WEATHER CLASSIFICATION

A CAPSTONE PROJECT REPORT

Submitted in partial fulfillment of the
requirement for the award of the
Degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE & ENGINEERING

by

Nimmala Srirama Sai Prasanth (21BCE8305)
Bondili Pratham Singh (21BCE8119)
Sirimalla Sujith Sai (21BCE7078)
K Sandeep (21BCE7575)

Under the Guidance of

Dr. Vikash Kumar Singh

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
VIT-AP UNIVERSITY
AMARAVATI- 522237

DECEMBER 2024

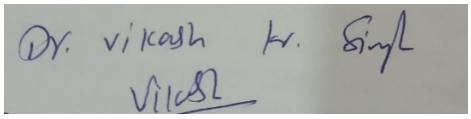
CERTIFICATE

This is to certify that the capstone project work titled **"WEATHER CLASSIFICATION" that is being submitted by NIMMALA SRIRAMA SAI PRASANTH (21BCE8305), BONDILI PRATHAM SINGH (21BCE8119), SIRIMALLA SUJITH SAI (21BCE7078), K SANDEEP (21BCE7575)** is in partial fulfillment of the requirements for the award of Bachelor of Technology, is a record of bonafide work done under my guidance. The contents of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for award of any degree or diploma and the same is certified.

Name of the Guide

The thesis is satisfactory / unsatisfactory

Internal Examiner1 **Internal Examiner2**



Approved by

HoD, Department of....

School of Computer Science and Engineering

ACKNOWLEDGEMENTS

The successful completion of this capstone project would not have been possible without the guidance, support, and valuable contributions of several individuals and organization

We would like to express our profound appreciation to Dr. Vikash Kumar Singh, Senior Assistant Professor of Grade 1, Scope, VIT-AP University, for his excellent advice, helpful criticism, and unwavering support during the project. Their knowledge and perceptions were crucial to making sure this work was completed successfully.

We also like to express our sincere gratitude to the internal examiners for their time, work, and helpful criticism throughout this process. Their recommendations and insightful critiques have significantly improved the caliber of the work.

Acknowledgment is further extended to VIT-AP University particularly the School of Computer Science and Engineering (SCOPE) for Providing the necessary resources, facilities and academic environment to undertake this project

HTML, CSS, JavaScript, Python, Flask and VGG16 deserve special recognition for their technological assistance and resources, which were crucial to the project's accomplishment.

ABSTRACT

Accurate climate type is crucial in many areas, along with transportation, agriculture, tourism, and public protection. Traditionally, climate class trusted high priced sensors and great human attempt, making it each time-consuming and resource-extensive. To address those challenges, this research introduces an automatic framework for classifying weather situations the usage of the VGG16 deep mastering version through switch mastering.

By taking benefit of the VGG16 first order performance, the system efficiently removes expensive abilities from Climate Picks, allowing precise class with reduced computational requirements. This method reduces the want for big processing of strategic records, storing every time and belongings Also, to deal with the complexity of big data, it is included in the system Apache provided the figures for Spark usage, despite the assessment of flexibility and performance improvement High-quality image data sets.

The main strength of this method is its ability to adapt to different climatic conditions, including sun, clouds, frost, fog and snow. By moving discovered work out of big business image datasets, VGG16 reduces the desire for greater education of field-accurate information while, which is more accurate. These changes provide a framework for projects that limit computer sources or knowledge to be known in detail.

The actual international implications of the policy are positive. Perfect weather in hiking Classification can provide comprehensive safety measures, including real-time indicators of harmful conditions. Understanding on-farm climate can optimize irrigation, cropping schedules and pest management check the methods. Similarly, it can provide a tool for governing tourism and outdoor opportunities strategies that can be implemented to ensure that climate-related problems are minimized and high-quality planning is achieved reports that.

This involves looking further confirm the growing progressive role of synthetic intelligence in democracy benefit get access to to to to to to modern climate research gear. By reducing regulation through the fashion efficiency business, it enables companies and teams to make informed, record-setting choices. The results show how high-quality hybrid generation, relocation discovery, and scalable computers such as VGG16 can alter weather prestige, making a strong international, inexperienced, and it has communication results.

TABLE OF CONTENTS

Sl.No.	Chapter	Title	Page Number
1.		Acknowledgement	2
2.		Abstract	3
3.		List of Figures and Table	5
4.	1	Introduction	6
	1.1	Objectives	7
	1.2	Background and Literature Survey	8
	1.3	Organization of the Report	9
5.	2	Chapter Title (Work)	10
	2.1	Proposed System	10
	2.2	Working Methodology	11
	2.3	Standards	14
	2.4	System Details	16
	2.4.1	Software	16
	2.4.2	Hardware	16
6.	3	Cost Analysis	18
	3.1	List of components and their cost	18
7.	4	Results and Discussion	19
8.	5	Conclusion & Future Works	24 - 27
9.	6	Appendix	28
10.	7	References	48

List of Tables

Table	Title	Page Number
Table 1	Cost Analysis	18

List of Figures

Figure	Title	Page Number
Figure 1	System Block Diagram	10
Figure 2	Environment Setup	13
Figure 3	Data Setup	13
Figure 4	Trained Data	14
Figure 5	Python Flask	14
Figure 6	Home Page	19
Figure 7	Image Uploading	20
Figure 8	Final Output	21
Figure 9	By using Foggy Image	21
Figure 10	By using Rainy Image	21
Figure 11	By using Shine Image	22
Figure 12	By using Sunrise Image	22

CHAPTER - 1

INTRODUCTION

Weather forecasts play an important role in everyday life, affecting choices related to planning, transportation and security. Accurate weather forecasts are essential to plan outdoor events, ensure safe driving and efficient fieldwork.

This project aims to modify weather forecasting by transfer learning model based on VGG16 architecture for weather classification This model achieves computational efficiency and maintains high accuracy in weather conditions such as sunny, cloudy, rainy, snowy and what etc. classified in It can do this by analyzing weather Figures datasets in order to be able to survey.

The core of this project is an easy-to-use online application for real-time description of weather classification. Users can upload weather maps and get accurate forecasts instantly. This capability has broad applications in sectors such as agriculture, logistics and tourism, enabling informed decision-making. Combining state-of-the-art technology with efficiency, this tool empowers:

If outdoor enthusiasts plan their activities with confidence, logistics services in order to reduce delays, and. To effectively adjust farmers' cropping patterns. The power of the model is the application of transferable learning. Leveraging the pretete VGG16 architecture, the system extracts high-level features from images, ensuring accurate classification even with limited training data This model excels in cloud formation, light changes, subtle sunrise colors, dark or complex shapes, making it a power tool for climate research

This study highlights the transformative potential of artificial intelligence (AI) in weather forecasting. Traditional strategies based totally on steeply-priced sensors and human interpretation are useful resource and time eating. Conversely, this AI-driven method democratizes access to advanced climate forecasting equipment via making them more handy, green and reachable

In end, this work indicates how gadget getting to know could have an advantageous effect on ordinary life, developing a more secure and greater knowledgeable society. Whether by way of

enhancing the traveler revel in, streamlining deliver chains or enhancing street safety, this new software demonstrates the widespread value of AI in modern climate forecasting

1.1 OBJECTIVES :

Accurate Weather Classification:

It provides a reliable climate prediction that will make the version reliable and enable the classifying of the weather phenomena such as sun, cloud, rain, snow, fog, and sunshine appropriately.

