Waiters Tip Prediction Using Machine Learning

[1]MAMIDI TARANI, [2]ADARI JAYANTH

[1]Assistant Professor, [2]MCA Final Semester,

[2]Master of Computer Applications,

[2]Sanketika Vidya Parishad Engineering College, Vishakhapatnam, Andhra Pradesh, India

ABSTRACT
This project aims to predict the amount of tip a customer might give at a restaurant using machine learning. We use a Linear Regression model that is trained on real restaurant data. The model takes various input features such as the total bill amount, the gender of the customer, whether the customer smokes or not, the day of the week, the time of the meal (lunch or dinner), and the size of the group. By analyzing these features, the model learns how each factor influences the tip amount. Once trained, it can predict the tip based on new customer information. This prediction can help restaurants in many ways. It can improve customer service by understanding which situations lead to better tips, support managers in making better decisions using data, and help staff performance by identifying what impacts tipping behavior. It can also be used in automated systems to suggest expected tips, making the billing process smarter. Overall, this project shows how machine learning can be applied in the food industry to understand customer behavior and enhance business strategies
IndexTerms: Tip Prediction, Linear Regression, Machine Learning, Restaurant Analytics, Customer Behavior, Bill Amount, Smoking Status, Mealtime (Lunch/Dinner), Party Size, Data-Driven Insights, Service Improvement, Predictive Modeling, Restaurant Management.

1.INTRODUCTION

The project titled *"Waiters Tip Prediction Using Machine Learning"* is designed to estimate the amount of tip a customer might leave in a restaurant setting. It utilizes a machine learning approach, specifically a Linear Regression model, trained on real-world restaurant data. The model considers multiple factors such as total bill amount, customer gender, smoking status, meal time (lunch or dinner), day of the week, and the group size. These features are preprocessed using one-hot encoding to match the format used during training. A web interface, built using Flask, allows users to input these variables and receive an instant prediction of the expected tip amount.[15] The backend loads a pre-trained model (tip_predictor_model.pkl) and processes inputs to maintain consistency in prediction accuracy. This application not only assists restaurant staff in understanding customer tipping behavior but also supports managers in making data-driven decisions to improve service. By identifying trends in tipping based on group size, day, or meal time, staff can be allocated and prepared accordingly.[12] Moreover, the prediction system can be integrated into billing systems to suggest optimal tip values. The project highlights the relevance of predictive modeling in the hospitality sector. It demonstrates how machine learning can streamline operations, personalize service, and boost employee motivation. Ultimately, the system serves as a practical and intelligent assistant for modern restaurant management. This combination of data science and service analytics is a step forward in digital transformation in dining experiences.[9]

1.1 Existing System

In the existing system, restaurants typically do not use predictive technologies to estimate customer tips. Instead, tipping behavior is left to the discretion of the customer and influenced by factors such as service quality, personal preference, and social norms. There is no structured mechanism to analyze how various factors like meal time, customer demographics, or group size affect tipping.[8] Moreover, restaurant management lacks tools to utilize historical data for forecasting or strategy building. Waitstaff performance is often evaluated without clear data on how service impacts tipping. Billing systems operate independently without any intelligent recommendation for tip amounts. As a result, opportunities to enhance customer satisfaction and optimize staffing are missed. The absence of automated predictions limits operational efficiency. Additionally, valuable insights remain untapped due to the lack of integration between customer behavior and billing records. There is no web interface or backend model in place to simulate or learn from past tipping data. Hence, the existing system is manual, reactive, and lacks data-driven capabilities. This highlights the need for a machine learning-based solution that can fill these gaps.[14]

1.1.1 Challenges:

- No prediction mechanism to estimate customer tips based on previous data.
- Waitstaff performance is evaluated without linking it to tipping behavior.
- Lack of analysis tools to identify patterns based on group size, gender, day, or meal time.
- Billing systems do not suggest tip amounts, missing opportunities to guide customers.
- Customer behavior trends related to tipping are not tracked or utilized.
- Restaurants cannot forecast high or low tipping periods due to lack of data insights.
- Entire process is manual and lacks intelligent automation.
- No centralized or structured data collection for tip-related analysis.
- Absence of real-time prediction tools for operational use.
- Limited or no integration of machine learning into restaurant decision-making.
- Inability to improve service strategies through predictive analytics.
- Staff scheduling is not optimized due to missing insight on tipping trends.

