IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Weather Forecasting Application With Ai Agent Using Lstm

An Intelligent Deep Learning-Based Weather Forecasting System with Conversational AI for Indian Cities

[1]Mukesh Kumar[st] Author, [2]Sonu Kumar[nd] Author, [3]Dhanraj Kumar Yadav[rd] Author

[4]Vikky Kumar[th] Author

[1]B-Tech CSE[st] Author, [2]B-Tech CSE[nd] Author, [3]B-Tech CSE[rd] Author , [4]B-Tech CSE[th] Author

[1]Department of Computer Science & Engineering[st] Author,

[1]Centurion University of Technology and Management, Paralakhemundi, Odisha, India

Abstract: Accurate weather forecasting is essential for effective planning in agriculture, transportation, disaster management, and daily human activities. This research presents an intelligent weather forecasting application that integrates deep learning-based time series prediction with an AI-powered conversational agent to deliver accurate and accessible weather insights. The proposed system utilizes a Long Short-Term Memory (LSTM) neural network to analyze historical meteorological data collected between 2015 and 2025 and predict key weather parameters, including temperature, precipitation, and humidity. The model performs weather condition classification such as sunny, rainy, and cloudy states. The framework supports forecasting for ten major Indian cities and provides predictions for up to 30 days ahead. An AI agent powered by Google Gemini enables natural language interaction, allowing users to request forecasts, compare cities, analyze trends, and obtain travel recommendations. The system is deployed using a Flaskbased web application with RESTful APIs for real-time access. Experimental results demonstrate an average classification accuracy of 82.62%, with consistent performance across diverse climatic regions.

Index Terms **-** Weather Forecasting, LSTM, Deep Learning, AI Agent, Google Gemini, Time Series, Flask, Multi-task Learning, Indian Cities, Natural Language Processing.

I. INTRODUCTION

Weather forecasting plays a vital role in modern society by supporting decision-making in sectors such as agriculture, transportation, disaster management, tourism, and urban planning. Accurate prediction of weather conditions helps individuals and organizations prepare for environmental changes, reduce risks caused by extreme weather events, and optimize daily activities. With increasing climate variability and unpredictable weather patterns, the demand for reliable and accessible forecasting systems has grown significantly.

Traditional weather forecasting methods are primarily based on numerical weather prediction (NWP) models that simulate atmospheric physics using complex mathematical equations. Although these systems provide accurate large-scale forecasts, they require high computational resources, specialized infrastructure, and expert interpretation. As a result, such approaches are often difficult to deploy in lightweight or user-centric applications. Recent advancements in artificial intelligence and deep learning have introduced data-driven alternatives capable of learning weather patterns directly from historical observations without explicit physical modeling.

Among deep learning techniques, Recurrent Neural Networks (RNNs) and particularly Long Short-Term Memory (LSTM) networks have demonstrated strong performance in time-series forecasting problems. LSTM models are designed to capture longterm temporal dependencies, making them highly suitable for weather data, which exhibits sequential and seasonal characteristics. By analyzing historical meteorological parameters such as temperature, precipitation, and humidity, LSTM models can generate accurate future predictions while adapting to nonlinear patterns in climate data.

This research proposes a Weather Forecasting Application with an AI Agent, combining deep learning-based forecasting with conversational intelligence. The system employs an LSTM neural network to perform multi-task prediction, including regression of meteorological variables and classification of weather conditions. The application supports forecasting for ten major Indian cities and provides predictions for up to 30 days ahead. Furthermore, an AI agent powered by Google Gemini allows users to request forecasts, compare weather across cities, analyze trends, and receive recommendations using natural language queries.

II. RELATED WORKS

Weather forecasting has been an active research area for several decades, evolving from traditional statistical techniques to advanced artificial intelligence and deep learning-based approaches. Early forecasting systems primarily relied on statistical and numerical methods such as autoregressive integrated moving average (ARIMA) models and numerical weather prediction (NWP). These approaches used mathematical representations of atmospheric dynamics to estimate future weather conditions. Although effective for large-scale predictions, such models required extensive computational resources and often struggled to capture nonlinear and complex temporal relationships present in meteorological data.