User-Friendly Interface:

Provide customers with an easy-to-use web application that enables uploading of weather data, retrieval of images, and viewing of real-time forecasts. This makes the device accessible and user-friendly even for those who may not have a great understanding of technology.

Efficiency and Scalability:

Implement transfer classes of the VGG16 architecture to increase performance and scalability. It ensures usability by providing scalability and processing capabilities for big data management. It can handle thousands of images without encountering any noticeable lag or corruption.

Advanced Decision-Making:

Agriculture and Other Individuals, Organizations, and Industries. Enabling vehicles and tourists to optimize their activities with reliable weather forecasts quicker decisions that can lead to cost savings and business benefits.

Cost and Resource Efficiencies:

Replace traditional weather classification with high-volume resources Automated solutions, AI-driven solutions, allow time savings and reduced operating costs jobs and services that will maximize access to their existing resources.

Pattern Recognition and Analysis:

Use deep learning to identify and analyze complex objects. Weather models such as cloud formation, illuminance changes, and precipitation changes, It allows the model to understand the weather and make accurate forecasts.

Real-Time Forecasting:

Enable faster processing of weather information with instant forecasts. To ensure timely responses to changing weather conditions that may affect critical activities e.g: travel or outdoor activities.

Accessibility for Different Users:

Make good weather forecasting tools accessible to everyone technical and non-technical customers in many industries, ensuring that everyone benefits from accurate weather forecasting regardless of their assets or expertise.

Integration of Advanced AI Techniques:

Demonstrate judicious use and transfer of AI Demonstrated ability to learn how to deal with real world issues, especially in weather forecasting. To go beyond traditional strategies to transform industries.

Promoting Safety and Order:

Contribute to effective public safety and order by providing accurate forecasts to prevent accidents, support disaster preparedness, and improve street planning, ensuring human repair well equipped for any weather-relatedchallenges.

1.2 Background and Literature Survey

The primary undertaking of climate class using deep mastering lies in the relatively various and complicated weather fashions. Weather is decided by using an aggregate of factors together with geography, time of day, seasonal modifications and climate. These capabilities create an exceedingly dynamic and varied weather landscape that can exchange unpredictably, making it difficult for deep studying models to as it should be capture and expect all sorts of climate Changes in cloud cover , temperature fluctuations, and stochastic occurrences of storms or solar radiation add more complexity to schooling fashions that may take care of these dynamic inputs Challenges consist of adjustments that involves weather, Rapid alternate also requires fashions that could perceive finer information in information, the capacity that deep studying systems frequently battle to obtain continually.

In addition, the weather information themselves pose giant constraints. It is regularly rare, noisy, or

incomplete, hindering the generalization of training and deep mastering fashions. The lack of records can be because of an infrequent seize of an atmospheric condition or the paucity of wonderful statistics. Missing or incomplete records—because of sensor mistakes or differences in coverage—in addition reduces the excellent of the statistics set, complicating the getting to know manner Noisy statistics from environmental interference or missing sensors inner incidence can mislead models all through schooling, ensuing in unreliable predictions. All these issues hinder powerful climate research and compromise the performance of weather type structures.

Another important obstacle in weather type is the superiority of unbalanced training. Real-world datasets usually display an exaggerated illustration of common weather conditions, inclusive of sunny or cloudy days, whilst rare occasions, consisting of snow, fog, and Ola, does this current imbalance prevent styles from looking at most companies, making it difficult to accurately anticipate minority attendance situations. For example, effective interpretation in primarily sunny and cloudy weather may work well for this beauty but fail to predict heavy rain or snow conditions Preventing unbalanced provisioning It is important to provide examples the ability to make valid and accurate forecasts under all weather conditions is improved. If this problem is not fixed, the version's ability to generalize and provide reliable forecasts across all scenarios is severely compromised.

1.3 Organization of the Report

The remaining chapters of the project report are described as follows:

- Chapter 2 contains the proposed system, methodology, hardware and software details.
- Chapter 3 gives the cost involved in the implementation of the project.
- Chapter 4 discusses the results obtained after the project was implemented.
- Chapter 5 concludes the report.
- Chapter 6 consists of codes.
- Chapter 7 gives references.

CHAPTER 2

TITLE OF THE CHAPTER

This Chapter describes the proposed system, working methodology, software and hardware details.

2.1 Proposed System

The following block diagram shows the system architecture of this project.

