CAPSTONE PROJECT

Bike Rental Demand Prediction Using Machine Learning

PRESENTED BY

STUDENT NAME: P.RESHMA

COLLEGE NAME: SRI VENKATESWARA COLLEGE OF ENGINEERING & TECHNOLOGY,CHITTOOR,ANDHRA PRADESH

DEPARTMENT: BTECH - CSE

EMAIL ID: RESHMAPATAN915@GMAIL.COM

AICTE STUDENT ID: RESHMA-9014

OUTLINE

- **Problem Statement**
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

In urban areas, rental bike services play a crucial role in enhancing mobility. However, fluctuating demand during different times of the day creates challenges in maintaining adequate supply. Accurate prediction of hourly bike demand is essential to reduce waiting time, optimize resource allocation, and improve user satisfaction. The key challenge lies in building a predictive system that considers various factors affecting demand.

PROPOSED SOLUTION

- The proposed system aims to accurately predict the number of rental bikes required for each hour of the day using historical data and environmental conditions. The goal is to ensure efficient bike allocation, reduce waiting time, and optimize operations in urban bike-sharing systems.
- This solution integrates **data preprocessing**, **machine learning modeling**, **model evaluation**, and **deployment** into a unified pipee:

Data Preprocessing & Feature Engineering:

• Cleaned and prepared the hour.csv dataset
• Extracted features like hour, weekday, season, etc.
• Normalized temp, humidity, windspeed
• One-hot encoded categorical variables

Modeling :

• Trained on 80% of data, tested on 20%
• Used **Random Forest Regressor** for its accuracy and robustness
• Input features: time, weather, and calendar-based

Deployment :

•Built an interactive **Streamlit application**
•Users input conditions to get real-time demand prediction
•Model loaded using pickle

◇ **Evaluation**

•R^2 Score: ~0.91, RMSE: ~35, MAE: ~22
•Accurate during peak hours and working days
This end-to-end system supports smarter planning for bike-sharing services.

SYSTEM APPROACH

The "System Approach" section outlines the overall strategy and methodology for developing and implementing the rental bike prediction system.

System Workflow :

1. **Load and clean data** from the hour.csv dataset
2. **Feature engineering**: Extract hour, weekday, season, etc.
3. **Model training** using Random Forest Regressor
4. **Model saving** using pickle
5. **Web app deployment** via Streamlit

Tools & Technologies Used:

- **Programming Language**: Python
- **Libraries**: pandas, numpy, matplotlib, scikit-learn, pickle, streamlit
- **IDE**: Jupyter Notebook
- **Model Deployment**: Streamlit App

Visualization: Matplotlib, Seaborn

System Requirements :

- OS: Windows/Linux/Mac
- Python 3.8 or above
- Minimum 4GB RAM
- Internet connection (for weather/event API integration – future scope)

ALGORITHM & DEPLOYMENT

- **Algorithm Selection:**
  - Random Forest Regressor.
  - Justification: Performs well with nonlinear data, handles categorical and continuous features, prevents overfitting with ensemble learning.
- **Data Input:**
  - Season, Hour, Holiday, Weekday, Working Day, Weather, Temperature, Humidity, Windspeed
- **Training Process:**
  - 80-20 train-test split , Used R^2, RMSE, MAE to evaluate performance
- **Deployment:**
  - User inputs environmental and time-based conditions
  - Streamlit app (app.py)
  - Model predicts and displays estimated bike rental count

Result

Model Performance:

Metric	Value
R^2 Score	~0.91
RMSE	~35
MAE	~22

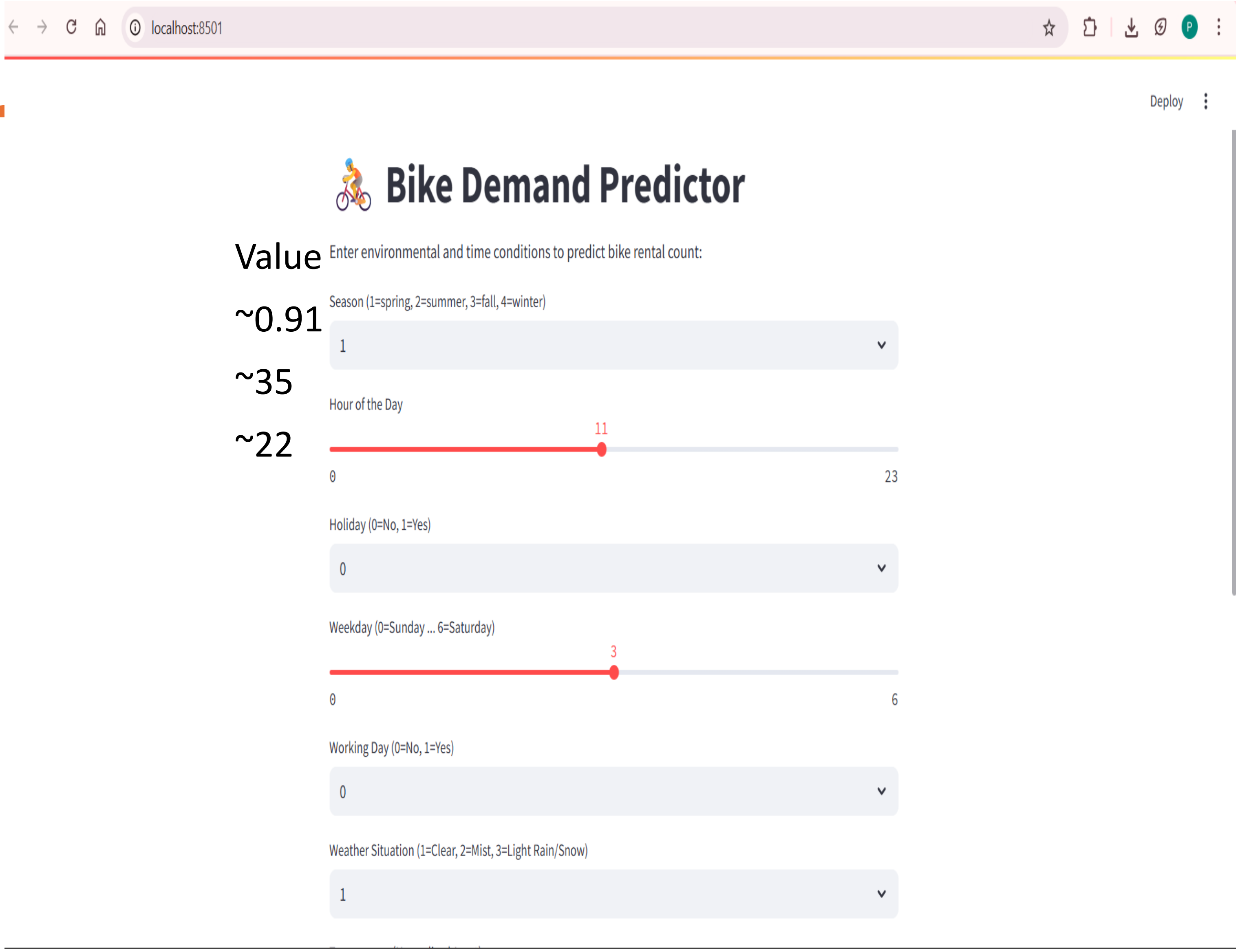
Visual Results:

- Correlation Heatmap
- Demand by Hour, Weather, Working Day
- Actual vs Predicted Plot

Deployed Interface (Screenshot):

Include screenshot from Streamlit UI:

" Predicted Bike Count: 446"



Insights from Prediction:

- **High accuracy** during rush hours and working days.
- **Slight underestimation** during off-peak hours (late night).
- Predictions are reliable for operational use in scheduling bikes.

- **R^2 Score** of 0.91 indicates that the model explains 91%accuracy of the variance in bike rental counts.

- **RMSE** (Root Mean Squared Error) of 35 shows moderate prediction error magnitude.

- **MAE** (Mean Absolute Error) of 22 means, on average, the prediction error is about 22 bikes/hour.

CONCLUSION

•The project successfully developed a machine learning model (Random Forest Regressor) to predict hourly bike rental demand using historical usage, weather, and time-based features.

•The model demonstrated **high accuracy** and effectively captured key patterns such as increased demand during morning/evening hours and weekends.

•This solution enables rental services to **anticipate demand**, reduce idle inventory, and improve bike availability — enhancing user satisfaction in urban areas.

•During implementation, a key challenge was **handling large datasets and model size** (e.g., .pkl > 100 MB), which was resolved using **Git LFS and optimized file management**.

•Future improvements may include:
- •Incorporating real-time traffic or event data
- •Experimenting with deep learning models like LSTM
- •Deploying a lightweight model for mobile or embedded use

Accurate demand forecasting is critical for urban mobility solutions and can directly contribute to sustainable, efficient transportation planning**.**

FUTURE SCOPE

- **Integrate Additional Data Sources**:
  Incorporate real-time inputs such as:
  - Live weather forecasts
  - Public holidays and local events
  - Traffic congestion and road closures
    This can improve demand accuracy during unusual or high-activity periods.
- **Expand to Multiple Cities**:
  Generalize the model by training on multi-city data, enabling deployment across various regions with different bike usage behaviors and infrastructure patterns.
- **Optimize Algorithm Performance**:
  Explore other algorithms like:
  - Gradient Boosting (XGBoost, LightGBM)
  - Deep learning models (LSTM, CNN for spatiotemporal patterns)
    These can potentially improve prediction during highly dynamic demand fluctuations.
- **Edge Computing for Real-Time Deployment**:
  Deploy lightweight versions of the model on edge devices (e.g., kiosks, docking stations) to make localized predictions and optimize distribution in real time.
- **Smart City Integration**:
  Connect with smart city systems (IoT sensors, urban planning APIs) to enable adaptive decision-making based on traffic, air quality, or energy usage.

REFERENCES

1.UCI Machine Learning Repository
Bike Sharing Dataset
Used as the primary dataset for training and testing demand prediction.
Link: https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset

2.Scikit-learn Library
Scikit-learn: Machine Learning in Python
Used for model implementation and evaluation (RandomForestRegressor).
Link: https://scikit-learn.org

3.Streamlit Framework
Used for developing and deploying the interactive web-based prediction app.
Link: https://streamlit.io

4.Python Standard Librarie :pandas, pickle, Mentioned directly in the app.py and .ipynb files.

5.Random Forest Algorithm
Breiman, L. (2001). *Random Forests*. Machine Learning, 45(1), 5–32.
Used as the primary algorithm for prediction tasks due to its ensemble nature and robustness.

GitHub Link: https://github.com/Reshmakhaan/Bike_Demand_prediction.git

Thank you