With the advancement of machine learning, researchers began applying algorithms such as Support Vector Machines (SVM), Decision Trees, and Random Forests for weather classification and rainfall prediction tasks. These models improved prediction performance compared to traditional statistical methods; however, they relied heavily on handcrafted features and were limited in modeling long-term temporal dependencies.

The emergence of deep learning significantly improved time-series forecasting capabilities. Long Short-Term Memory (LSTM) networks were proposed with gated memory mechanisms that allow the model to retain relevant historical information over extended periods. LSTM-based models have since been widely adopted for weather prediction tasks, including temperature forecasting, rainfall estimation, and climate trend analysis. Hybrid models combining CNNs and LSTMs have also been explored to capture both spatial and temporal features of weather data.

Recent research has also focused on multi-task learning frameworks that simultaneously perform regression and classification tasks, enabling models to predict both numerical weather parameters and categorical weather conditions. In parallel, conversational artificial intelligence has gained popularity for enabling natural language interaction with intelligent systems. The proposed research addresses these gaps by combining a lightweight LSTM-based forecasting model with an AI-powered conversational agent.

III. PROPOSED METHODOLOGY

The proposed Weather Forecasting Application integrates deep learning-based time series prediction with an intelligent AI agent to provide accurate, scalable, and interactive weather forecasting. The methodology focuses on learning temporal weather patterns from historical data and delivering predictions through an accessible web-based system. The overall framework consists of data preprocessing, LSTM model development, multi-task prediction, AI agent integration, and system deployment.

A. Dataset Collection and Description

The model is trained using historical weather data collected from 2015 to 2025 for ten major Indian cities. The dataset contains daily meteorological observations representing diverse climatic conditions such as coastal, tropical, and semi-arid environments. The primary features used include T2M (Temperature at 2 meters), PRECTOTCORR (Corrected Precipitation), QV2M (Specific Humidity), and Weather Condition Labels derived as categorical classes such as Sunny, Rainy, or Cloudy. Each city dataset is processed independently to build specialized forecasting models capable of capturing localized weather patterns.

B. Data Preprocessing and Feature Engineering

Raw weather data requires preprocessing to ensure consistency and improve model learning capability. Time-series data were arranged chronologically to preserve temporal dependencies. Missing observations were filled using interpolation techniques to maintain continuity. Feature values were scaled using Min-Max normalization to stabilize neural network training. A fixed-length historical window of 30 days was used as input to predict future weather conditions. The dataset was split into Training Set (70%), Validation Set (15%), and Test Set (15%) to ensure fair evaluation and prevent overfitting.

C. LSTM-Based Forecasting Model

To capture temporal dependencies in weather data, a Long Short-Term Memory (LSTM) neural network was implemented. The proposed architecture consists of: Input Layer (30-day historical weather sequence), Bidirectional LSTM Layer (128 units), LSTM Layer (64 units), Dense Layer (64 neurons with ReLU activation), and Dropout Layer (0.2 to reduce overfitting). The network learns temporal relationships between historical weather variables and future atmospheric conditions. The model performs two prediction tasks simultaneously: regression prediction for continuous variables (temperature, precipitation, humidity) and classification prediction for weather condition categories using a Softmax output layer.

D. Training Configuration

The model was trained using the Adam optimizer with a learning rate of 0.001, batch size of 64, and a forecast horizon of 1 day (iteratively extended to 30-day forecasts). Early stopping with patience of 5 epochs was employed along with a learning rate scheduler. The total loss function combines regression loss (Mean Squared Error) and classification loss (Categorical Cross-Entropy) in a joint optimization objective.

E. AI Agent Integration

An intelligent AI agent enhances system usability by enabling natural language interaction. The agent is implemented using the LangChain framework, LangGraph state management, and Google Gemini large language model. The agent follows the ReAct (Reasoning + Acting) paradigm. Upon receiving a user query, the agent interprets intent through reasoning steps, selects the appropriate forecasting tool, retrieves model predictions, and generates human-readable responses. Agent capabilities include weather forecasting queries, city comparison, trend analysis, travel recommendations, and statistical summaries.

F. System Deployment

The complete system was deployed as a web-based application using Flask framework for the backend, TensorFlow and Keras for deep learning, Pandas and NumPy for data processing, and HTML/CSS/JavaScript for the frontend. RESTful API endpoints support: retrieving available cities, generating next-day predictions, producing 30-day forecasts, and AI conversational interaction. The application runs efficiently on standard CPU systems without requiring specialized hardware.