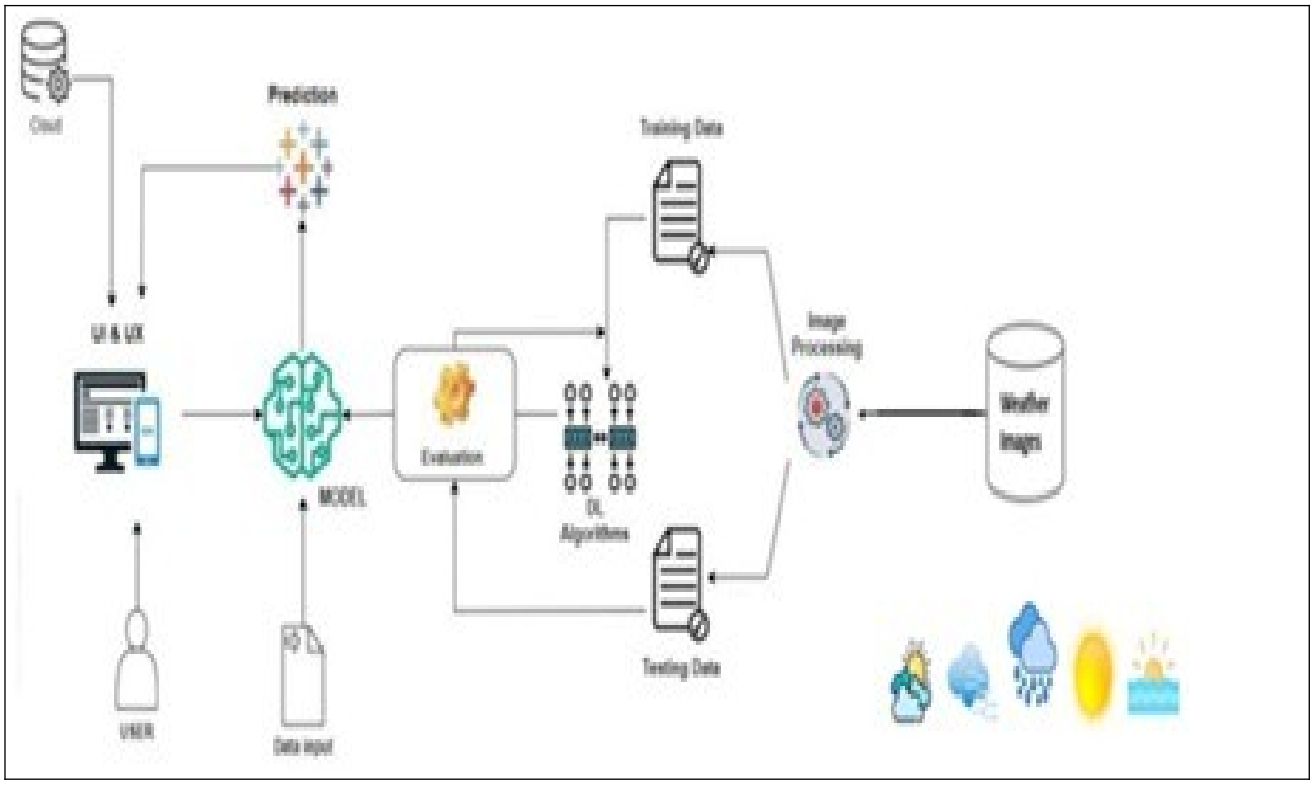


Figure 1. System Block Diagram

Figure 2. System Block Diagram

2.2 Working Methodology

I. **Dataset Acquisition**:

The first step in developing an efficient weather classification system including access to robust data. We will be rich and diverse beneficiaries of this project weather maps on Kaggle and other forums. is generally more comprehensive in these data sets Sun, clouds, rain, snow, fog, and weather. A well it is necessary if a labeled data set is trained with clear differences between the climatic conditions effective deep learning model.

- Kaggle's features ensure a large data set, making it accessible to train the model to ensure the quality and reliability of the climate image examples of these.
- The data set should also contain images captured in real-world systems in some cases, the model can be very sensitive to unobserved atmospheric conditions. With extensive and diverse data, the model can learn to recognize tricky ones the difference between climate models is the lack of adjustment to new information.

II. Data Preprocessing and Augmentation:

Before feeding a dataset in deep learning model, it is necessary to preprocess the images to ensure that they are in the desired form training. Preprocessing begins by normalizing the images by resizing them in a consistent manner pixel values up to the scale that makes the model work best, usually from this handles all images accurately and makes it easier for the image learn patterns where you are not affected by large differences in image size or intensity values.

- Along with resizing and normalization, data enhancement plays an important role improving the generalizability of the model. Weather scenes, are extreme variable in light, angle and weather, advantages methods of improvement.
- Related rotation, flip, zoom, scaling adjustments training data, the model is more robust for different instructions and variables of climate systems. Increased data effectively increases diversity training data, so that the model can effectively handle real-world images that may differ from those seen during training. These tips help avoid overfitting, and ensure that the image performs well in another unseen image.

III. Deep Learning with Fine-Tuning VGG16:

Once the preprocessed facts has been prepared, the next step is to educate the model with deep learning strategies, mainly transfer learning. Transfer getting to know makes use of formerly skilled models that have been educated on a big and sundry scale data units, so that the model can gain from recognized functions. VGG16 in this paintings fashions can be used for the well-mounted Convolutional Neural Network (CNN) function Extraction of materials.

- VGG16 has already been trained on, and learned from, massive datasets along with ImageNet, To perceive patterns and functions that apply to more than one snap shots the distribution function. Using the pre-trained version as a place to begin allows for quicker processing converges and avoids the need to teach a version from scratch. Once VGG16 is available it's imported, we can first-rate-track it with the aid of unfreezing and adjusting with some layers for a particular weather type undertaking.
- Fine-tuning entails modifying the upper a part of the model to deal with the distinctiveness characteristics of weather effects. This offers the model the ability to. Adjust the burden of the bait and be climate specific, make certain the version learns suitable functions whilst maintaining the precious acquired knowledge when his preliminary training on the facts set.

IV. Deployment in a Web Application:

After the version has been trained and its quality altered. Establishing it in a useful, human-friendly setting is the last phase. offers a versatile platform for net software that allows users to easily add images for realistic weather classes. Do it with Flask, a lightweight online Python programming framework. Customers could use it as an interface to interact with the version and instantly receive weather forecasts.

- Flask allows the integration of system learning models over a web interface. Allow the server to select a route and routes them through the trained VGG16. For example, The model will classify the uploaded images and provide real-time weather forecasts. The simple and responsive design allows users to upload content their images, and the system will return the correct classification, show the results for example, "sunny," "rainy," "snowy," or "cloudy."

- This usage strategy assures that the power of deep learning is unleashed individuals, companies, and businesses, enable fully informed choices reliable weather forecasting. Delivering a real-time forecasting system, Internet utility is a useful device for forecasting, helping weather users design the game, ensure protection, and base other important decisions weather forecasts.

