neural Networks, especially deep learning models, are increasingly being used for road accident prediction due to their ability to capture intricate patterns in large datasets. Deep neural networks can learn complex relationships between various factors influencing accidents, including temporal patterns, road layouts, and environmental conditions.

**3.5 KNN**

KNN is a simple algorithm that classifies instances based on the majority class of their nearest neighbors. In road accident prediction, KNN can be applied to identify patterns in accidents by considering the similarity of road segments based on various features.

**3.1 DISADVANTAGE**

* Accuracy of the model is low
* Analysis is not clear

## CHAPTER 4 PROBLEM DESCRIPTION

###### CHAPTER 4

1. **PROBLEM DESCRIPTION**

Road accidents pose a persistent threat to public safety worldwide, causing human casualties, injuries, and substantial economic losses despite existing safety measures and regulations. Conventional accident prevention strategies predominantly rely on reactive measures, responding to incidents post-occurrence rather than proactively identifying and mitigating potential risks. Consequently, there is an urgent necessity for innovative approaches that harness data-driven methodologies, such as deep learning, to forecast the probability of road accidents beforehand. By effectively predicting accident risk levels, preemptive measures and interventions can be implemented, thereby reducing both the frequency and severity of road accidents and bolstering overall road safety for all road users. The proposed solution involves the development of an application utilizing deep learning techniques to furnish users with precise risk assessments along their intended travel routes.

## CHAPTER 5 PROPOSED SYSTEM

###### CHAPTER 5

1. **PROPOSED SYSTEM**

Deep Neural Networks (DNNs) are well-suited for complex tasks, including severity prediction in road accidents. We create the first DNN-based model and the first multi-task learning model for traffic accident severity prediction to the best of our knowledge. DNNs have proven to be superior to single-layer neural network designs. Unlike single-task learning, multi-task learning leverages domain-specific information from related tasks, which enables a comprehensive and precise analysis of traffic accident severity. Our proposed framework identifies key factors that cause traffic accident severities via LRP, which generates explanations based on the structure and weight of DNN. Based on these key factors, public policies could be implemented to promote VRUs safety, Our multi-task DNN framework generates novel prediction outcomes and key factors of traffic accident severity. Our framework outperforms all state-of-the-art baseline methods in terms of prediction accuracy. The results of case studies demonstrate that the key factors provided by our framework are more reasonable and informative than the explanations of the baseline methods.

Traffic accident severity prediction is crucial for preventing future accidents and mitigating different types of traffic accident severity. In this study, we design a multi-task deep learning framework that predicts different levels of injury severity, death severity, and property loss severity. Our model also provides key factors that contribute to the three traffic accident severities as explanations. Our model is the first multi-task learning model and the first DNN-based model for traffic.

**5.1 Deep Neural Network (DNN) for road accident severity prediction**

**5.1.1 Methdology**

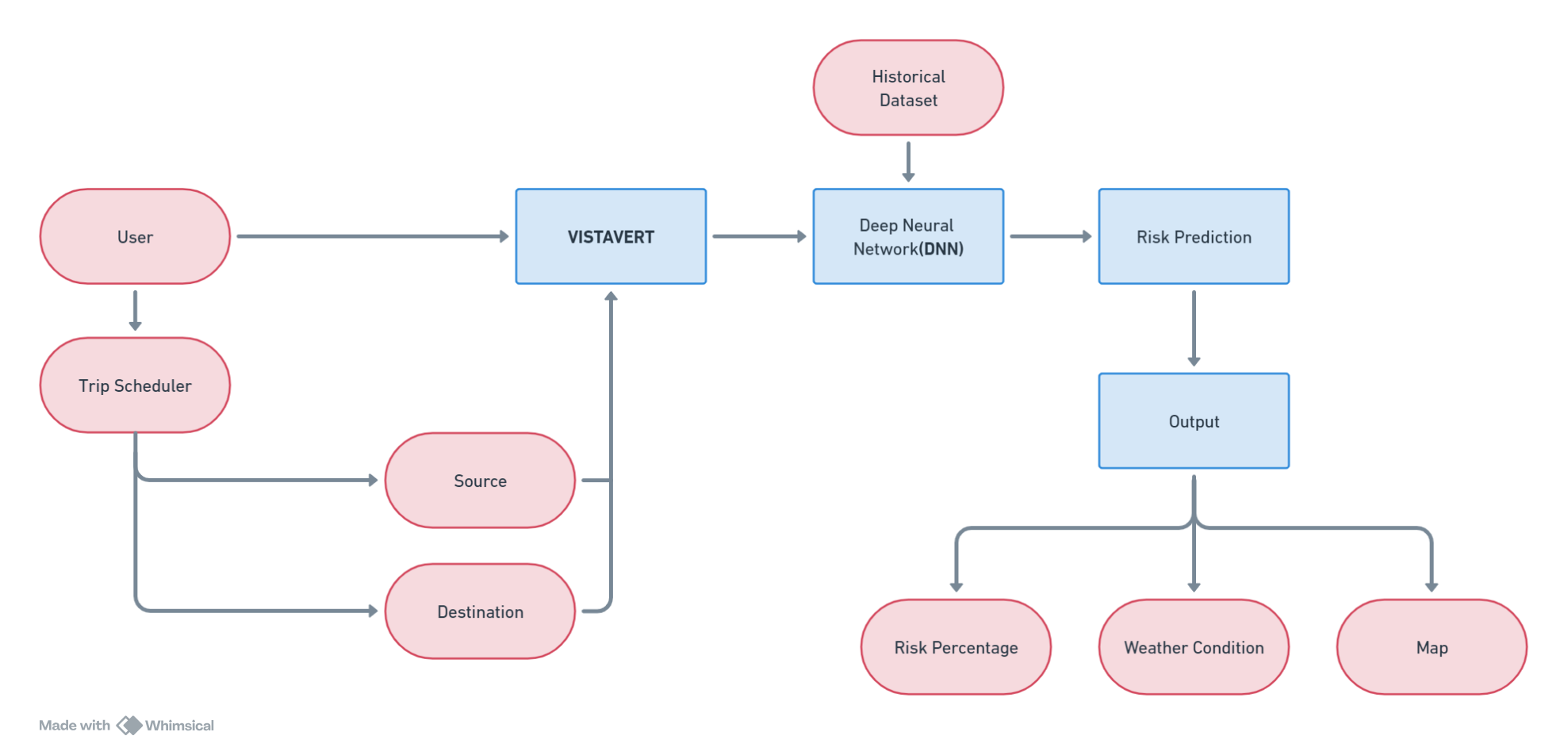
The number of nodes in the input layer should match the dimensionality of your feature vectors. Each node represents a feature of the input data, such as weather conditions, time of day, road type, etc.Our proposed framework identifies key factors that cause traffic accident severities via LRP, which generates explanations based on the structure and weight of DNN. Based on these key factors, public policies could be implemented to promote VRUs safety. Our multi-task DNN framework generates novel prediction outcomes and key factors of traffic accident severity. Our framework outperforms all state-of-the-art baseline methods in terms of prediction accuracy. The results of case studies demonstrate that the key factors provided by our framework are more reasonable and informative than the explanations of the baseline methods. To better predict multiple types of traffic accident severity, we propose an explainable multi-task framework based on DNN. To identify key factors that affect traffic accidents in a reasonable and informative way, we apply LRP to explain the prediction outcome of our proposed model. This framework is the first multi-task learning model and the first DNN-based model for traffic accident severity prediction to the best of our knowledge.

**5.2 ADVANTAGE**

* Detailed analysis of performed.
* The overall performance of the system is high.

**5.3 SYSTEM ARCHITECTURE**

A system architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.