1.2 Proposed system:

The proposed system introduces a machine learning-based solution to predict the tip amount a customer may give at a restaurant. Using a Linear Regression model, it analyzes key input features such as total bill, customer gender, smoking status, day of the week, time of the meal, and party size. These features are processed using one-hot encoding to train the model accurately. A user-friendly web application, developed using Flask, allows restaurant staff to input this data and receive instant tip predictions.[5] The backend loads the trained model and ensures the input data format matches the training schema for precise output. By integrating this system into restaurant operations, management can gain insights into tipping behavior and make informed decisions. The model also helps predict customer generosity during specific days or time slots, supporting better staff allocation. Automated tip suggestions during billing can improve transparency and customer satisfaction. The system promotes data-driven strategies to boost performance. It modernizes restaurant analytics and replaces guesswork with reliable predictions. Ultimately, it transforms tip estimation into a smart, efficient, and real-time process.[11]

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 07 | July – 2025 DOI: 10.55041/ISJEM04916
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

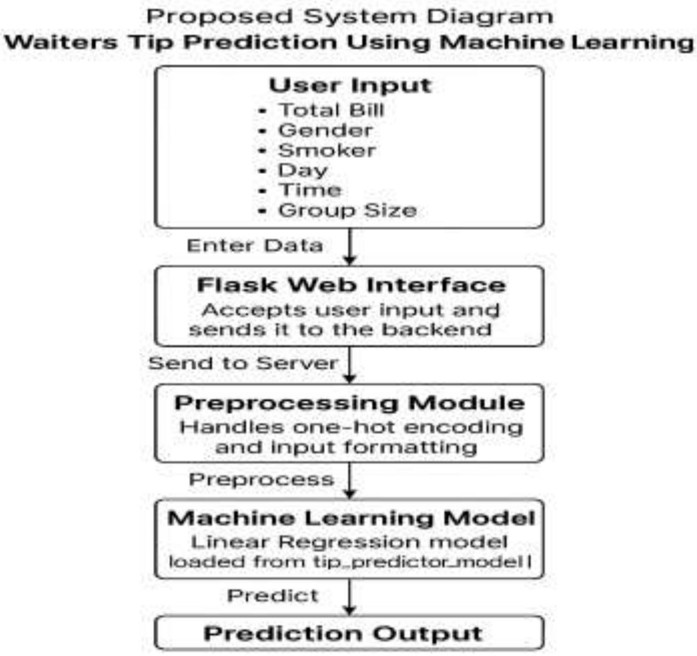

Fig: 1 Proposed Diagram

1.2.1 Advantages:

Accurate Tip Predictions

- Uses a trained machine learning model to predict tip amounts based on real input data.

Improved Customer Insight

- Identifies how customer behavior (e.g., meal time, group size) impacts tipping patterns.

Data-Driven Decision Making

- Helps restaurant managers make informed decisions based on tip prediction trends.

Optimized Staff Scheduling

- Allows better staff allocation during high or low tipping periods.

Real-Time Output

- Delivers instant tip predictions via a user-friendly web interface.

Enhanced Service Quality

- Enables restaurants to personalize service based on predictive insights.

Automated System

- Reduces manual effort and guesswork with intelligent automation.

Scalable and Extendable

- Can be integrated with billing systems or extended with historical logging and dashboards.

2.1 Architecture:

The architecture of the Waiters Tip Prediction System is built as a web-based machine learning application. It is composed of four major layers: Presentation Layer, Application Layer, Model Layer, and Storage Layer. These layers work together to take user input, preprocess it, predict the tip amount using a trained model, and display the result.[13]

1. Presentation Layer (Front-End)

- Built using HTML (rendered by Flask)
- Allows users (restaurant staff) to enter:
    - Total Bill
    - Gender

ISSN: 2583-6129
DOI: 10.55041/ISJEM04916

  - Smoker Status
  - Day of the Week
  - Time of Meal
  - Party Size
- Sends input data to the server via POST request

2. Application Layer (Flask Web Server)

- Developed using Python with Flask framework
- Accepts input data from the front-end
- Calls the preprocessing module to transform raw data into model-ready format
- Sends formatted data to the ML model for prediction
- Receives prediction result and sends it back to the front-end for display

3. Model Layer (Machine Learning Logic)

- Uses a Linear Regression model trained on restaurant tipping data
- Model is stored in a .pkl file (tip_predictor_model.pkl)
- Includes a feature_columns.pkl file to ensure input consistency with training schema
- Outputs a numerical value predicting the tip amount