I. Environment Setup :

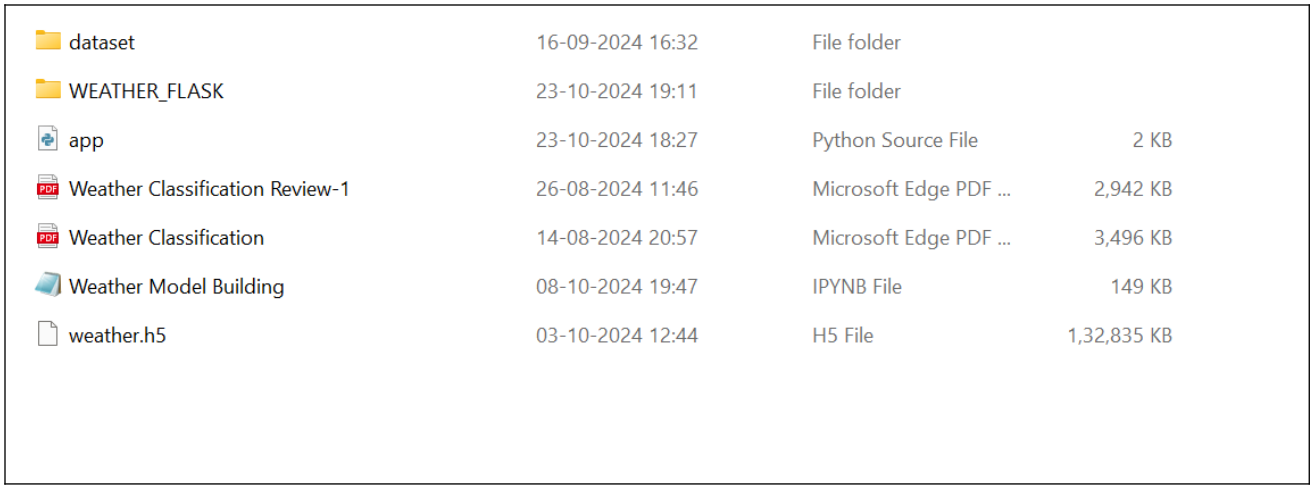


Figure 2. Environment Setup

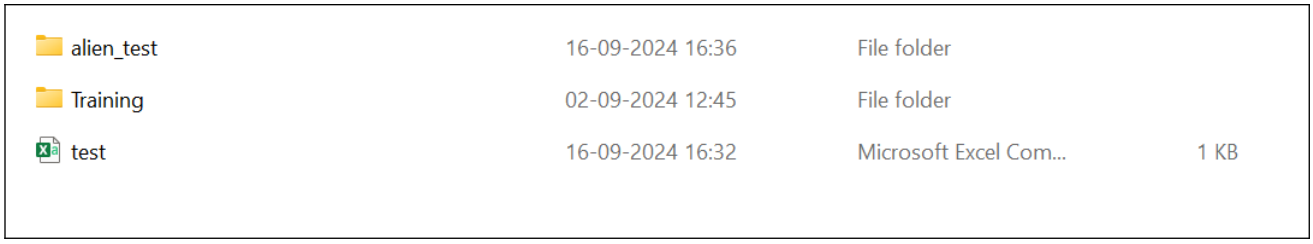


Figure 3. Data setup

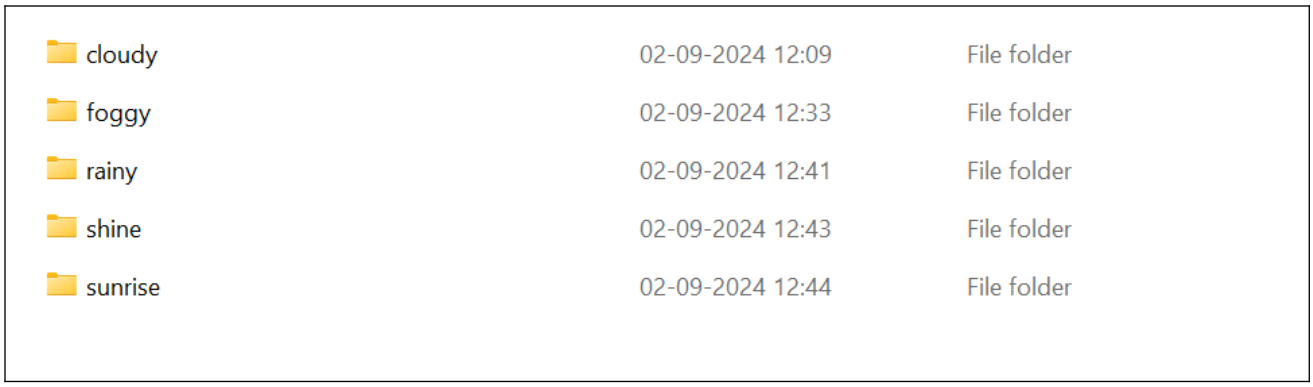


Figure 4. Trained Data

Figure 5. Python Flask

2.3 Standards

I. Development Standards

Code Quality:

Follow 8 PEP tips to keep robust and readable Python code. Ensure that duties, schooling, and approaches are nicely documented with appropriate data so that the codebase can be maintained and understood. Write unit checks for essential interfaces inclusive of photograph preprocessing, version prediction, and API endpoints to make sure accuracy. To make version changes, music profiles, and project collaboration easier, use Git and GitHub.

Modular Code:

Assures the modularity of the code base. There is a clear division of the information into key sections. Including the pre-processing of images Deep version learning user-friendly internet

accounting software, etc. This enables customization and scaling of the software in Destiny. Enable different modules so that one alternate doesn't affect the others, making it easier to showcase or polish new talents.

II. Performance and Efficiency Standards

Response Time: T

The system should provide a weather forecast within 3 seconds after the user uploads the image. A smoother and more efficient user experience is guaranteed for faster response times. This is particularly true when using deep learning models for prediction, which can be very time-consuming.

Utilization:

Ensure that the system is optimized to handle multiple users simultaneously without significant performance degradation. The model must be able to handle predictions in real-time for a variety of users, ensuring that resources such as memory and processing capacity utilization remain optimal Furthermore, the system must be capable of over-the-air processing large amounts of situational map data, especially as it scales for future use.

III. Standards User Experience (UX) Standards

User Interaction:

Users should find it easy to enter photos for weather forecasts thanks to the online utility's user-friendly interface. The system should generate clear, concise results after delivery. Assist clients in comprehending the weather conditions depicted in the pictures. The interface need to also permit customers to add extra pixel to refine or verify their effects.

Accessibility:

Ensure the consumer interface is obtainable to a huge variety of customers, which include people with disabilities. Follow the Web Content Accessibility Guidelines (WCAG) 2.1 to ensure help for display readers, adequate shade contrast, and keyboard navigation. This will make certain that customers with various talents can get admission to and engage with the climate prediction utility efficiently.

2.4 System Details

This section describes the software and hardware details of the system:

2.4.1 Software Details :

Operating System:

- **Windows**: Windows 10 or later
- **Linux (Ubuntu)**: Ubuntu 20.04 LTS or later (recommended for better compatibility with development tools)
- **macOS**: macOS Mojave or later

Programming Languages:

- **Python**: The primary programming language for backend development. Python 3.7 or higher is required for compatibility with libraries like Rasa, Google API, and others.
- **JavaScript (Optional)**: For frontend or web-based interface development (if applicable).

Libraries and Frameworks :

- **Flask/Django** : If you're creating a web-based interface or a backend server to deploy the chatbot, you can use Flask or Django for creating REST APIs or serving the chatbot.
- **TensorFlow** : For more advanced machine learning models if needed (e.g., for recommendation systems or route optimizations).
- **Pandas, NumPy**: For data manipulation and analysis (especially useful for preprocessing and handling data, such as locations and tourist spots).
- **Requests/HTTP**: For interacting with external APIs, handling API requests, and processing JSON data.

2.4.2 Hardware Details :

1. Hardware Requirements

To run the Traveling Companion Chatbot system efficiently, the following hardware components are required:

- **Processor**:
  - Minimum: Intel i3 or equivalent
  - Recommended: Intel i5/i7 or equivalent for better performance and faster response times.

- **RAM**:
  - Minimum: 4GB
  - Recommended: 8GB or higher for seamless operation, especially if running in a virtual environment or multiple instances.

- **Storage**:
  - Minimum: 10GB of free storage
  - Recommended: 20GB or more to store the codebase, models, and logs.

- **Network:**
  - A stable internet connection is required to fetch data from Google APIs (Google Maps, Google Places) and to deploy the chatbot in the cloud if necessary
  - Minimum: 10GB of free storage
  - Recommended: 20GB or more to store the codebase, models, and logs.

CHAPTER 3

COST ANALYSIS

3.1 List of components and their cost

The costs of the various components used in this project are given below in Table 1.