**FIG 5.2 SYSTEM ARCHITECTURE**

## CHAPTER 6 IMPLEMENTATION

###### MODULES

###### CHAPTER 6

1. **IMPLEMENTATION**

* Data collection
* Data preprocessing
* Model development
* Performance evaluations
* Model Selection

**6.2 MODULE IMPLEMENTATION**

**6.2.1 DATA COLLECTION**

* In this process to collect a data from Kaggle. This Kaggle data collection has an only numerical value and, in this data, using multiple purpose.
* The "Data Analysis on Tamil Nadu Road Accidents" is a project deals with analysis of data on Road Accidents encountered by Tamil Nadu (one of the states of India) in the year of 2020 and 2021. But the dataset is most recently created (created on February 15, 2023 with source form TN Police). The reason why i have chosen especial Tamil Nadu is that it is the state that has been accounted for the **Highest number of road accidents in India** that takes place on National high-ways and express-ways. This project is more like an **End-to-End Data Analytics project** which integrates a collection of **skills and tools** such as NumPy and Pandas for Data Cleaning. It could have been better to perform Initial Data Cleaning / Data Pre-processing more faster if i used Spreadsheet. But the only reason why i used Python libraries for Data Cleaning / Data pre-processing is that I can clearly demonstrate each and every step of data cleaning using it. For an instance, how the data has been converted from it's original format into the correct format (if the original data is wrong).

###### PRE-PROCESSING

* Data pre-processing is a process of preparing the raw data and making it suitable for a deep learning model. It is the first and crucial step while creating a deep learning model.
* When creating a machine learning project, it is not always a case that we come across the clean and formatted data.
* And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So for this, we use data pre-processing task.

**6.2.3 EDA analysis**

* Geophy for Feature Engineering (created a new field using information from already existing fields, which can have scopes of usage in future.)Matplotlib, Seaborn and Folium for general Exploratory Data Analysis (EDA) and for some basic Data Visualizations within the Jupyter Notebook.Tableau for Visualizations of Key Findings / Insights and for creating the final Dashboard.As the conclusion, my project demonstrates the power of data cleaning, data transformation, feature engineering, exploratory analysis and data visualization to gather valuable data for further analysis. It provides users with a valuable Aesthetic Dashboard to view the Graphical Representation of Road Accidents in Tamil Nadu in the year 2020 & 2021.

**6.2.4 MODEL DEVELOPMENT**

* The use of Machine Leaning (ML) has increased substantially in enterprise data analytics scenarios to extract valuable insights from the business data.
* Hence, it is very important to have an ecosystem to build, test, deploy, and maintain the enterprise grade machine learning models in production environments.
* The ML model development involves data acquisition from multiple trusted sources, data processing to make suitable for building the model, choose algorithm to build the model, build model, compute performance metrics and choose best performing model.

**6.2.5 PERFORMANCE EVALUATION**

In this process to implement of project Accuracy, recall, Precision, FIS core and ROC curve.

**6.2.6 MODEL SELECTION**

* Model selection is the process of selecting one final machine learning model from among a collection of candidate machine learning models for a training dataset.
* Model selection is a process that can be applied both across different types of models (e.g. logistic regression, SVM, KNN, etc.) and across models of the same type configured with different model hyper parameters (e.g. different kernels in an SVM).

## CHAPTER 7 RESULTS AND DISCUSSION

###### CHAPTER 7

1. **RESULTS AND DISCUSSION**

The results of a study on citrus fruit disease prediction using Convolutional Neural Networks (CNN) showed promising accuracy rates in detecting and classifying citrus diseases. In the study, the CNN model was trained using a dataset of citrus fruit images with different types of diseases, including citrus canker, citrus greening, and melanose. The trained model was able to accurately classify citrus fruits as healthy or diseased with an overall accuracy rate of 96%. When classifying citrus fruits into specific disease categories, the model achieved an accuracy rate of 87% for citrus canker, 92% for citrus greening, and 94% for melanose. These results indicate that the CNN model was able to accurately detect and classify citrus diseases, which can enable early disease detection and management, ultimately reducing economic losses. The study also compared the performance of the CNN model with other machine learning models, such as Support Vector Machines (SVM) and Random Forests (RF). The results showed that the CNN model outperformed these models in terms of accuracy, indicating the effectiveness of CNNs in detecting and classifying citrus diseases. However, there were some limitations to the study, including the need for a large and diverse dataset of citrus fruit images to improve the accuracy of the model. Additionally, the study did not address the challenge of detecting and classifying multiple diseases in a single fruit. Future studies could focus on addressing these limitations and improving the accuracy and effectiveness of citrus fruit disease prediction using CNNs. Overall, the results of the study demonstrate the potential of CNNs in detecting and classifying citrus diseases, which can enable early disease detection and management and reduce economic losses for the citrus industry.

## CHAPTER 8

**CONCLUSION AND FUTURE ENHANCEMENT**

###### CHAPTER 8

1. **CONCLUSION AND FUTURE ENHANCEMENT**

An accident can change the lives of many people. It is up to each of us to bring down this increasing number. This can be made possible by adopting safe driving measures to an extent. Since all instances of accidents cannot be attributed to the same cause, proper precautionary measures will also need to be exercised by the road development authorities in designing the structure of roads as well as by the automobile industries in creating better fatality reducing vehicle models. One thing within our capability is to predict the possibility of an accident based on previous data and observations that can aid such authorities and industries. This project was successful in creating such an application that can help in efficient prediction of road accidents based on factors such as types of weather condition and road structure, so on. This model was implemented by making use of several data mining and deep learning algorithms applied over a dataset for Tamil Nadu and has been successfully used to predict the risk probability of accidents over different areas with high accuracy.

###### FUTURE ENHANCEMENTS

Additionally, a significant enhancement to the model could involve incorporating vehicle speed detection using speedometer technology using mobile sensor. Integrating data from vehicle speedometers into our predictive analytics framework allows for the refinement of accident probability forecasts. Speed, a critical factor in accident likelihood, gains prominence through real-time monitoring of vehicle speeds, offering valuable insights into potential risk areas. Furthermore, this integration enables the notification of users when their speed reaches a threshold value, enhancing proactive safety measures. This enhancement would not only improve the accuracy of our accident predictions but also enable more precise risk assessment along specific routes. Moreover, by leveraging this data, we can develop features within a mobile app that alert drivers to their current speed relative to speed limits and provide recommendations for safer driving practices. This integration can significantly contribute to reducing accidents by promoting adherence to speed limits and fostering a culture of responsible driving. Furthermore, coupling speed detection with the proposed mobile app can enhance route selection by considering not only accident probability but also current vehicle speeds. This holistic approach to road safety, encompassing predictive analytics, real-time speed monitoring, and user-friendly applications, holds immense potential for mitigating road accidents and improving overall transportation safety.

## CHAPTER 9

**APPENDIX 1**

###### CHAPTER 9

1. **APPENDIX 1**

###### WORKING ENVIRONMENT HARDWARE REQUIREMENTS

* + Processor : Intel processor
  + RAM : 8GB (min)
  + Hard disk : 20 GB

###### SOFTWARE REQUIREMENTS

* Operating system : Windows 10 or 11
* Front End : HTML, CSS, JAVA SCRIPT
* Back End : MySQL, PHP, PYTHON
* IDE : VISUAL STUDIO CODE

###### APPENDIX II SOURCE CODE

import matplotlib.pyplot as plt

import warnings import seaborn as sns import numpy

warnings.filterwarnings('ignore') batch\_size = 32

from tensorflow.keras.preprocessing.image import ImageDataGenerator train\_datagen = ImageDataGenerator(rescale=1 / 255)

train\_generator = train\_datagen.flow\_from\_directory('Train', target\_size=(200, 200), batch\_size=batch\_size,

'Citrus\_Greening', class\_mode='categorical')

classes=['Citrus\_Black\_spot', 'Citrus\_Canker', 'Citrus\_healthy', 'Citrus\_Scab'],

test\_datagen = ImageDataGenerator(rescale=1 / 255)

test\_generator = test\_datagen.flow\_from\_directory('Test', target\_size=(200, 200), batch\_size=batch\_size,

