International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 6, pp 218-222, June-2024

https://doi.org/10.47001/IRJIET/2024.806029

A Survey on Weather App Forecasting Using Machine Learning

[1]Dr. Kavyashree N, [2]Lakshmi T L

[1]Assistant Professor, Department of MCA, SSIT, Tumkur, Karnataka, India
[2]4th Semester MCA, SSIT, Tumkur, Karnataka, India
Authors E-mail: [1]kavyashree1283@gmail.com, [2]lakshmitl1024@gmail.com

Abstract **- Weather forecasting is one of the most scientifically and technologically challenging problems around the world in the last century. To make an accurate prediction is indeed, one of the major challenges that meteorologists are facing all over the world. To predict the conditions of the atmosphere for a given location, Weather Forecasting is used. Weather forecasting is made by collecting numerous data predicted by very proper understanding of the collected data. Weather simply refers to the condition of air on the earth at given place and time. It is a continuous, data-intensive, multidimensional, dynamic and chaotic process. These processes make weather forecasting a formidable challenge.**

Keywords: Weather application, API integration, MVC architecture, Openweathermap, Django Framework.

I. INTRODUCTION

Weather plays a crucial role in our daily lives, influencing our activities, travel plans, and overall well-being. The Weather App emerges as a practical solution for users seeking reliable weather information at their fingertips. Built on Django's robust framework, this application offers users a convenient platform to access weather forecasts, current conditions, and weather alerts for any location worldwide. Whether planning a trip, scheduling outdoor activities, or simply staying informed, the Weather App provides users with essential weather data in an intuitive and user-friendly interface.

Weather forecasting means predicting the weather and telling how the weather changes with change in time. Change in weather occurs due to movement or transfer of energy. Many meteorological patterns and features like anticyclones, depressions, thunderstorms, hurricanes and tornadoes occur due to the physical transfer of heat and moisture by convective process. Clouds are formed by evaporation of water vapour. As the water cycle keeps on evolving the water content in the clouds increases which in turn leads to precipitation.

Weather forecasting means predicting the weather and telling how the weather changes with change in time. Change in weather occurs due to movement or transfer of energy. Many meteorological patterns and features like anticyclones, depressions, thunderstorms, hurricanes and tornadoes occur due to the physical transfer of heat and moisture by convective process. Clouds are formed by evaporation of water vapour. As the water cycle keeps on evolving the water content in the clouds increases which in turn leads to precipitation. Weather forecasting is made by collecting numerous data predicted by very proper understanding of the collected data. Weather simply refers to the condition of air on the earth at given place and time. It is a continuous, data-intensive, multidimensional, dynamic and chaotic process. These processes make weather forecasting a formidable challenge.

The rest of this manuscript is organized in following manner: Section 2 defines literature review based on the weather app forecasting and Section 3 of the paper provides methodology. The results and the analysis of this research are described in Section 4 model view controller, API integration and result and discussions finally, Section 5 discuss the overall conclusion of this research.

II. LITERATURE REVIEW

[1] Mark Holmstrom, Dylan Liu, et.al concluded that both linear and functional regression did not perform as well as professional weather forecasting methods but in the longer run differences in their performances decreased, suggesting that over a longer period of time, Machine learning can indeed outperform professional and traditional methods. Linear regression is a low bias and high variance algorithm and hence its accuracy can be improved by collecting further data.

[2] Sanyam Gupta, Indumathy, et.al suggested and proposed an efficient and accurate weather prediction and forecasting model using linear regression concepts and normal equation model. All these concepts are a part of machine learning. The normal equation is a very efficient weather prediction modeland using the entities temperature, humidity and dew-point, it canbe used to make reliable weather predictions. This model also facilitates decision making in day to day life. It can yield better results when applied to cleaner and larger datasets.

ISSN (online): 2581-3048
Volume 8, Issue 6, pp 218-222, June-2024
https://doi.org/10.47001/IRJIET/2024.806029

[3] Aditya Grover, et.al in their work made a weather prediction model that predicts by considering the joint influence of key weather variables. They also made a kernel and showed that interpolation of space can be done by using GPS with such a kernel, taking into account various weather phenomena like turbulence. They also performed temporal analysis within a learner based on gradient tree and augmented the system using deep neural network.