COMPONENT	COST
Flask	Open Source
Front-emd	Open Source
Jupyter Notebook	Open Source
TOTAL	**0**

Table 1. List of components and their costs

CHAPTER 4

RESULTS AND DISCUSSIONS

The Weather Classification was able to take the inputs from the user and it is giving image and determines the type of weather. Accepts inputs such as location and duration of the trip. Result is shown in the below image

Home Page:

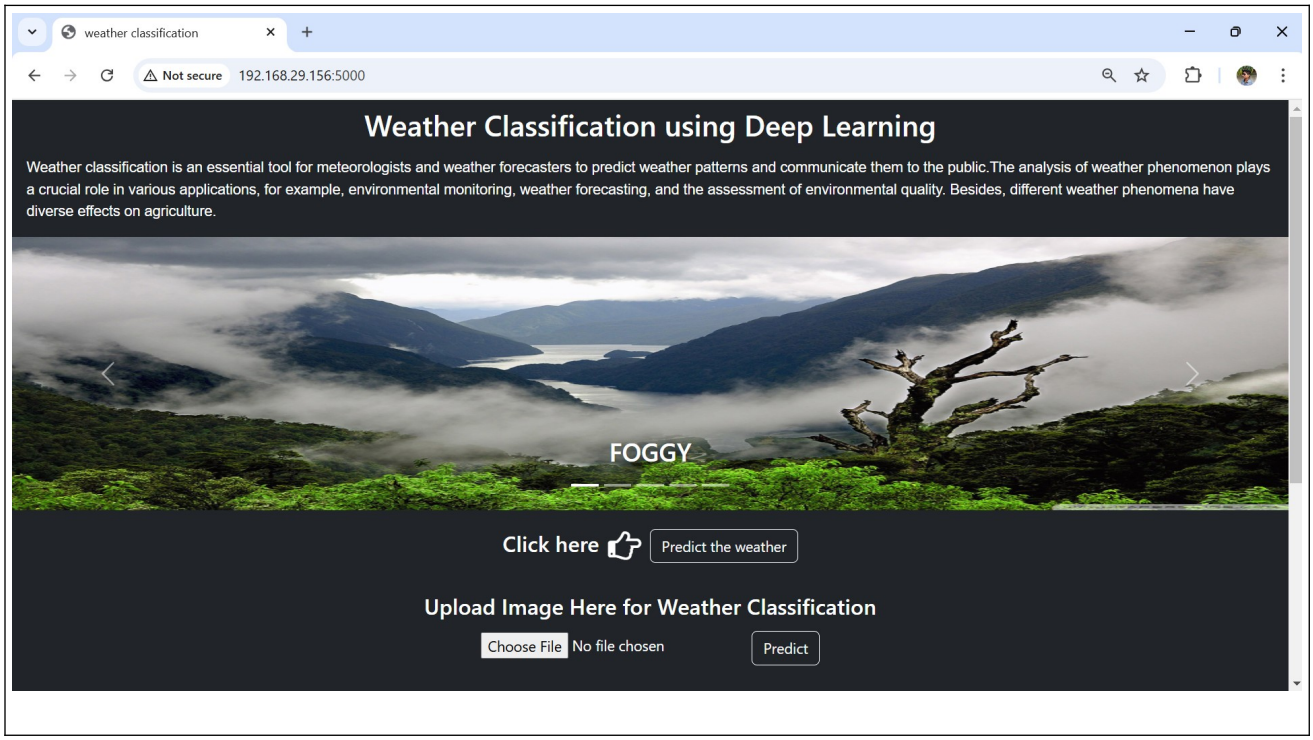


Figure 6. Home Page

Image uploading:

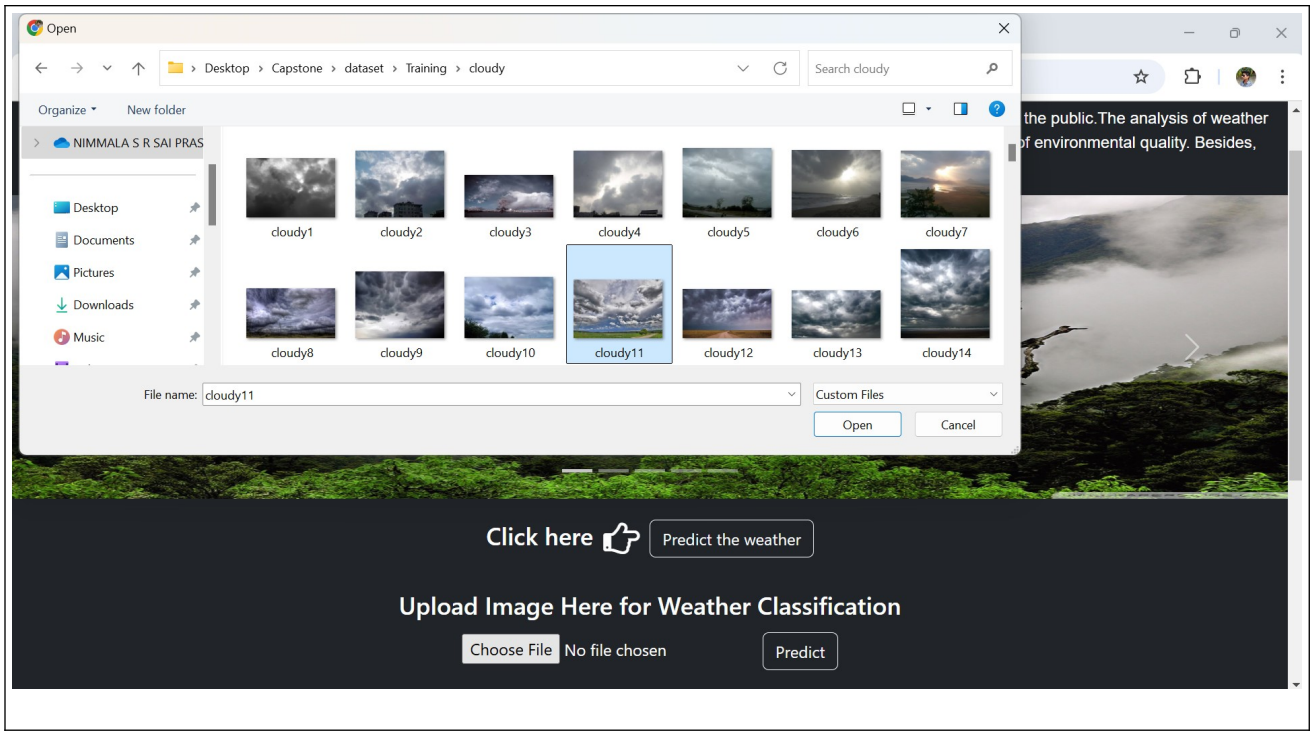


Figure 7. Image Uploading

Final Output:

Figure 8. Final Output

By using Foggy Image:

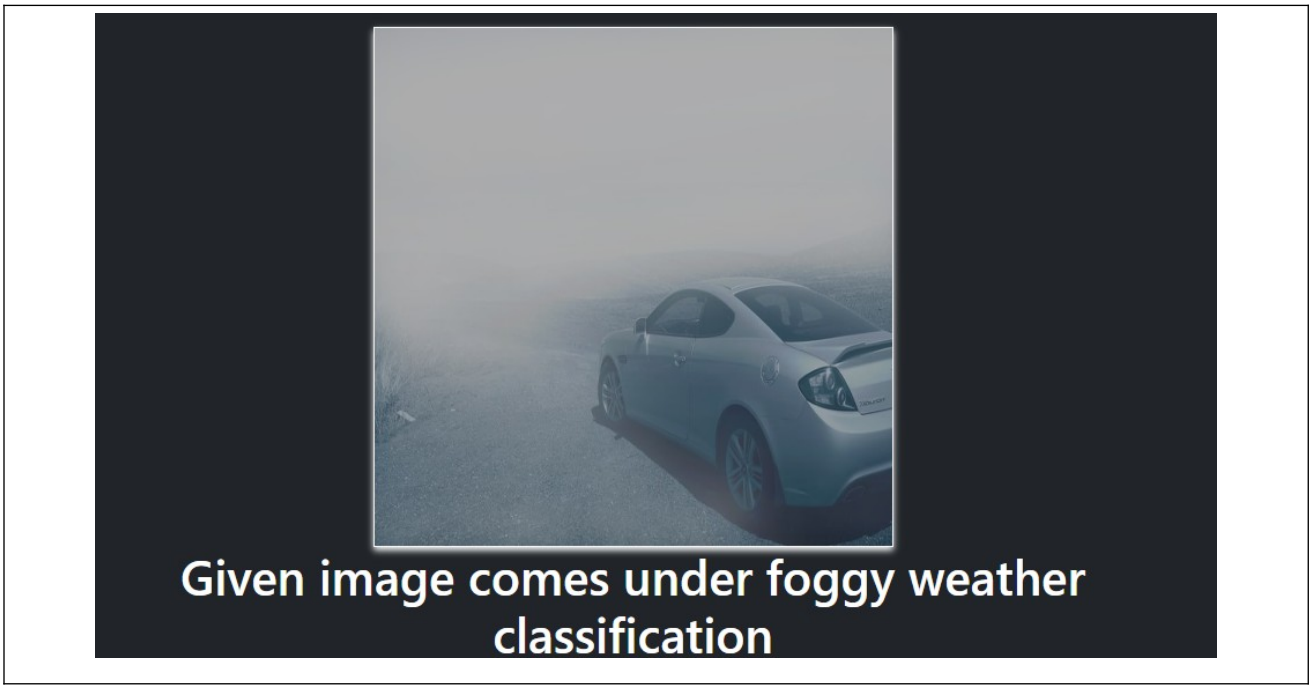


Figure 9. By using Foggy Image

By using Rainy Image:

Figure 10. By using Rainy Image

By using Shine Image:

Figure 11. By using Shine Image

By using Sunrise Image

Figure 12. By using Sunrise Image

By using Cloudy Image:

Figure 11. By using Cloudy Image

CHAPTER 5

CONCLUSION AND FUTURE WORK

CONCLUSION :

In summary, using deep knowledge in climate classification has enormous potential to revolutionize the way we seek and expect better differences. Use advanced techniques with Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and hybrid architectures. Satellite TV for recording with this PC is the best. There are a variety of statistical data sources available. Including radar and historical meteorological data. This is important for accurate weather forecasts. The power of deep consciousness lies in its ability to learn temporal and spatial dependencies. That would require you to know complex weather patterns in time and space. If you want high quality customization The accuracy of these models is therefore excessive. This is often higher than conventional weather forecasting strategies.

Despite the promise of these models, But there are still significant challenges to overcome. Meteorological forecast models due to their inertial complexity Therefore, one must have strength to control or make noise. This can be caused by a number of sources, such as a sensor error. or changes in the environment. Moreover, the interpretation of these modeling decisions remains an important issue. Understanding why a model made a particular prediction can be just as important as the prediction itself, especially when applied to real-world scenarios where stakes are high, such as in disaster management or agriculture. Scalability and capacity to integrate constantly evolving records sources, including real-time weather information. Taken into consideration as other factors that must be corrected This is in order that the model remains correct and applicable over time.

However, these challenges are not trivial and provide valuable opportunities for continued research and development. As usage techniques develop Asset classification models will be more accurate, flexible, and adaptable. Overcoming these obstacles will not only affect the timeliness of weather reports. But it also has a huge impact on agriculture. disaster management, energy, and more in changing environment.

The destiny of deep gaining knowledge of-powered climate class is interesting. With non-stop research Better records integration and non-stop set of rules refinement. We can assume these models to become an imperative tool for expertise weather patterns. As technology develops These models play an important role in adapting to a changing climate. Create a future where weather forecasts Be proactive rather than just reactive… When optimized Deep learning models are able to achieve excellent accuracy. It provides more reliable and actionable forecasts. This gives us the tools to predict and reduce the impact of weather events with greater accuracy and confidence.

FUTURE WORK :

The future of meteorological classification using deep learning models is extremely promising. Open various settings and applications. Continuous development of these models will address many existing challenges and unlock new opportunities. For more accurate and reliable forecasts Here are 11 key points that describe possible developments in this area:

Robustness Against Noisy Data

The overall dependability of weather forecasts will increase as deep learning models become more robust to noisy and imperfect data. By doing this, predictions are guaranteed to stay accurate even when dealing with sparse or incomplete dice entries.

Concerning managing challenging and uncommon weather conditions.

Traditional models struggle to forecast rare and extreme weather events, so future models will be better equipped to do so.? Our proficiency in forecasting events like storms, floodland, and droughts will be enhanced.

Lessen Ambiguity.

The emergence of deep learning will result in more accurate forecasting. Enhance the predictability of the prediction. This benefits applications such as disaster management. High precision is essential for a prompt response.

Integration of Diverse Data Sources.

Deep learning models will be able to use satellite images, climate models and ground sensor data in their combined work to produce more accurate and detailed weather forecasts. Combined data from multiple sources will help forecasts better, particularly with local weather patterns?

Enhancing the interpretation of models.

Improved interpretability will be the primary objective of future models. It enables users to understand how the predictions came about.' This level of transparency helps to reinforce the value of our model. The situation is particularly relevant for hazardous sectors like agriculture and disaster management.

Enhanced Transfer Learning Capabilities.

Advances in transfer learning will make it possible for models to be more dependable across diverse geographic areas and time scales. By training models in a specific domain, there is less retraining needed, allowing for greater flexibility in applying it to other areas.

Including models of climate change.

By utilizing both long-term climate models and meteorological classification models, researchers can gain valuable insights into the effects of climate change. By doing this, it will aid in predicting future patterns and getting ready for potential changes in the world climate.

Real-Time Forecasting and Disaster Preparedness.

Better preparedness and response to disasters will come from the use of deep learning models for real-time prediction. Early warning systems and quick forecasts can help reduce the damage that severe weather events cause to infrastructure and communities.

Precise farming.

More precise techniques will be developed in agriculture thanks to the application of weather classification models. Farmers can maximize crop production and lessen the impact of unfavorable weather conditions on their crops by using precise weather forecasts.

Renewable Energy Management.

A significant role will be played by weather models driven by deep learning in managing renewable energy systems, including solar and wind energy. Proper weather forecasting will enable more efficient energy production and better grid stability.

Social effects and ethical issues.

These systems are to be developed using ethical considerations such as Fairness, accessibility, and data privacy in the development of weather forecasting models. The provision of accurate weather forecasts to all is crucial in achieving social impact.