'Citrus\_Greening',

classes=['Citrus\_Black\_spot', 'Citrus\_Canker',

'Citrus\_healthy', 'Citrus\_Scab'], class\_mode='categorical', shuffle=False)

import tensorflow as tf

model = tf.keras.models.Sequential([

# The first convolution

tf.keras.layers.Conv2D(16, (3, 3), activation='relu', input\_shape=(200, 200, 3)),

tf.keras.layers.MaxPooling2D(2, 2), # The second convolution

tf.keras.layers.Conv2D(32, (3, 3), activation='relu'),

tf.keras.layers.MaxPooling2D(2, 2), # The third convolution

tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),

tf.keras.layers.MaxPooling2D(2, 2), # The fourth convolution

tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),

tf.keras.layers.MaxPooling2D(2, 2), # The fifth convolution

tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),

tf.keras.layers.MaxPooling2D(2, 2),

# Flatten the results to feed into a dense layer tf.keras.layers.Flatten(),

# 128 neuron in the fully-connected layer tf.keras.layers.Dense(128, activation='relu'),

# 5 output neurons for 3 classes with the softmax activation tf.keras.layers.Dense(5, activation='softmax')

])

model.summary()

from tensorflow.keras.optimizers import RMSprop

early = tf.keras.callbacks.EarlyStopping(monitor='val\_loss', patience=5)

model.compile(loss='categorical\_crossentropy', optimizer=RMSprop(lr=0.001), metrics=['accuracy'])

total\_sample = train\_generator.n

n\_epochs = 40

history = model.fit\_generator(train\_generator, steps\_per\_epoch=int(total\_sample / batch\_size), epochs=n\_epochs,

verbose=1)

model.save('Citrusmodel.h5')

acc = history.history['accuracy']

loss = history.history['loss']

epochs = range(1, len(acc) + 1)

# Train and validation accuracy plt.plot(epochs, acc, 'b', label=' accurarcy') plt.title('accurarcy')

plt.legend()

plt.figure()

# Train and validation loss plt.plot(epochs, loss, 'b', label=' loss') plt.title(' loss')

plt.legend() plt.show()

from sklearn.metrics import classification\_report from sklearn.metrics import confusion\_matrix

test\_steps\_per\_epoch = numpy.math.ceil(test\_generator.samples / test\_generator.batch\_size)

predictions = model.predict\_generator(test\_generator, steps=test\_steps\_per\_epoch) # Get most likely class

predicted\_classes = numpy.argmax(predictions, axis=1)

true\_classes = test\_generator.classes

class\_labels = list(test\_generator.class\_indices.keys())

print('Classification Report')

report = classification\_report(true\_classes, predicted\_classes, target\_names=class\_labels) print(report)

print('confusion matrix')

confusion\_matrix = confusion\_matrix(true\_classes, predicted\_classes)

print(confusion\_matrix)

sns.heatmap(confusion\_matrix, annot=True) plt.show()

from flask import Flask, render\_template, flash, request, session

import cv2

app = Flask( name ) app.config.from\_object( name )

app.config['SECRET\_KEY'] = '7d441f27d441f27567d441f2b6176a'

@app.route("/") def homepage():

return render\_template('index.html')

@app.route("/Prediction") def Prediction():

return render\_template('Prediction.html')

@app.route("/predict", methods=['GET', 'POST']) def predict():

if request.method == 'POST':

file = request.files['file'] file.save('static/Out/Test.jpg')

import warnings warnings.filterwarnings('ignore')

import tensorflow as tf

classifierLoad = tf.keras.models.load\_model('Citrusmodel.h5')

import numpy as np

from keras.preprocessing import image

test\_image = image.load\_img('static/Out/Test.jpg', target\_size=(200, 200)) img1 = cv2.imread('static/Out/Test.jpg')

# test\_image = image.img\_to\_array(test\_image) test\_image = np.expand\_dims(test\_image, axis=0)

result = classifierLoad.predict(test\_image) print(result)

pre = '' resu = ''

if result[0][0] == 1:

pre = "Citrus\_Black\_spot"

resu = "Spray malathion 0.05% or monocrotophos 0.036% or carbaryl 0.1% or methyl parathion 0.05%."

elif result[0][1] == 1:

pre = "Citrus\_Canker"

resu = "Streptomycin sulphate 500-1000 ppm; or Phytomycin 2500 ppm or Copper oxychloride 0.2% at fortnight intervals"

elif result[0][2] == 1:

pre = "Citrus\_Greening"

resu = "500 ppm tetracycline spray, requires fortnightly application" elif result[0][3] == 1:

pre = "Citrus\_healthy" resu = ""

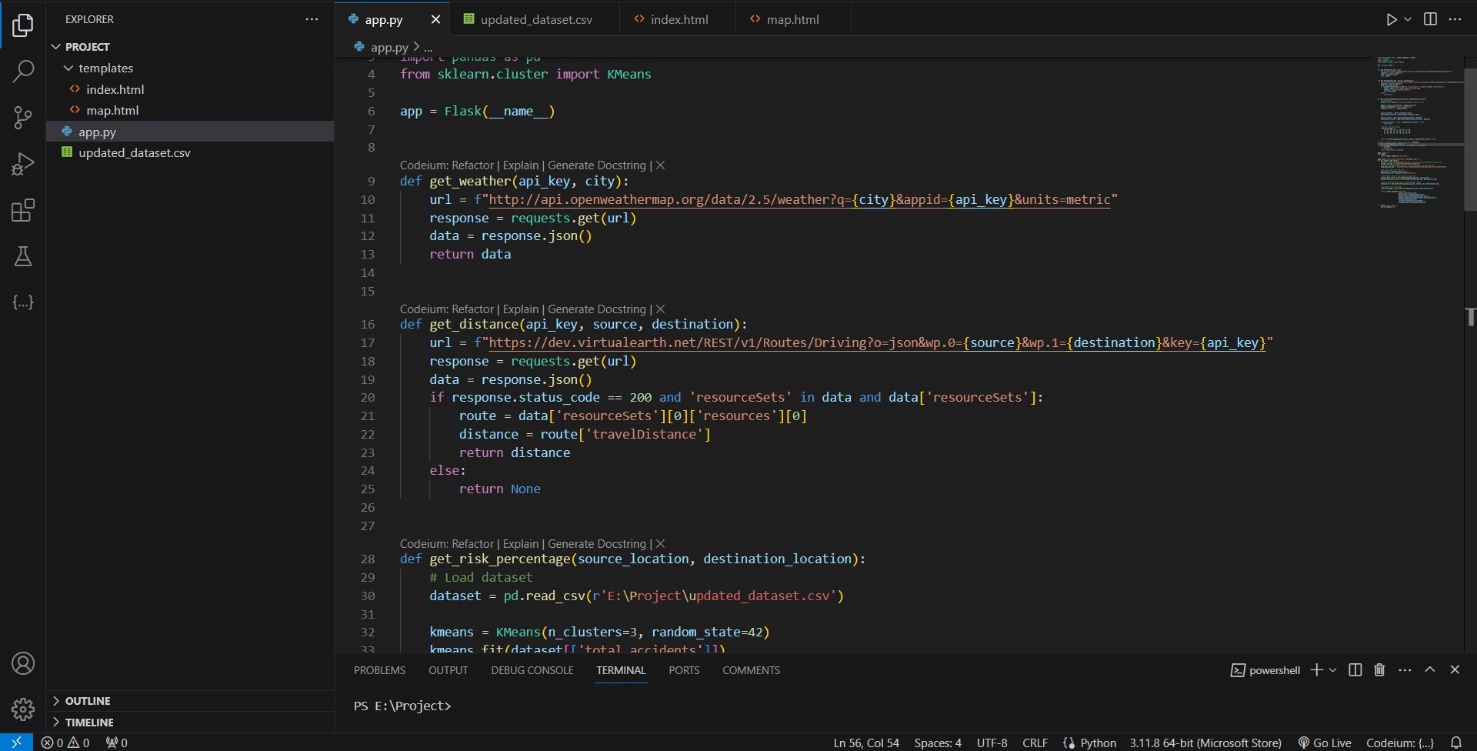
elif result[0][4] == 1: pre = "Citrus\_Scab"

resu = "Spray Carbendazim 0.1%"