4. Storage Layer

- Includes serialized model files:
  - tip_predictor_model.pkl (trained model)
  - feature_columns.pkl (feature schema used during training)
- Can be extended to include a database for storing predictions and inputs (optional future scope)

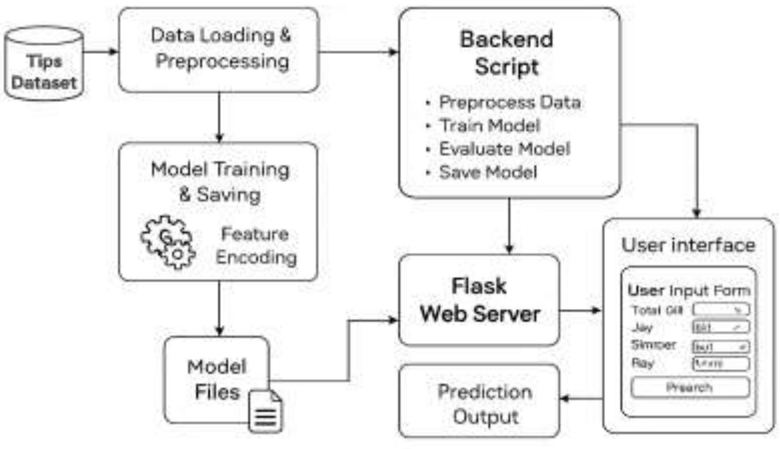

Fig:2 Architecture

UML DIAGRAMS

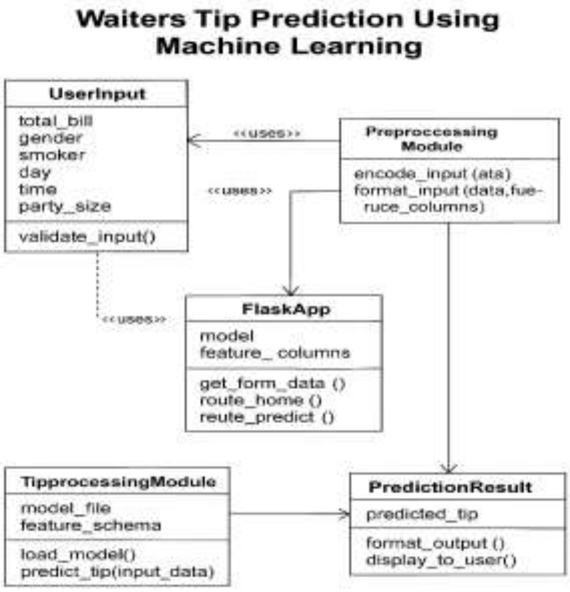

Fig 3: Class diagram

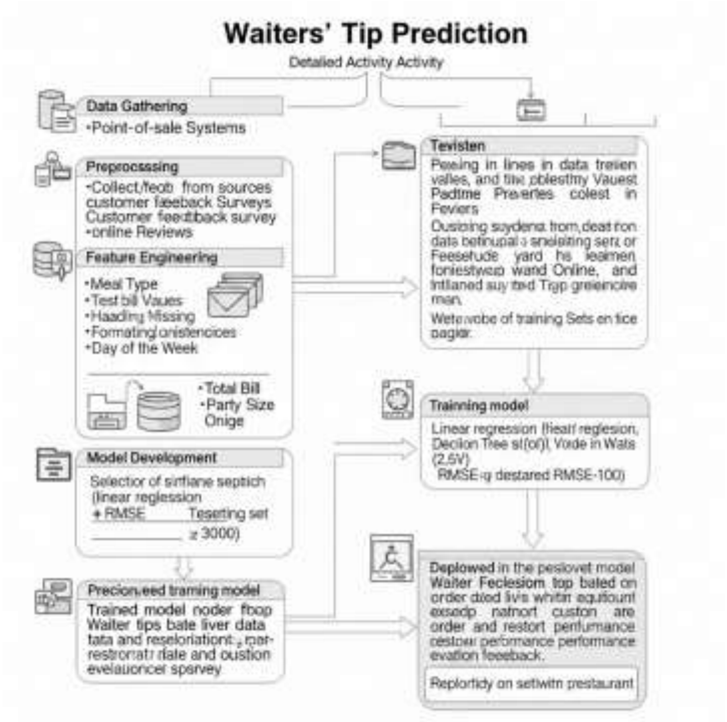

Fig 4: Activity diagram

2.2 Algorithm:

Linear Regression Algorithm

Linear Regression is a fundamental machine learning algorithm used for predicting a numeric (continuous) value based on one or more input features. In your project, it helps predict the tip amount a customer might leave based on several factors like total bill amount, gender, smoking status, day of the week, meal time, and party size. The algorithm works by finding a mathematical relationship between the input features (independent variables) and the target value (dependent variable – in this case, the tip). It does this by fitting a straight line (or hyperplane for multiple variables) that minimizes the difference between the predicted and actual tip values using a method called least squares. Each feature is assigned a weight (coefficient) that shows how much it contributes to the tip. For example, if the "total bill" has a higher weight, it means the bill strongly influences the tip. This model is trained on historical data, and once trained, it can predict the tip for new customer data by applying the learned equation. Linear Regression is simple, fast, and works well when the relationship between inputs and output is linear and interpretable, making it a perfect fit for restaurant-based tip prediction systems.[10]

2.3 Techniques:

- **Supervised Learning**

The project uses supervised learning, where the model is trained on historical data with known tip values. It learns from this labeled data to make future predictions.[6]

- **Linear Regression**

This is the core machine learning technique used. It models the relationship between input variables (like bill amount, gender, etc.) and the output variable (tip amount).

- **One-Hot Encoding**

Categorical inputs such as day, time, gender, and smoker status are converted into a numerical format using one-hot encoding so that the model can process them effectively.[4]

- **Feature Engineering**