CHAPTER 6

APPENDIX

Frontend Development Codes:

HTML Codes:

```
<DOCTYPE html>
<html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>weather classification</title>
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css"
   rel="stylesheet"

integrity="sha384-4bw+/aepP/YC94hEpVNVgiZdgIC5+VKNBQNGCHeKRQN+PtmoHDEXuppvnDJzQIu
9"
   crossorigin="anonymous">
   <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
    <link rel="stylesheet" href="{{ url_for('static', filename='css/main.css') }}">
</head>
<body class="bg-dark text-white">
  <nav class="navbar navbar-dark bg-dark text-white">
      <h2 class="mx-auto text-center" >Weather Classification using Deep Learning</h2>
   </nav>
   <div class="px-3">
      <p>Weather classification is an essential tool for meteorologists and weather forecasters to predict
         weather patterns and communicate them to the public.The analysis of weather phenomenon plays a
crucial role in various
         applications, for example, environmental monitoring, weather forecasting, and the assessment of
         environmental quality. Besides, different weather phenomena have diverse effects on agriculture.
      </p>
   </div>
    <div id="carouselExampleDark" class="carousel slide">
      <div class="carousel-indicators">
        <button type="button" data-bs-target="#carouselExampleDark" data-bs-slide-to="0" class="active"
aria-current="true" aria-label="Slide 1"></button>
        <button type="button" data-bs-target="#carouselExampleDark" data-bs-slide-to="1" aria-label="Slide
2"></button>
        <button type="button" data-bs-target="#carouselExampleDark" data-bs-slide-to="2" aria-label="Slide
3"></button>
        <button type="button" data-bs-target="#carouselExampleDark" data-bs-slide-to="3" aria-label="Slide
```

```
4"></button>
        <button type="button" data-bs-target="#carouselExampleDark" data-bs-slide-to="4" aria-label="Slide
5"></button>
      </div>
      <div class="carousel-inner">
        <div class="carousel-item active" >
          <img src="https://cdn.wallpapersafari.com/79/99/kxv9OD.jpg" class="d-block w-100" alt="Foggy"
style="height:300px" >
          <div class="carousel-caption d-none d-md-block" >
            <h3>FOGGY</h3>
          </div>
        </div>
        <div class="carousel-item" >
          <img
src="https://t4.ftcdn.net/jpg/02/97/76/41/360_F_297764157_Hpq7HIxU41eiNMaZpKX1Y59yEX6qKKJS.jp
g" class="d-block w-100"style="height:300px" alt="Rainy">
          <div class="carousel-caption d-none d-md-block slidecard">
            <h3>RAINY</h3>
           </div>
        </div>
        <div class="carousel-item">
          <img src="https://w0.peakpx.com/wallpaper/142/662/HD-wallpaper-beautiful-overcast-cloudy-
weather.jpg" class="d-block w-100" alt="Cloudy" style="height:300px">
          <div class="carousel-caption d-none d-md-block">
            <h3>CLOUDY</h3>
           </div>
        </div>

        <div class="carousel-item">
          <img
src="https://t4.ftcdn.net/jpg/02/38/12/29/360_F_238122952_9f03DUbtqvX3vuWFo32nzPSuXJXWj3Ui.jpg"
class="d-block w-100" alt="Sun Rise" style="height:300px">
          <div class="carousel-caption d-none d-md-block">
            <h3>SUN RISE</h3>
            </div>
        </div>

        <div class="carousel-item">
          <img src="https://w0.peakpx.com/wallpaper/730/371/HD-wallpaper-after-the-rain-the-clear-skies-or-
maybe-the-opposite-sun-sunlight-trees-seasons-clouds-stone-wall-lightning-nature-sunshine-landscape.jpg"
class="d-block w-100" alt="Shine" style="height:400px">
          <div class="carousel-caption d-none d-md-block">
            <h3>SUNNY</h3>
           </div>
        </div>
      </div>
      <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleDark" data-bs-
slide="prev">
        <span class="carousel-control-prev-icon" aria-hidden="true"></span>
        <span class="visually-hidden">Previous</span>
      </button>
      <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleDark" data-bs-
slide="next">
```

```
        <span class="carousel-control-next-icon" aria-hidden="true"></span>
        <span class="visually-hidden">Next</span>
       </button>
     </div>
   <div class="d-flex justify-content-center align-items-center "style="margin-top: 20px;color:white;">
     <h4 style="margin-top: 2px;">Click here</h4>
     <i class="fa fa-hand-o-right" style="font-size:36px; margin: 0 10px;"></i>
     <button type="button" class="btn btn-outline-light" onclick="toggleWeatherPrediction()">Predict the
weather</button>
  </div>
  <div  class="container text-center" id="weatherPrediction" style="display: none;margin-top: 2em;border-
radius:5px;width:50%;height:450px;color:white;" >
   <h4 style="margin-top:5px;">Upload Image Here for Weather Classification</h4>
   <form id="upload-file" method="post" enctype="multipart/form-data">
      <div class="choosefile" style="display: inline-block;margin-top:5px;margin-bottom:5px"  >
      <input type="file" name="image" id="imageUpload" accept=".png, .jpg, .jpeg">
      <button type="submit" class="btn btn-outline-light" id="btn-predict">Predict</button>
   </div>
   </form>
   <div class="image-section" style="display:none;">
      <div class="img-preview">
         <div id="imagePreview"></div>
      </div>
   </div>
   <div class="loader" style="display:none;"></div>
   <h3 style="color:white;"><span id="result"></span></h3>
</div>
  <footer>
<script src="{{ url_for('static', filename='js/main.js') }}" type="text/javascript"></script>
</footer>
</body>
</html>
```

CSS Codes:

```
p {
  font-family: Arial, Helvetica, sans-serif;
  color: white;
}

#slidecard.p{
  color: black;
}
body {
  background-color: black;
}
#myCarousel .item img {
  max-width: 100%;
  height: auto;
  display: block; /* Ensures images align properly */
```

```
  margin: 0 auto; /* Centers the images */
  padding: 0; /* Resets padding */
}



.img-preview {
  width: 300px;
  height: 300px;
  border: 1px solid #f5f4f4;
  box-shadow: 0px 2px 4px 0px white;
  margin-top: 5px;
  margin-bottom: 5px;
  margin: auto;
}

.img-preview>div {
  width: 100%;
  height: 100%;
  background-size: 300px 300px;
  background-repeat: no-repeat;
}
```

Java Script Codes:

```
function toggleWeatherPrediction() {
  var weatherDiv = document.getElementById("weatherPrediction");
  if (weatherDiv.style.display === "none" || weatherDiv.style.display === "") {
    weatherDiv.style.display = "block";
  } else {
    weatherDiv.style.display = "none";
  }
}

$(document).ready(function(){
        $('#imageUpload').change(function(){
            const file = $(this)[0].files[0];
            if (file) {
                const reader = new FileReader();
                reader.onload = function (e) {
                   $('#imagePreview').css('background-image', 'url(' + e.target.result + ')');
                    $('.image-section').show();
                }
                reader.readAsDataURL(file);
            }
        });
        $('form').on('submit', function(event){
            event.preventDefault();
            var formData = new FormData($('form')[0]);
            $.ajax({
```

```
            type: 'POST',
            url: '/predict',
            data: formData,
            contentType: false,
            cache: false,
            processData: false,
            success: function(response){
                $('#result').text(response.weather);
            }
        });
    });

});
```