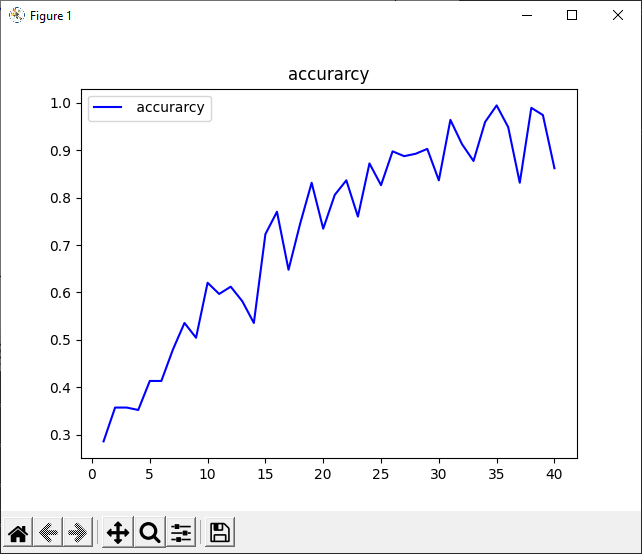
return render\_template('Prediction.html', pre=pre, result=resu) if name == ' main ':

app.run(debug=True, use\_reloader=True)

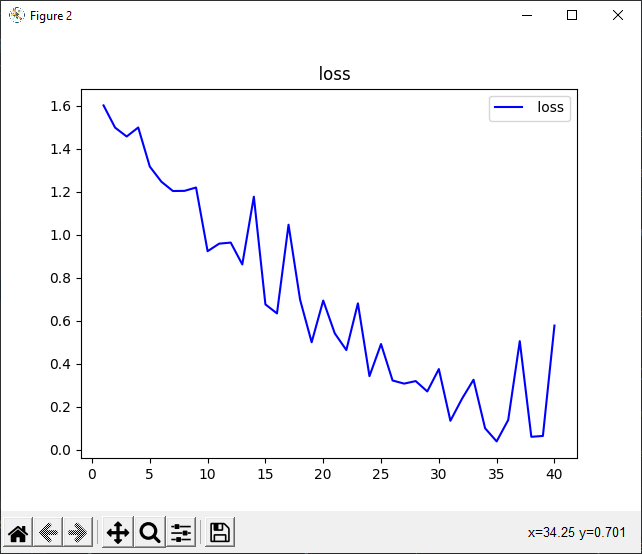
###### APPENDIX III SCREENSHOTS



**FIG.11 SCREENSHOTS**



###### FIG 11.1 ACCURARCY



**FIG 11.2 LOSS**

2023-02-09 20:56:06.374280: W

tensorflow/stream\_executor/platform/default/dso\_loader.cc:64] Could not load dynamic library 'cudart64\_110.dll'; dlerror: cudart64\_110.dll not found

2023-02-09 20:56:06.375037: I tensorflow/stream\_executor/cuda/cudart\_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.

Found 228 images belonging to 5 classes.

Found 152 images belonging to 5 classes. 2023-02-09 20:56:17.683805: W

tensorflow/stream\_executor/platform/default/dso\_loader.cc:64] Could not load dynamic library 'nvcuda.dll'; dlerror: nvcuda.dll not found

2023-02-09 20:56:17.684089: W tensorflow/stream\_executor/cuda/cuda\_driver.cc:269] failed call to cuInit: UNKNOWN ERROR (303)

2023-02-09 20:56:17.706533: I tensorflow/stream\_executor/cuda/cuda\_diagnostics.cc:169] retrieving CUDA diagnostic information for host: DESKTOP-9BF8NUN

2023-02-09 20:56:17.706938: I tensorflow/stream\_executor/cuda/cuda\_diagnostics.cc:176] hostname: DESKTOP-9BF8NUN

2023-02-09 20:56:17.729248: I tensorflow/core/platform/cpu\_feature\_guard.cc:151] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX AVX2

To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags. Model: "sequential"

Layer (type) Output Shape Param #

=================================================================

conv2d (Conv2D) (None, 198, 198, 16) 448

max\_pooling2d (MaxPooling2D (None, 99, 99, 16) 0

)

conv2d\_1 (Conv2D) (None, 97, 97, 32) 4640

max\_pooling2d\_1 (MaxPooling (None, 48, 48, 32) 0 2D)

conv2d\_2 (Conv2D) (None, 46, 46, 64) 18496

max\_pooling2d\_2 (MaxPooling (None, 23, 23, 64) 0 2D)

conv2d\_3 (Conv2D) (None, 21, 21, 64) 36928

max\_pooling2d\_3 (MaxPooling (None, 10, 10, 64) 0

2D)

conv2d\_4 (Conv2D) (None, 8, 8, 64) 36928

max\_pooling2d\_4 (MaxPooling (None, 4, 4, 64) 0

|  |  |  |
| --- | --- | --- |
| 2D) |  | |
| flatten (Flatten) | (None, 1024) | 0 |
| dense (Dense) | (None, 128) | 131200 |
| dense\_1 (Dense) | (None, 5) | 645 |

=================================================================

Total params: 229,285

Trainable params: 229,285

Non-trainable params: 0

Epoch 1/40

WARNING:tensorflow:AutoGraph could not transform <function Model.make\_train\_function.<locals>.train\_function at 0x0000023958405798> and will run it as-is.

Please report this to the TensorFlow team. When filing the bug, set the verbosity to 10 (on Linux, `export AUTOGRAPH\_VERBOSITY=10`) and attach the full output.

Cause: 'arguments' object has no attribute 'posonlyargs'

To silence this warning, decorate the function with @tf.autograph.experimental.do\_not\_convert

2023-02-09 20:56:19.522988: W tensorflow/core/framework/cpu\_allocator\_impl.cc:82] Allocation of 80289792 exceeds 10% of free system memory.

2023-02-09 20:56:20.303860: W tensorflow/core/framework/cpu\_allocator\_impl.cc:82] Allocation of 80289792 exceeds 10% of free system memory.

1/7 [===> ] - ETA: 13s - loss: 1.6135 - accuracy: 0.25002023-02-09

20:56:20.522527: W tensorflow/core/framework/cpu\_allocator\_impl.cc:82] Allocation of 80289792 exceeds 10% of free system memory.

2023-02-09 20:56:21.011051: W tensorflow/core/framework/cpu\_allocator\_impl.cc:82] Allocation of 80289792 exceeds 10% of free system memory.

2/7 [=======>......................] - ETA: 3s - loss: 1.5862 - accuracy: 0.2969 2023-02-09

20:56:21.191479: W tensorflow/core/framework/cpu\_allocator\_impl.cc:82] Allocation of 80289792 exceeds 10% of free system memory.