[4] Muthu lakshmi A, ME (SE), et.al in their work proposed a methodology that aims at providing an efficient and accurate weather forecasting models to predict and monitor the weather datasets to predict rainfall. In the past, the parameters of weather were recorded only for the present time. But in the future, work will be done to make a working model of selection that can be used for classifying the framework for continuous monitoring of the climatic attributes

[5] Divya Chauhan, et.al made a comparison in their paper, which shows that the algorithms such as k-mean clustering and decision trees are well suited for mining data to predict future weather conditions. If we increase the size of the training set, the accuracy at first increases but then it slowly decreases after a particular period of time, depending on the size of the dataset.

[6] Piyush Kapoor, et.al concluded that if we perform comparison of weather condition variation by sliding window algorithm, the results are highly accurate except for the months of seasonal change. The results can be altered by changing the size of the window. Accuracy of the unpredictable months can be increased by increasing the window size to one month.

[7] Qing Yi Feng1, et.al have made a machine-learning toolbox which is based on climate data gathered from analysis and reconstruction of complex networks. It can also handle data containing multiple variables from these networks. The development of predictor models in the toolbox is dynamic and data-driven.

[8] Siddharth S. Bhatkande, et.al In their work the authors have used data mining technique and Decision tree algorithm as a means to classify weather parameters like maximum temperature, minimum temperature in terms of day, month and year.

[9] John K. Williams, et.al have shown in their work that a set of skill ful predictors for thunderstorm initiation can be identified by using the random forest machine learning algorithm. The random forest method can also be used to identify “regimes” in which they can improve the skill of the application by using forecast logic.

III. METHODOLOGY

The development of the Weather App involves integrating external weather APIs with Django to retrieve and display weather data. Django's Model-View-Controller (MVC) architecture provides a structured approach to building the application. The Model layer may include database models for caching weather data or user preferences. The View layer renders HTML templates for user interaction, while the Controller layer handles business logic, including querying weather APIs, processing data, and serving weather information to users.

This study employed a sequential approach to design and implement a weather application using the Django framework. Django was selected due to its robustness in handling complex web applications and its suitability for rapid development. The methodology involved several key stages: requirement analysis, system design, implementation, testing, and deployment.

During the requirement analysis phase, the essential features of a weather application were identified, including real-time weather updates, location-based forecasts, and user authentication. These requirements guided the system design, which focused on creating a scalable architecture capable of handling concurrent user requests and integrating with third-party APIs for weather data retrieval.

The implementation phase utilized Django’s MVC (Model-View-Controller) architecture to separate concerns and ensure maintainability. The backend was developed using Django’s ORM (Object-Relational Mapping) to interact with a PostgreSQL database, which stored user profiles, location data, and cached weather information. The frontend was designed with Django’s template engine to render dynamic HTML pages and JavaScript for asynchronous updates.

Testing involved unit testing for individual components and integration testing to validate the interaction between different modules. The application was deployed on a cloud platform (e.g., AWS) to ensure scalability and reliability. Continuous integration and deployment (CI/CD) practices were employed to automate the build and deployment processes, enhancing the development workflow.

Overall, this methodology facilitated the successful development of a weather application that provides accurate weather forecasts based on location data, leveraging Django’s strengths in web development and API integration.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 6, pp 218-222, June-2024

https://doi.org/10.47001/IRJIET/2024.806029

IV. MODEL VIEW CONTROLLER

In the context of a weather application built with Django, Model-view-controller (MVC) is a design pattern that separates the application into three interconnected components:

Model: Defines classes that represent data entities such as User, Location, and Weather Data. These models handle database operations (e.g., storing user profiles, caching weather forecasts) and encapsulate the application's business logic related to data management.

- **Responsibility:** Represents the data and business logic of the application. In a weather app, models would define how weather data is structured and stored in the database. For example, there might be models for User profiles, Location data, and Weather information.
- **Activities:** Models interact with the database through Django's ORM (Object-Relational Mapping), handling tasks such as saving user preferences, storing location coordinates, and caching weather forecasts.

View: Consists of HTML templates (using Django's template engine) that present information to users. Views render dynamically generated pages based on data retrieved from models. For instance, a view might display weather forecasts for a specific location or allow users to update their preferences.

- Responsibility: Handles the presentation layer of the application. Views in Django are typically represented by HTML templates rendered with data from the controllers and models.
- Activities: In a weather app, views would render pages where users can see weather information, manage their profiles, and interact with location-based forecasts. Views ensure that data from models is displayed in a user-friendly manner.