IV. RESULTS AND DISCUSSIONS

The proposed Weather Forecasting Application was evaluated to measure its effectiveness in predicting meteorological parameters and classifying weather conditions across multiple cities. The experiments focused on assessing forecasting accuracy, model generalization, multi-city performance consistency, and the usability improvements introduced by the AI agent integration.

A. Weather Classification Results

The classification module predicts categorical weather conditions such as Sunny, Rainy, and Cloudy. Experimental results show stable performance across all cities. The LSTM model achieved an average classification accuracy of 82.62%, precision of 83.32%, recall of 82.62%, and F1-Score of 82.34%. Among all cities, Chennai achieved the highest accuracy (89.35%), likely due to relatively stable seasonal patterns compared to highly variable inland climates. Minor classification confusion occurred between visually similar atmospheric conditions such as cloudy and partly cloudy days, which is common in weather prediction tasks.

B. Regression Forecasting Performance

The regression component predicts continuous meteorological variables including Temperature (T2M), Precipitation (PRECTOTCORR), and Humidity (QV2M). The predicted values closely followed real observations, indicating successful temporal learning by the LSTM network. Temperature forecasts showed the lowest prediction error due to strong seasonal trends. Precipitation prediction exhibited slightly higher variability because rainfall patterns are inherently stochastic. Humidity predictions remained consistent across coastal cities. Low RMSE values confirmed that predicted outputs maintained close alignment with actual measurements.

C. Multi-City Performance Analysis

The system was evaluated across ten Indian cities representing diverse climatic zones: Coastal (Chennai, Mumbai), Inland (Delhi, Jaipur), Tropical (Bengaluru, Hyderabad), and Eastern region (Kolkata, Bhubaneswar). Accuracy across cities ranged between 78.70% and 89.35%, demonstrating strong generalization capability. The model successfully adapted to regional weather variations without requiring architectural modifications, confirming that the proposed framework scales efficiently for multi-location forecasting applications.

D. Long-Term Forecast Evaluation

Recursive prediction was used to generate forecasts up to 30 days ahead. Results indicate high reliability for short-term predictions (1-10 days), with a gradual increase in prediction uncertainty beyond 20 days, which is expected in time-series forecasting. Overall trend consistency was maintained across the forecasting horizon. Despite long-range prediction challenges, the system preserved seasonal trends and temperature trajectories effectively.

E. AI Agent Performance Analysis

The integration of the AI agent significantly enhanced user interaction and system usability. The agent successfully performed natural language forecast queries, city-to-city weather comparisons, trend analysis and summaries, and travel recommendations based on predicted conditions. The ReAct-based reasoning mechanism allowed the agent to autonomously select forecasting tools and generate context-aware responses. User interaction testing demonstrated that conversational access reduced complexity compared to traditional dashboard-based systems.

V. CONCLUSION

This research presented a comprehensive Weather Forecasting Application integrated with an AI Agent, combining deep learningbased time series prediction with conversational artificial intelligence to provide accurate and accessible weather insights. The proposed framework utilizes a Long Short-Term Memory (LSTM) neural network to analyze historical meteorological data and predict key weather parameters including temperature, precipitation, humidity, and overall weather conditions. By adopting a multitask learning approach, the system efficiently performs both regression and classification within a unified model architecture.

Experimental evaluation across ten major Indian cities demonstrated consistent forecasting performance, achieving an average classification accuracy of 82.62% with stable precision, recall, and F1-score values. The model successfully captured temporal weather patterns and generalized well across diverse climatic regions, confirming its robustness and scalability. Furthermore, the recursive forecasting strategy enabled predictions for up to 30 days while maintaining realistic seasonal trends.

A significant contribution of this work is the integration of an intelligent AI agent powered by Google Gemini, which allows users to interact with weather forecasts through natural language queries. The agent enhances usability by providing automated reasoning, city comparisons, trend analysis, and travel recommendations. Overall, the results validate that integrating deep learning forecasting with conversational AI creates an intelligent, scalable, and user-friendly weather forecasting solution suitable for real-world deployment.