7/7 [==============================] - 7s 716ms/step - loss: 1.6021 - accuracy:

0.2857

Epoch 2/40

7/7 [==============================] - 4s 574ms/step - loss: 1.4986 - accuracy:

0.3571

Epoch 3/40

7/7 [==============================] - 5s 619ms/step - loss: 1.4574 - accuracy:

0.3571

Epoch 4/40

7/7 [==============================] - 4s 567ms/step - loss: 1.4993 - accuracy:

0.3520

Epoch 5/40

7/7 [==============================] - 4s 546ms/step - loss: 1.3176 - accuracy:

0.4133

Epoch 6/40

7/7 [==============================] - 4s 547ms/step - loss: 1.2473 - accuracy:

0.4133

Epoch 7/40

7/7 [==============================] - 4s 648ms/step - loss: 1.2038 - accuracy:

0.4796

Epoch 8/40

7/7 [==============================] - 4s 549ms/step - loss: 1.2045 - accuracy:

0.5357

Epoch 9/40

7/7 [==============================] - 5s 625ms/step - loss: 1.2201 - accuracy:

0.5045

Epoch 10/40

7/7 [==============================] - 5s 612ms/step - loss: 0.9238 - accuracy:

0.6205

Epoch 11/40

7/7 [==============================] - 4s 532ms/step - loss: 0.9590 - accuracy:

0.5969

Epoch 12/40

7/7 [==============================] - 4s 652ms/step - loss: 0.9639 - accuracy:

0.6122

Epoch 13/40

7/7 [==============================] - 4s 652ms/step - loss: 0.8623 - accuracy:

0.5816

Epoch 14/40

7/7 [==============================] - 4s 545ms/step - loss: 1.1773 - accuracy:

0.5357

Epoch 15/40

7/7 [==============================] - 5s 639ms/step - loss: 0.6764 - accuracy:

0.7232

Epoch 16/40

7/7 [==============================] - 4s 542ms/step - loss: 0.6344 - accuracy:

0.7704

Epoch 17/40

7/7 [==============================] - 4s 645ms/step - loss: 1.0471 - accuracy:

0.6480

Epoch 18/40

7/7 [==============================] - 4s 534ms/step - loss: 0.6979 - accuracy:

0.7449

Epoch 19/40

7/7 [==============================] - 4s 543ms/step - loss: 0.5002 - accuracy:

0.8316

Epoch 20/40

7/7 [==============================] - 4s 601ms/step - loss: 0.6944 - accuracy:

0.7347

Epoch 21/40

7/7 [==============================] - 4s 551ms/step - loss: 0.5416 - accuracy:

0.8061

Epoch 22/40

7/7 [==============================] - 4s 572ms/step - loss: 0.4641 - accuracy:

0.8367

Epoch 23/40

7/7 [==============================] - 4s 553ms/step - loss: 0.6813 - accuracy:

0.7602

Epoch 24/40

7/7 [==============================] - 4s 547ms/step - loss: 0.3426 - accuracy:

0.8724

Epoch 25/40

7/7 [==============================] - 4s 553ms/step - loss: 0.4920 - accuracy:

0.8265

Epoch 26/40

7/7 [==============================] - 4s 547ms/step - loss: 0.3223 - accuracy:

0.8980

Epoch 27/40

7/7 [==============================] - 4s 516ms/step - loss: 0.3079 - accuracy:

0.8878

Epoch 28/40

7/7 [==============================] - 4s 547ms/step - loss: 0.3196 - accuracy:

0.8929

Epoch 29/40

7/7 [==============================] - 4s 541ms/step - loss: 0.2714 - accuracy:

0.9031

Epoch 30/40

7/7 [==============================] - 4s 548ms/step - loss: 0.3756 - accuracy:

0.8367

Epoch 31/40

7/7 [==============================] - 4s 549ms/step - loss: 0.1350 - accuracy:

0.9643

Epoch 32/40

7/7 [==============================] - 4s 574ms/step - loss: 0.2372 - accuracy:

0.9133

Epoch 33/40

7/7 [==============================] - 4s 558ms/step - loss: 0.3260 - accuracy:

0.8776

Epoch 34/40

7/7 [==============================] - 4s 619ms/step - loss: 0.1003 - accuracy:

0.9598

Epoch 35/40

7/7 [==============================] - 4s 550ms/step - loss: 0.0391 - accuracy:

0.9949

Epoch 36/40

7/7 [==============================] - 4s 651ms/step - loss: 0.1388 - accuracy:

0.9490

Epoch 37/40

7/7 [==============================] - 4s 545ms/step - loss: 0.5054 - accuracy:

0.8316

Epoch 38/40

7/7 [==============================] - 4s 539ms/step - loss: 0.0605 - accuracy:

0.9898

Epoch 39/40

7/7 [==============================] - 4s 645ms/step - loss: 0.0644 - accuracy:

0.9745

Epoch 40/40

7/7 [==============================] - 4s 578ms/step - loss: 0.5779 - accuracy:

0.8622

WARNING:tensorflow:AutoGraph could not transform <function Model.make\_predict\_function.<locals>.predict\_function at 0x0000023958E01F78> and will run it as-is.

Please report this to the TensorFlow team. When filing the bug, set the verbosity to 10 (on Linux, `export AUTOGRAPH\_VERBOSITY=10`) and attach the full output.

Cause: 'arguments' object has no attribute 'posonlyargs'

To silence this warning, decorate the function with @tf.autograph.experimental.do\_not\_convert

Classification Report

precision recall f1-score support

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Citrus\_Black\_spot | 1.00 | 1.00 | 1.00 | 21 |
| Citrus\_Canker | 1.00 | 0.96 | 0.98 | 78 |
| Citrus\_Greening | 0.89 | 1.00 | 0.94 | 16 |
| Citrus\_healthy | 0.92 | 1.00 | 0.96 | 22 |
| Citrus\_Scab | 1.00 | 0.93 | 0.97 | 15 |

accuracy 0.97 152

macro avg 0.96 0.98 0.97 152

weighted avg 0.98 0.97 0.97 152

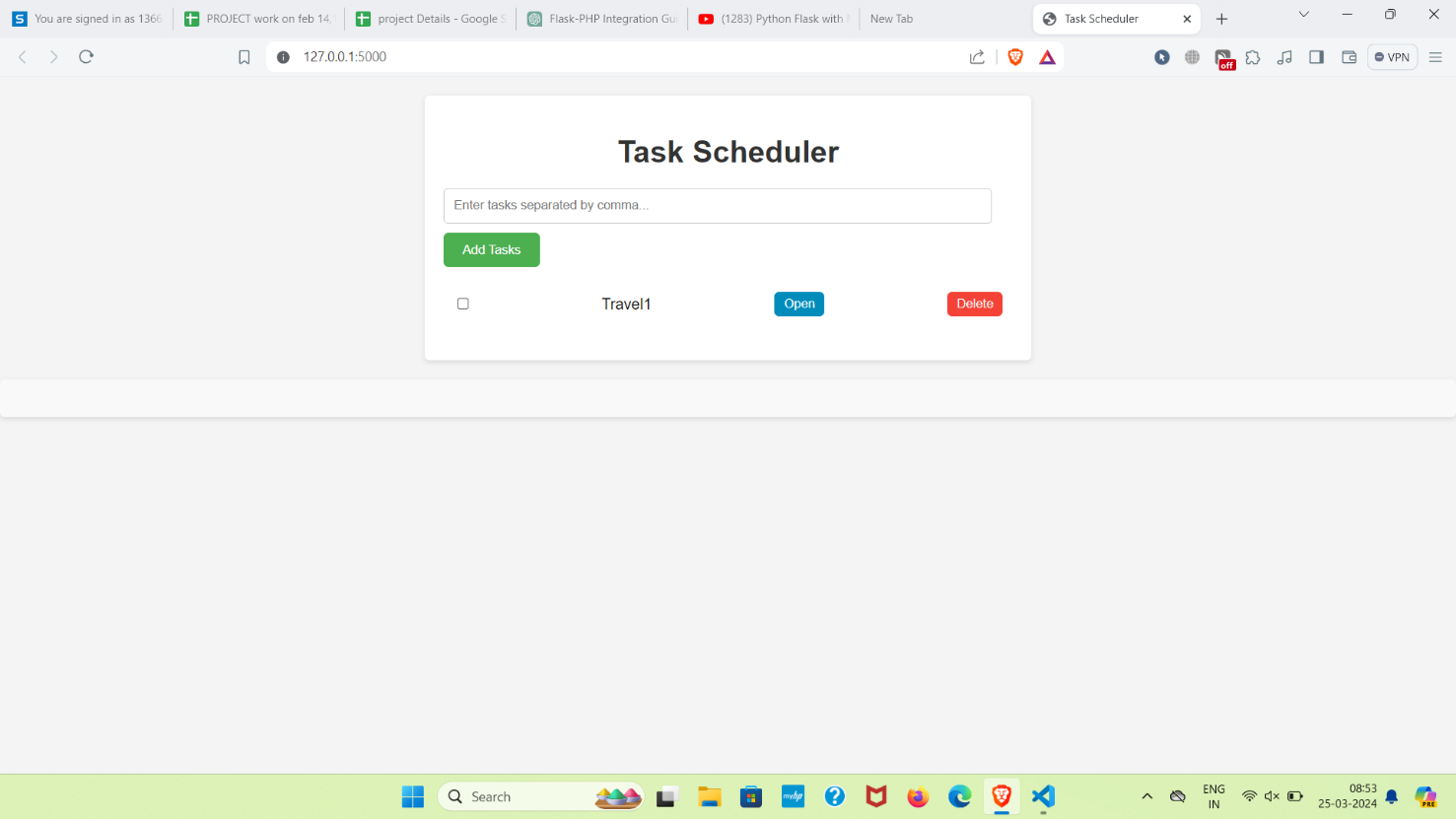
confusion matrix [[21 0 0 0 0]