Controller: Implemented as Django views, which receive HTTP requests (e.g., GET, POST) from users via URLs defined in the application's URL configuration. Controllers fetch data from models, process user input, and render the appropriate templates to generate a response.

- **Responsibility:** Contains the application logic that binds the model and view together. In Django, controllers are represented by views, which receive HTTP requests from users, fetch data from models as needed, and render appropriate templates.
- **Activities:** Controllers (views) handle user interactions, process form submissions (e.g., updating user preferences or querying weather data), and coordinate data flow between models and views.

V. API INTEGRATION

Introduction to API Integration: API integration is fundamental to the functionality of the weather application, enabling real-time retrieval of weather data from external sources. This section outlines the process of selecting, integrating, and utilizing APIs within the Django framework to enhance the app's capabilities.

Selection of Weather APIs: The initial step in API integration involved selecting suitable weather APIs based on criteria such as data accuracy, geographic coverage, and available features. The chosen APIs, including OpenWeatherMap, were evaluated for their ability to provide comprehensive weather data required by the application.

Integration Steps:

Choose a Weather API: Select a weather API provider that suits your needs. Popular choices include OpenWeatherMap, or any other provider that offers weather data through an API.

Obtain API Key: Sign up on the chosen weather API provider's website to obtain an API key. This key will be required to authenticate your requests to the API.

VI. RESULT AND DISCUSSIONS

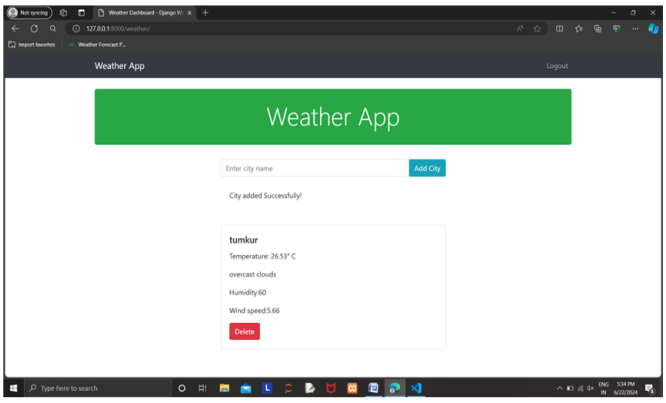

Figure 1: Sample images to enter the location

The Weather App is a web-based application that provides real-time weather information for various locations. Built using the Django framework, the app fetches weather data from a reliable third-party API and presents it in a user-friendly interface. Users can search for current weather conditions by entering the name of a city, and they will receive detailed information including temperature, humidity, wind speed, and weather conditions. Users can view weather forecasts for their current location or search for weather data for any specified location worldwide.

ISSN (online): 2581-3048
Volume 8, Issue 6, pp 218-222, June-2024
https://doi.org/10.47001/IRJIET/2024.806029

VII. CONCLUSION AND FUTURE SCOPE

In conclusion, the future of weather apps holds tremendous potential for innovation and enhancement.

By leveraging advancements in technology such as artificial intelligence, machine learning, IoT devices, and augmented reality, weather apps can evolve beyond simple forecast displays to become indispensable tools for users in various aspects of their lives.

These enhancements could include hyper-local forecasting for precise weather predictions, personalized recommendations for activities and health considerations, integration with smart home devices for automated responses to weather conditions, and even augmented reality overlays for immersive experiences.

Moreover, social integration and accessibility features can ensure that weather information is inclusive and fosters community engagement.

Future scopes of improvement in present methodologies are:

Weather Alerts and Notifications: Integrate real-time weather alerts and notifications to notify users about severe weather conditions in their area. This could be done through email, SMS, or push notifications.

Forecast data: Provide weather forecasts for next few days or hourly. We can add last 10 days time line of weather forecasting of the selected location as well.

Social Sharing: Add basic social sharing buttons so users can easily share current weather conditions or forecasts on popular social media platforms.

Multi-Language Support: Offer support for multiple languages to cater to a global audience. Allow users to select their preferred language for weather information and interface.

REFERENCES

[1] Jaseena, K. U., and Binsu C. Kovoor. "Deterministic weather forecasting models based on intelligent predictors: A survey." *Journal of king saud university-computer and information sciences* 34.6 (2022): 3393-3412.