Input features are selected and formatted to match the structure the model expects. This includes encoding and organizing them based on the original training schema (feature_columns.pkl).

- **Model Serialization (Pickling)**

The trained model is saved using Python's pickle module as tip_predictor_model.pkl, allowing it to be reused in the Flask web app for predictions without retraining.

- **Flask Web Framework**

Flask is used to build a lightweight web interface that accepts user inputs and displays predicted tip results, enabling real-time interaction with the model.[7]

2.4 Tools:

- **Python**

The main programming language used to build the machine learning model and the web application.

- **Pandas**

A Python library used for data handling and manipulation (e.g., loading, formatting, and preprocessing tabular data).

- **NumPy**

Used for handling numerical data and array operations needed during preprocessing and prediction.

- **Scikit-learn** (sklearn)

The core machine learning library used to train the Linear Regression model and perform encoding like one-hot encoding.

- **Pickle**

Used for saving (.pkl) the trained model and feature column list, so they can be reused without retraining.

- **Flask**

A lightweight web framework used to create the web interface (server, routing, form handling, etc.) for tip prediction.

- **HTML**

Used to design the front-end web form where users enter input values like bill amount, gender, etc.

- **Jupyter Notebook (.ipynb)**

Used during development for training, testing, and validating the machine learning model interactively

ISSN: 2583-6129
DOI: 10.55041/ISJEM04916

2.5 Methods:

- **Data Collection**

A restaurant dataset is used that contains details like total bill, gender, smoker status, meal time, day of the week, and party size, along with the tip amount.[3]

- **Data Preprocessing**

Raw data is cleaned and transformed. Categorical values (like gender, day, etc.) are converted to numerical form using one-hot encoding to make them suitable for machine learning.

- **Feature Selection**

Only relevant input features are selected for training the model, such as total_bill, size, gender, smoker, day, and time.

- **Model Training**

A **Linear Regression** model is trained using the preprocessed dataset to learn the relationship between the input features and the tip amount.[1]

- **Model Evaluation**

The trained model is evaluated (likely using metrics like R^2 score or Mean Squared Error) to ensure it predicts accurately before being saved.