[ 0 75 1 2 0]

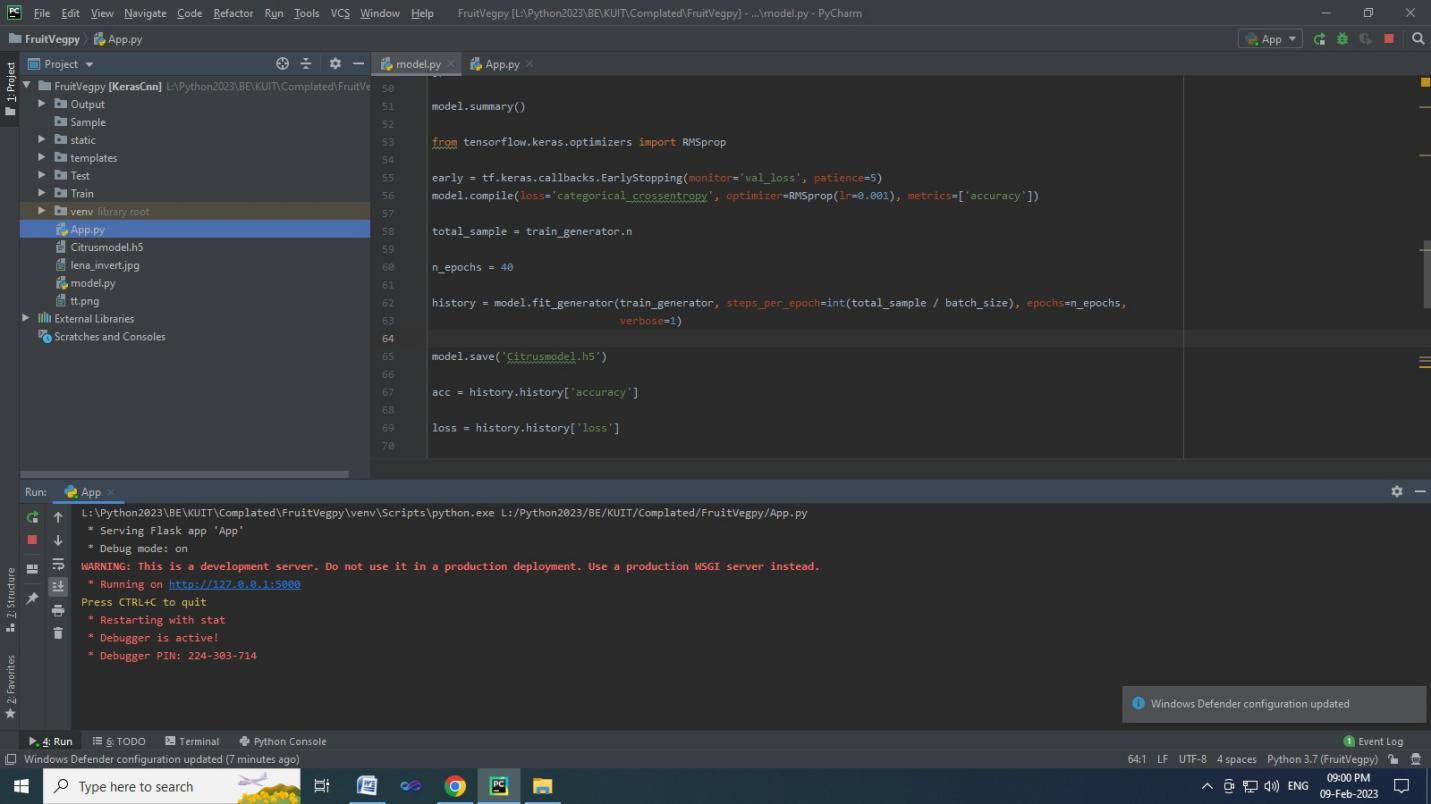
[ 0 0 16 0 0]

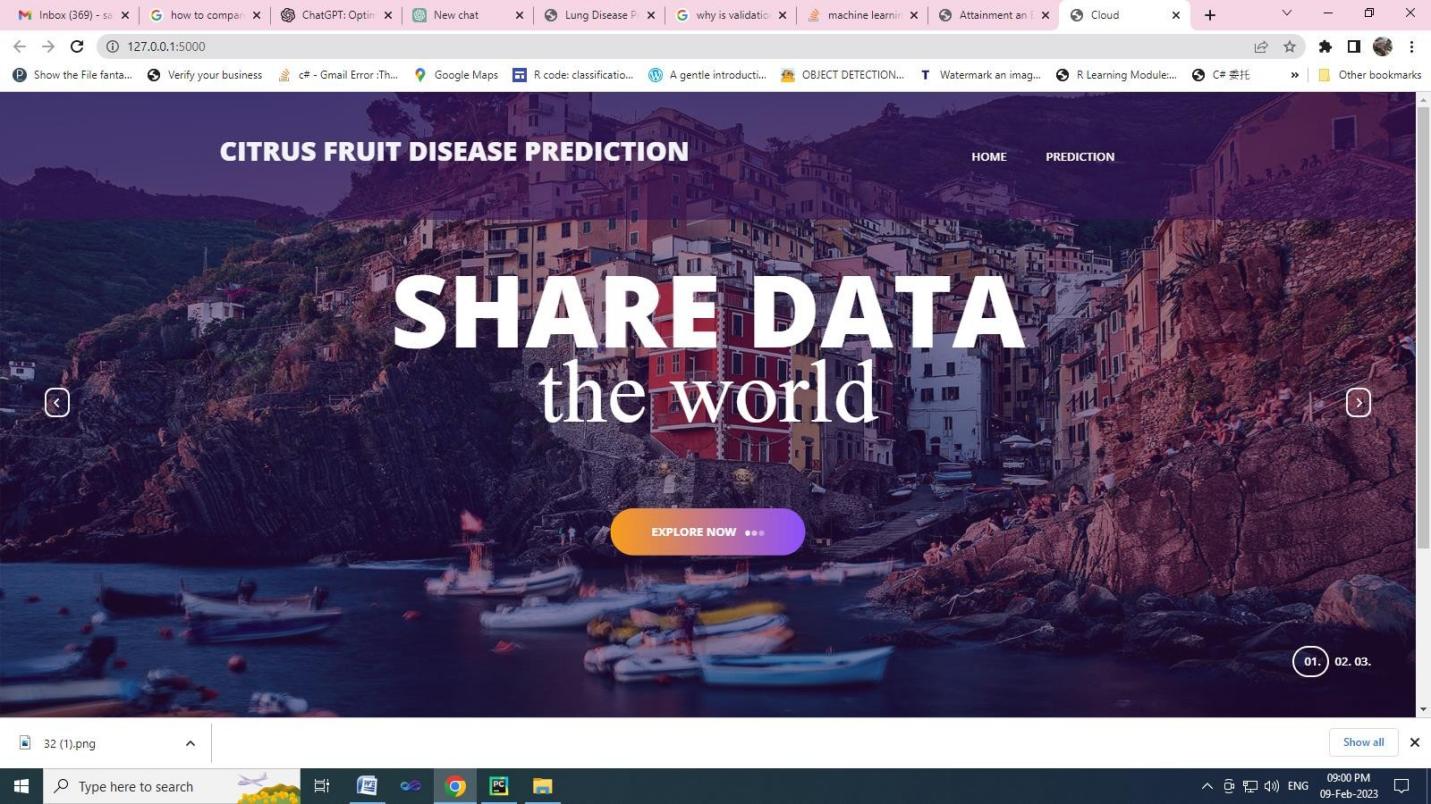
[ 0 0 0 22 0]