[2] Biswas, M., Dhoom, T., & Barua, S. (2020). Weather forecast prediction: an integrated approach for analyzing and measuring weather data. *International Journal of Computer Applications*, 182(34), 20-24.

[3] A.G. Salman, B. Kanigoro and Y. Heryadi, "Weather forecasting using deep learning techniques," *2019 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2019, pp. 281-285.

[4] Sushmitha Kothapalli, S. G. Totad, "A Real-Time Weather Forecasting and Analysis", *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI-2020)*, pp 1567- 1570.

[5] Mr. Sunil Navadia, Mr. Jobin Thomas, Mr. Pintukumar Yadav, Ms. Shakila Shaikh, "Weather Prediction: A novel approach for measuring and analyzing weather data", *International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), (I-SMAC 2020)*, IEEE, pp 414-417.

[6] Amruta A. Taksande, P. S. Mohod, "Applications of Data Mining in Weather Forecasting Using Frequent Pattern Growth Algorithm", *International Journal of Science and Research (IJSR)*, Volume 4 Issue 6, June 2021, pp 3048-3051.

[7] Bitterman, A., M. Krocak, J. Ripberger, C. Silva, H. JenkinsSmith, S. Ernst, and S. Stormer, 2022: Maintaining the series: Public reception, understanding, and responses to severe weather forecasts and warnings in the contiguous United States. *University of Oklahoma Institute for Public Policy Research and Analysis*, 34 pp.

[8] Charness, N., and E. A. Bosman, 2019: The handbook of aging and cognition, F.I.M. Craik and T.A. Salthouse, Eds., *Lawrence Erlbaum*, 495–551.

[9] A.Gaviria-Pabón, A. Forney, and A. Bitterman, 2021: Continuing the series: Public reception, understanding, and responses to severe weather forecasts and warnings in the contiguous United States. *University of Oklahoma Center for Risk and Crisis Management*, 33 pp.

[10] Bitterman, A., M. Krocak, J. Ripberger, C. Silva, H. Jenkins Smith, S. Ernst, and S. Stormer, 2022: Maintaining the series: Public reception, understanding, and responses to severe weather forecasts and warnings in the contiguous United States. *University of Oklahoma Institute for Public Policy Research and Analysis*, 34 pp.

[11] Lazo, J. K., R. E. Morss, and J. L. Demuth, 2019: 300 billion served: Sources, perceptions, uses, and values of weather forecasts. *Bull. Amer. Meteor. Soc.*, 90, 785–798.

[12] Nix-Crawford, B., 2020: The trust factor between meteorologists and viewers and the effects on local television audience retention. *M.S. Thesis, College of Media Arts and Design, Drexel University*, 82 pp.

ISSN (online): 2581-3048
Volume 8, Issue 6, pp 218-222, June-2024
https://doi.org/10.47001/IRJIET/2024.806029

[13] Nunley, C., and K. Sherman-Morris, 2020: What people know about the weather. *Bull. Amer. Meteor. Soc.*, 101, E1225–E1240.

[14] Reuter, C., and T. Spielhofer, 2022: Towards social resilience: A quantitative and qualitative survey on citizens' perception of social media in emergencies in Europe. *Technol. Forecast. Social Change*, 121, 168–180.

[15] Perreault, M. F., J. B. Houston, and L. Wilkins, 2019: Does scary matter: Testing the effectiveness of new National Weather Service tornado warning messages. *Commun. Stud.*, 65, 484–499.

[16] Van Deursen, A. J. A. M., C. L. Bolle, S. M. Hegner, and P. A. M. Kommers, 2021: Modeling habitual and addictive smartphone behavior: The role of smartphone usage types, emotional intelligence, social stress, selfregulation, age, and gender. *Comput. in Human Behav.*, 45, 411–420.

[17] Krocak, M., J. Ripberger, C. Silva, H. Jenkins-Smith, S. Ernst, A. Bell, and J. Allen, 2020: Measuring change: Public reception, understanding, and responses to severe weather forecasts and warnings in the contiguous United States. *University of Oklahoma Center for Risk and Crisis Management*, 31 pp.

Citation of this Article:

Dr. Kavyashree N, & Lakshmi T L. (2024). A Survey on Weather App Forecasting Using Machine Learning. *International Research Journal of Innovations in Engineering and Technology - IRJIET*, 8(6), 218-222. Article DOI https://doi.org/10.47001/IRJIET/2024.806029

****