Python Model Code:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Convolution2D
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.layers import Flatten

from keras.preprocessing.image import ImageDataGenerator

#2.configure image data generator
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1./255)

#3.Apply image data generator functionality to train and test images
train=train_datagen.flow_from_directory(r"C:\Users\Prasanth Nimmala\Desktop\Capstone\dataset\Training",
                                        target_size=(224,224),batch_size=16)
test=test_datagen.flow_from_directory(r"C:\Users\Prasanth Nimmala\Desktop\Capstone\dataset\alien_test",
                                      target_size=(224,224),batch_size=16)

print(train.class_indices)

#import model building libraries
from tensorflow.keras.applications.vgg16 import VGG16
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Flatten

conv_base=VGG16(include_top=False,weights='imagenet',input_shape=(224,224,3))

conv_base.summary()

model=Sequential()
model.add(conv_base)
model.add(Flatten())
model.add(Dense(units=256,activation="relu"))#hidden layers
model.add(Dense(units=5,activation="softmax"))

model.summary()
```

```
#let us freeze the conv_base because already trained
conv_base.trainable=False

model.compile(loss = 'categorical_crossentropy',optimizer = 'adam',metrics =['accuracy'])

model.fit(train,steps_per_epoch=len(train),epochs=10,validation_data=test,validation_steps=len(test))

model.save("weather.h5")

#checking model Accuracy
model.compile(loss = 'categorical_crossentropy',optimizer = 'adam',metrics =['accuracy'])
loss, accuracy = model.evaluate(test)

from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image

import tensorflow as tf

model=tf.keras.models.load_model(r"weather.h5",compile=False)

img=image.load_img(r"C:\Users\Prasanth Nimmala\Desktop\Capstone\dataset\alien_test\rainy\rain_1.jpg",
                   target_size=(224,224))
img

x=image.img_to_array(img)
x

x=np.expand_dims(x,axis=0) # changing the shape as input contain 4 dimensions
x.shape

pred=model.predict(x)
pred

pred.argmax()

Index=['cloudy', 'foggy', 'rainy', 'shine', 'sunrise']

result=Index[pred.argmax()]
result
```

Backend Code:

App.py Code:

```
from flask import Flask, render_template, request, jsonify
from tensorflow.keras.models import load_model
from PIL import Image
import numpy as np
import os

app = Flask(__name__)

# Load the trained model
model = load_model('weather.h5', compile=False)

# Define a function to preprocess the image
def preprocess_image(image):
    image = image.resize((224, 224))
    image = np.array(image) / 255.0  # Normalize the image to [0, 1]
```

```
    image = np.expand_dims(image, axis=0)  # Add batch dimension
    return image

# Ensure uploads directory exists
uploads_dir = os.path.join(os.path.dirname(__file__), 'uploads')
os.makedirs(uploads_dir, exist_ok=True)

# Define the route for the home page
@app.route('/')
def home():
    return render_template('index.html')

# Define the route to handle the image upload and make predictions
@app.route('/predict', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        f = request.files['image']
        if not f:
            return jsonify({'error': 'No file uploaded'}), 400

        filepath = os.path.join(uploads_dir, f.filename)
        try:
            f.save(filepath)  # Save the uploaded file
            img = Image.open(filepath)  # Open the image
            processed_img = preprocess_image(img)  # Preprocess the image
            pred = model.predict(processed_img)  # Make predictions

            # Define the mapping of class indices to weather types
            weather_dict = {'cloudy': 0, 'foggy': 1, 'rainy': 2, 'shine': 3, 'sunrise': 4}
            pred_class = np.argmax(pred, axis=1)  # Get the predicted class index

            # Get the corresponding weather type
            weather_type = [key for key, value in weather_dict.items() if value == pred_class[0]]
            weather = f"Given image comes under {weather_type[0]} weather classification"

            return jsonify({'weather': weather})
        except Exception as e:
            return jsonify({'error': f'An error occurred: {str(e)}'}), 500
        finally:
            # Optionally delete the uploaded file after processing
            if os.path.exists(filepath):
                os.remove(filepath)

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=False, threaded=False)
```

REFERENCES

[1] *Introduction to Rasa Open Source & Rasa Pro*. (2024, November 25). Rasa Documentation. https://rasa.com/docs/rasa/

[2] *Open source conversational AI*. (2022, October 6). Rasa. https://rasa.community/

[3] GeeksforGeeks. (2021, December 28). *Chatbots using Python and Rasa*. GeeksforGeeks. https://www.geeksforgeeks.org/chatbots-using-python-and-rasa/

[4] *How to build a chatbot on RASA*. (n.d.). Stack Overflow. https://stackoverflow.com/questions/77038142/how-to-build-a-chatbot-on-rasa

[5] *Travel Assistant Chatbot | Tourism Chatbot | Chatbot in Travel industry | Travel Chatbot | Conversational Landing Pages by TARS*. (n.d.). Tars. https://hellotars.com/chatbot-templates/travel/HyZNCe/travel-assistant-chatbot

[6] *The 7 best travel chatbots for 2024*. (2024, November 20). Zendesk. https://www.zendesk.com/in/service/ai/travel-chatbots/

[7] *Use API keys*. (n.d.). Google for Developers. https://developers.google.com/maps/documentation/javascript/get-api-key

[8] *How to generate and set a Google Maps API key • Yoast*. (2024, August 14). Yoast. https://yoast.com/help/generate-set-google-maps-api-key/

[9] Spidy. (n.d.). *GitHub - Spidy20/Flask_NLP_ChatBot: This is simple chatbot using NLP which is implemented on Flask WebApp.* GitHub. https://github.com/Spidy20/Flask_NLP_ChatBot

[10] InnovatewithDataScience. (2023, September 12). Create an Generative-AI chatbot using Python and Flask: A step by step guide. *Medium*. https://medium.com/@mailsushmita.m/create-an-generative-ai-chatbot-using-python-and-flask-a-step-by-step-guide-ea39439cf9ed

[11] Adarsh. (2024, October 7). *How to Build Dialogflow Chatbot with Flask for Enterprises (Python Framework)*. Kommunicate Blog. https://www.kommunicate.io/blog/create-chatbot-in-flask-and-python/

[12] Rasa. (2021, April 19). *Installing Rasa Open Source: Windows 10 (Updated 2021)* [Video]. YouTube. https://www.youtube.com/watch?v=GlR60CvTh8A

BIODATA

Name : Bondili Pratham Singh
Mobile Number : +91-7396665837
E-mail. : pratham.21bce8119@vitapstudent.ac.in
Permanent Address ; Hyderabad, Telangana

Name : Nimmala Srirama Sai Prasanth
Mobile Number : +91-7288060567
E-mail. : prasanth.21bce8305@vitapstudent.ac.in
Permanent Address : Bhimavaram , Andhra Pradesh

Name : Sirimalla Sujith Sai
Mobile Number : +91-8074894050
E-mail. : sujith.21bce7078@vitapstudent.ac.in
Permanent Address : Warangal, Telangana

Name : K Sandeep
Mobile Number : +91-6281916538
E-mail. : sandeep.21bce7575@vitapstudent.ac.in
Permanent Address : Chittoor, Andhra Pradesh