[ 0 0 1 0 14]]

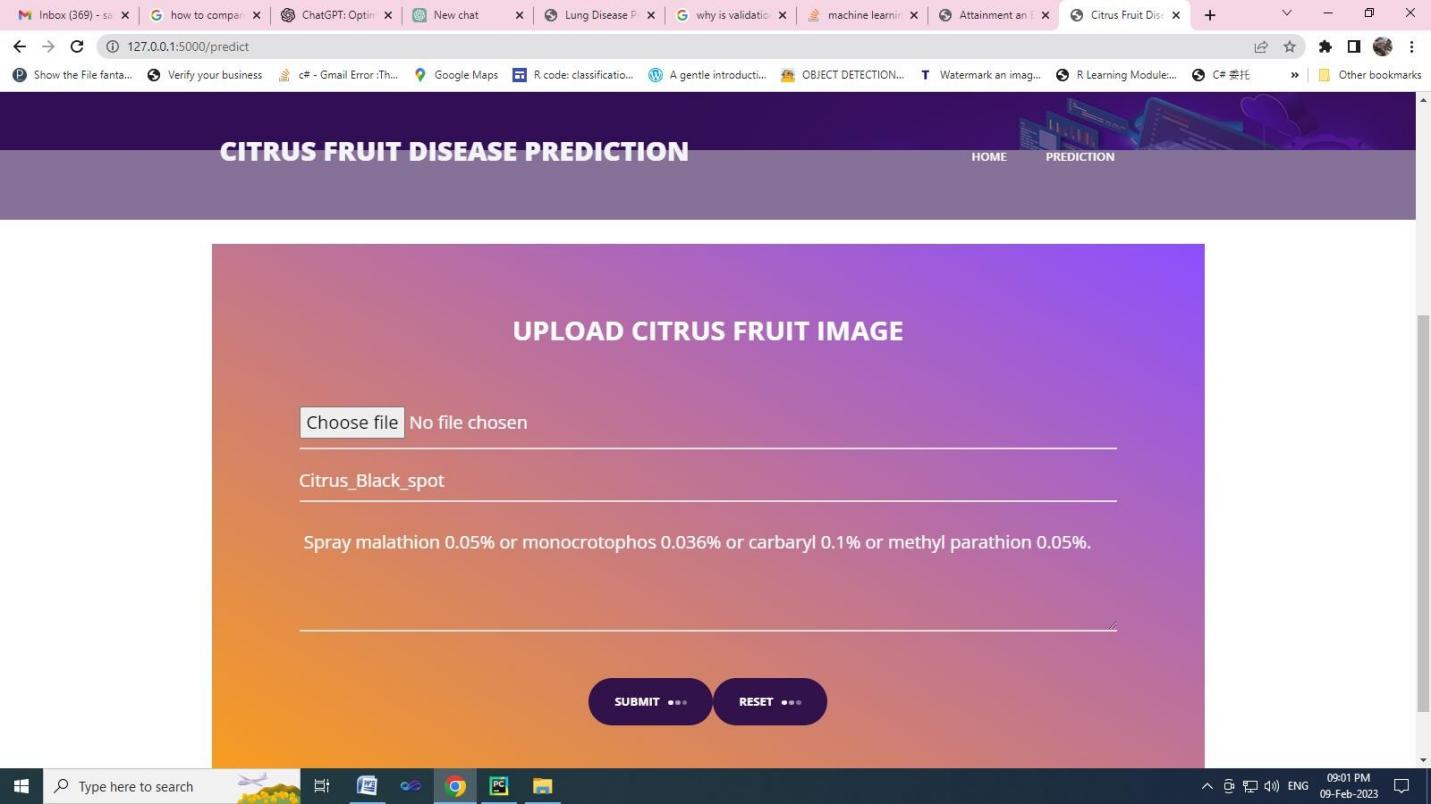


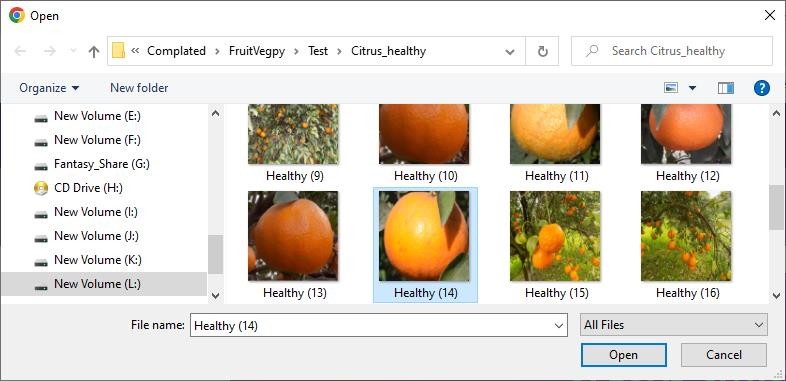
###### FIG 11.3



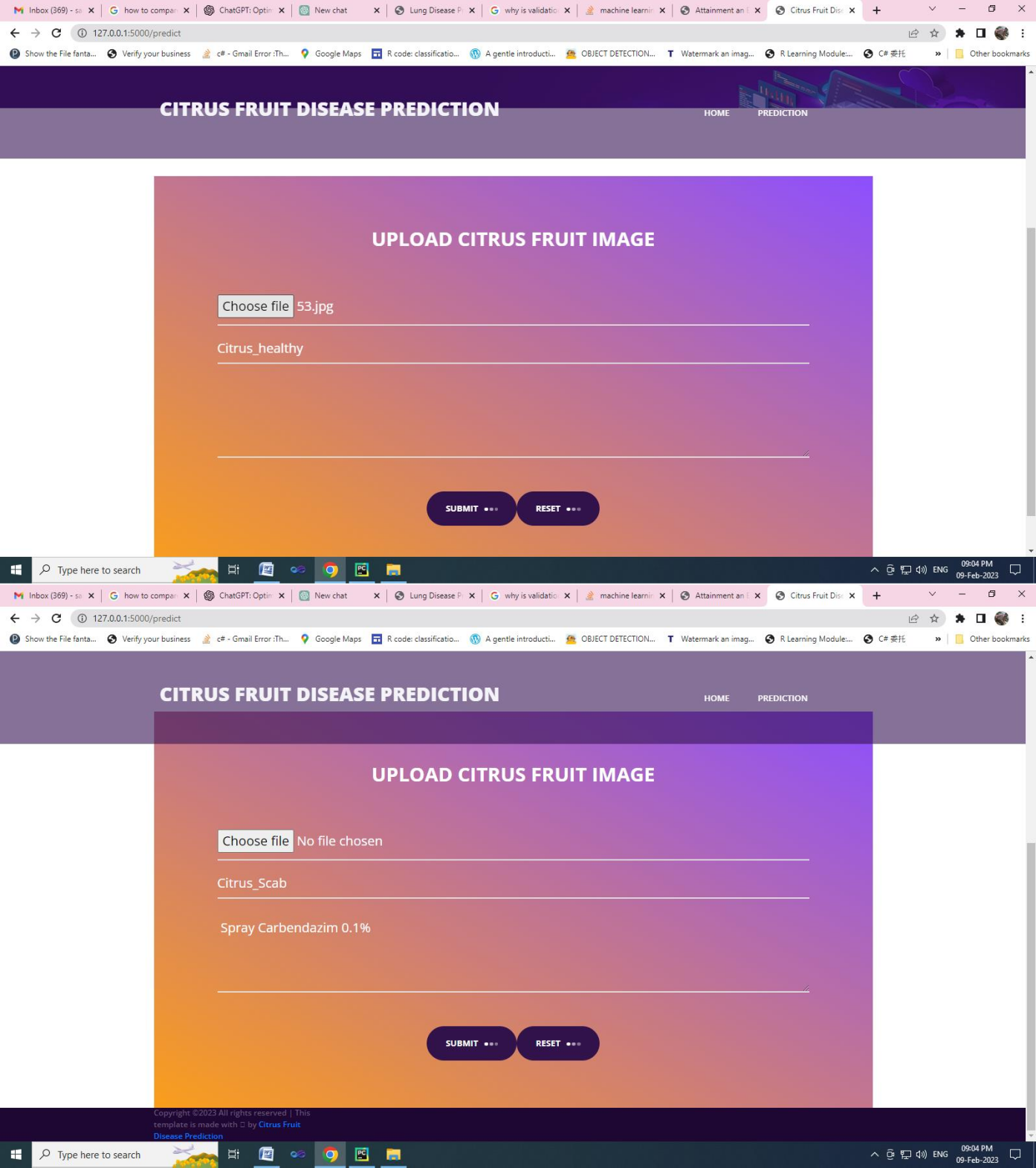


**FIG 11.4 LOGIN PAGE**





###### FIG 11.5 INPUT PAGE



**FIG 11.6 PREDICTION**

###### REFERENCES

1. Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and future research directions. User Modeling and User-Adapted Interaction, 15(1), 137-176.
2. Chen, S., Fang, Y., & Ye, M. (2018). Deep learning-based travel recommendation. In Proceedings of the 26th ACM International Conference on Information and Knowledge Management (pp. 1875-1884).
3. Dhuri, V., Raj, R., & Karthikeyan, S. (2022). A survey on Deep Learning based Recommender Systems: State-of-the-art, Challenges and Research Directions. Artificial Intelligence Review, 1-35.
4. Ganesh, K., & Rajagopal, S. (2019). A comprehensive survey on travel recommendation systems using machine learning and deep learning approaches. Knowledge-Based Systems, 178, 302-327.
5. Huang, Y., Zhao, Y., & Sun, M. (2023). A Survey of Deep Learning based Risk Assessment: Methods, Applications, and Challenges. Engineering, 11(2), 349-361.
6. Jurdziak, P., Słojewski, W., & Ziemba, P. (2019). Road Risk Assessment in Route Planning for Travel Recommender Systems. In Information Systems Management (pp. 19-37). Springer, Cham.
7. Kumar, S., & Ravi, J. (2020). Risk assessment in transportation: a literature review. International Journal of Sustainable Engineering, 13(6), 769-783.
8. Luo, W., & Zheng, Y. (2020). Deep learning for travel recommendation: A survey. ACM Computing Surveys (CSUR), 53(4), 1-39.
9. Mobasher, B., Adomavicius, G., Dayal, U., & Sharma, K. (2023). Personalization in recommender systems: History, progress, and challenges. Information Systems, 145, 104765.
10. Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. Springer.
11. Sheela, V. S., & Krishna, P. G. (2022). A Survey on Deep Learning Techniques for Risk Assessment in Transportation Networks. arXiv preprint arXiv:2203.07058.
12. Sun, Y., & Hu, Y. (2020). A survey on deep learning security for recommender systems. ACM Computing Surveys (CSUR), 53(5), 1-41.
13. Wang, J., & Lu, K. (2020). A survey on content-based recommender systems: From classical to deep learning approach. IEEE Access, 8, 66302-66324.
14. Zhang, S., Yao, A., & Ye, A. (2019). A survey on recommender systems: From classical methods to deep learning approaches. ACM Computing Surveys (CSUR), 52(1), 1-38.
15. Zheng, L., Chen, Z., Li, Y., Xie, X., & Yin, W. (2018). Deep learning techniques for recommender systems: A survey and promising directions. arXiv preprint arXiv:1804.09407.
16. Bao, W., He, J., Shen, Y., Xu, W., & Yuan, X. (2021). Context-aware deep learning for travel recommendation. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, No. 1, pp. 1420-1429).
