Architecture Components

Flight Fare Prediction

Develop By:

Vijay Kumar Shah

Date:

03/09/2024

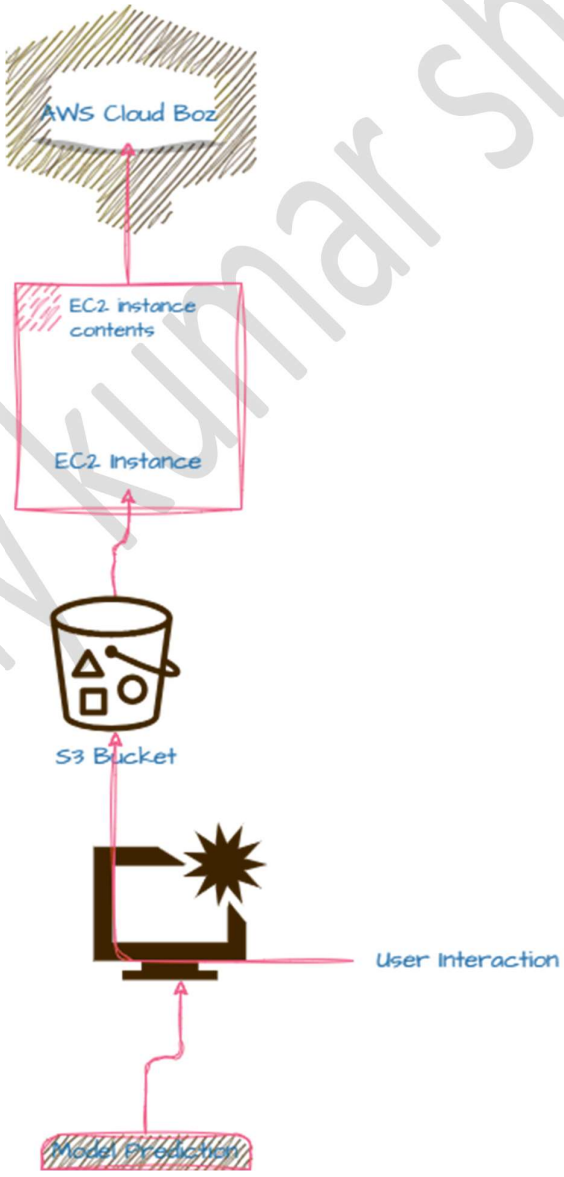
Writer	Vijay Kumar Shah
Document Version	0.2
Last Revised Date	03/09/2024

Architecture

1. **AWS Cloud**:
    - All the services are hosted within the AWS environment.
    - The cloud provides security, scalability, and reliability.
2. **Flask Application**:
    - **EC2 Instance**: Host the Flask web application on an EC2 instance. This is the server where the web app runs.
    - **S3 Bucket**: Store static files (like home.html).
    - **RDS (Relational Database Service)**: If you use a database, it can be hosted on RDS.
3. **User Interaction**:
    - **Browser (User Interface)**: The user interacts with the web application via a browser, which sends HTTP requests.
4. **Load Balancer (Optional)**:
    - For distributing incoming traffic across multiple EC2 instances.
5. **API Gateway (Optional)**:
    - To securely expose the Flask application to the web.
6. **Machine Learning Model**:
    - **S3 Bucket**: Store the trained model (flight_rf.pkl) in an S3 bucket.
    - The model is loaded into the Flask application on the EC2 instance.
7. **Application Logic**:
    - The Flask app processes incoming requests and interacts with the machine learning model to generate predictions.

Architecture

8. **Client (User) Request**:
    - **HTML Form Submission**: Users submit flight details via an HTML form (via the Flask app), and these details are sent to the Flask server.
    - **Prediction Response**: The prediction is processed and returned as an HTTP response.
9. **Visualization & Deployment**:
    - **CloudFront** (Optional): CDN for faster content delivery.
    - **Route 53** (Optional): Domain name routing.



Architecture

Conclusion

This flight price prediction system, developed using Flask as the web framework and a Random Forest model for prediction, demonstrates the integration of machine learning with web development to create an intuitive and user-friendly application. By leveraging data related to airlines, source and destination locations, stops, and flight times, the model can accurately predict flight prices based on user inputs.

The application successfully implements a machine learning model that was trained on a substantial dataset, which ensures that the predictions are reliable and relevant to real-world scenarios. By deploying this model on a Flask-based web application, users can easily access the prediction service, making it highly scalable and accessible from anywhere.

This project highlights the potential of combining various technologies like Python, Pandas, scikit-learn, and Flask to solve practical problems in the aviation industry. In addition, the deployment of the application in the cloud, along with the utilization of AWS services, further demonstrates how modern cloud infrastructure can be effectively employed to host and manage machine learning models.

Future work could focus on enhancing the model's performance by incorporating more features, refining the prediction algorithm, or expanding the scope to cover additional regions and airlines. Overall, this project provides a robust foundation for future enhancements in flight price prediction systems, showcasing the power of machine learning in creating impactful