- **Model Serialization**

The trained model and feature structure are saved as .pkl files using Python's pickle module for future use in the web app.[2]

3. METHODOLOGY

3.1 Input:

The input information for the project consists of several features collected from restaurant customers, which are used to predict the tip amount. These inputs include the total bill amount, which is the overall cost of the customer's meal, and the size of the group, indicating how many people were dining together. Additional categorical inputs include the gender of the customer (Male or Female), their smoking status (Yes or No), the day of the week when the meal occurred (Thur, Fri, Sat, or Sun), and the time of the meal (Lunch or Dinner). These values are entered by the user through a web form built using HTML and Flask. Before feeding the data into the model, all categorical inputs are converted into numerical format using one-hot encoding. These features serve as the independent variables for the machine learning model, which uses them to predict the target variable — the tip amount.

```python
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
print(f"Model Mean Squared Error (MSE) on test set: {mse:.2f}")
print(f"Model Root Mean Squared Error (RMSE) on test set: {rmse:.2f}")

# --- 4. Save the Trained Model and Feature Columns ---
# These files will be saved in the Colab environment's current directory
with open('tip_predictor_model.pkl', 'wb') as file:
    pickle.dump(model, file)
print("Model saved as tip_predictor_model.pkl")

with open('feature_columns.pkl', 'wb') as file:
    pickle.dump(feature_columns, file)
print("Feature columns saved as feature_columns.pkl")

print("--- Model Training Script Finished ---")
```

```
--- Starting Model Training Script ---
Tips dataset loaded.
DataFrame head:
   total_bill   tip     sex smoker  day    time  size
0       16.99  1.01  Female     No  Sun  Dinner     2
1       10.34  1.66    Male     No  Sun  Dinner     3
2       21.01  3.50    Male     No  Sun  Dinner     3
3       23.68  3.31    Male     No  Sun  Dinner     2
4       24.59  3.61  Female     No  Sun  Dinner     4
Categorical features encoded.
DataFrame head after encoding:
   total_bill   tip  size  sex_Female  smoker_No  day_Fri  day_Sat  day_Sun  \
0       16.99  1.01     2        True       True    False    False     True
1       10.34  1.66     3       False       True    False    False     True
2       21.01  3.50     3       False       True    False    False     True
3       23.68  3.31     2       False       True    False    False     True
4       24.59  3.61     4        True       True    False    False     True
```

Fig: tips dataset

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 07 | July – 2025 DOI: 10.55041/ISJEM04916
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

3.2 Method of Process:

The process in this project begins with the user entering input data through a web form, which includes fields such as total bill, gender, smoker status, day of the week, time of meal, and group size. Once the data is submitted, it is sent to the Flask web server, where the application receives and validates the input. Next, the data is passed to a preprocessing module, where categorical variables are converted into numerical format using one-hot encoding, and the input is arranged according to the trained model’s feature structure. This processed data is then fed into the pre-trained Linear Regression model, which uses the learned weights to predict the tip amount. The predicted value is then returned to the Flask application, which sends it back to the web interface, where the user can view the result. This end-to-end process combines form handling, data transformation, machine learning prediction, and result display in a seamless real-time system.

3.3 Output:

The main output of the project is the predicted tip amount that a customer is likely to give based on the input features provided through the web form. After processing the inputs (such as total bill, group size, gender, smoker status, day, and time), the machine learning model — specifically, a Linear Regression model — generates a numerical prediction representing the expected tip. This prediction is then displayed on the web interface in a user-friendly format (e.g., "Predicted Tip: ₹35.75"). The output is dynamic, meaning it changes based on the user's input values. It allows restaurant staff or management to instantly see how various customer factors influence tipping behavior, making the system practical for real-time usage in operational decision-making or service planning.