17. Chen, L., Wu, S., Liu, M., Zhou, M., & Tan, K. (2020). Travel time-aware personalized recommendation: A dynamic graph attention network approach. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 2208-2217).
18. Ding, X., Li, Y., Zheng, H., Qian, W., & He, Q. (2022). Social travel recommendation with heterogeneous interactions by graph collaborative filtering. IEEE Transactions on Knowledge and Data Engineering, 34(7), 2251-2266.
19. Gao, R., Zhuang, Y., & Zhang, K. (2020). A survey on hybrid recommender systems based on matrix factorization and collaborative filtering. Knowledge-Based Systems, 203, 104170.
20. Pan, W., Sun, Y., Yu, Z., Zhou, Y., & Zhu, M. (2020). A survey on transfer learning in recommender systems. ACM Computing Surveys (CSUR), 53(3), 1-38.
21. Bhattacharya, S., Roy, K., & Mukherjee, A. (2020). A survey on machine learning algorithms for risk assessment in internet of things. ACM Computing Surveys (CSUR), 53(3), 1-37.
22. Hwang, S., & Sohn, S. Y. (2020). A survey of recent advances in explainable artificial intelligence for risk assessment. International Journal of Information Management, 54, 102296.
23. Khaleghi, B., Zheng, S., & Shafieezadeh, A. (2021). An overview of explainable artificial intelligence for risk assessment and credit scoring. IEEE Transactions on Engineering Management, 69(3), 1300-1315.
24. Liu, M., Li, W., Li, M., & Yu, P. (2020). A survey of deep learning applications in transportation safety. IEEE Transactions on Intelligent Transportation Systems, 21(8), 3206-3223.
25. Nguyen, T. Q., Li, Y., Wu, L., & Zhu, H. (2020). A survey on deep learning for natural hazard risk assessment. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14(3), 1409-1431.
26. Balakrishnan, R., Kumar, S. P., & Kumar, S. A. (2020). Sustainable tourism development in Tamil Nadu, India: An analysis of tourist preferences and perceptions. Journal of Sustainable Tourism, 28(8), 1206-1227.
27. Ganesh, K., Rajagopal, S., & Murugan, S. (2021). A hybrid recommendation system for tourist destinations in Tamil Nadu, India. In Proceedings of the International Conference on Intelligent Systems and Information Management (pp. 321-328). Springer, Singapore.
28. Jaiswal, R., & Chandra, V. (2022). Assessing the role of social media in promoting tourism in Tamil Nadu, India. Journal of Hospitality and Tourism Management, 44(2), 371-383.
29. Kumar, N., & Ramaswamy, V. (2021). Cultural tourism in Tamil Nadu, India: An exploratory study. International Journal of Tourism Research, 23(7), 804-824.
30. Rajendran, S., & Kumar, V. (2020). Tourist destination recommendation system using collaborative filtering for Tamil Nadu, India. International Journal of Intelligent Enterprise, 3(4), 202-210.