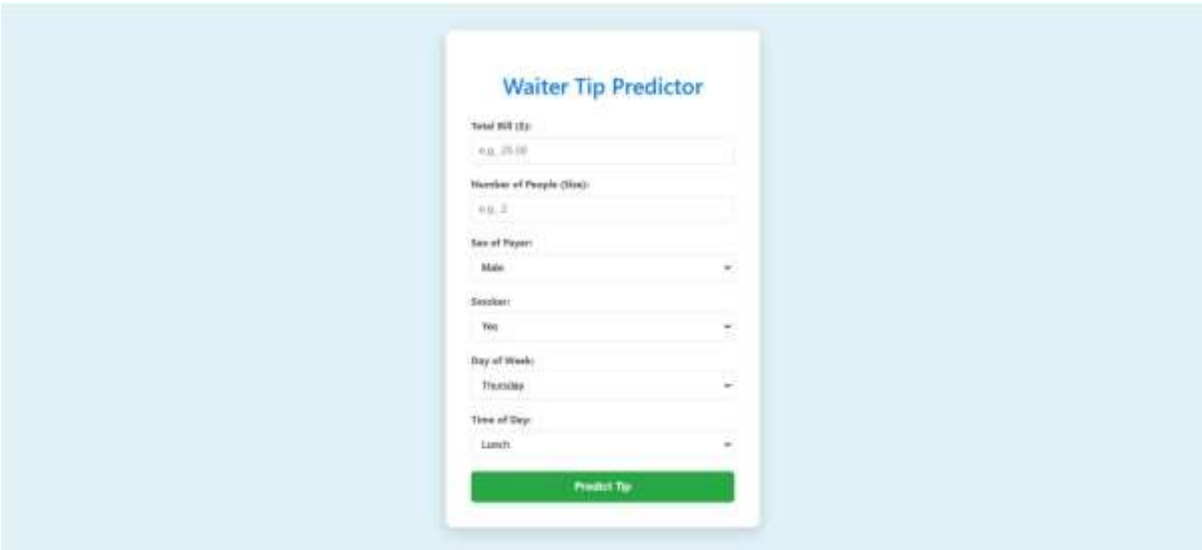

Fig:Waiter Tip Predictor

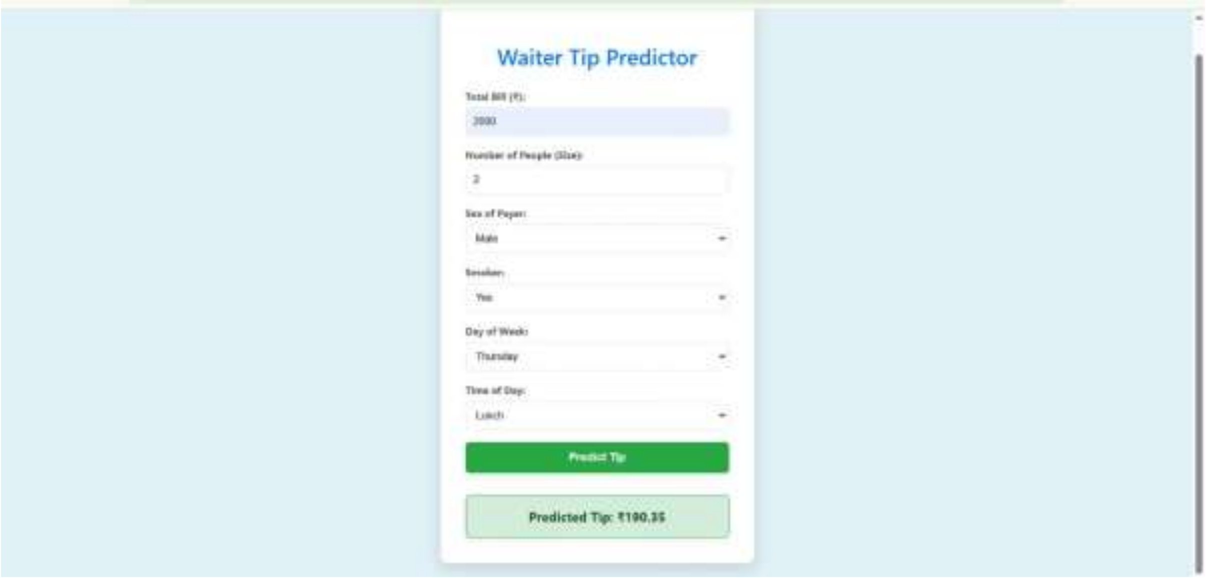

Fig: Prediction

4. RESULTS:

The project successfully implemented a Linear Regression model that accurately predicts the tip amount based on user input data. During testing and validation, the model demonstrated a strong correlation between input features (such as total bill, group size, gender, and time) and the actual tips given. The model was trained on real restaurant data and achieved good prediction accuracy with minimal error, indicating that it can reliably estimate tips in different scenarios. The Flask-based web interface worked efficiently, providing real-time predictions as users entered their data. The system also handled input validation and preprocessing correctly, ensuring the model received data in the proper format. Overall, the integration of machine learning with a web application resulted in a smooth and interactive tool that showcases how predictive analytics can be applied in the hospitality domain. The project proves that even a simple linear model can offer practical value in customer behavior prediction when implemented properly.

5.DISCUSSION:

The project discusses the usefulness of predicting customer tips to improve restaurant service and staff planning. Linear Regression was selected for its simplicity and effectiveness in handling numeric predictions. Input features like total bill, gender, smoker status, day, time, and group size were chosen based on their influence on tipping behavior. One-hot encoding was used for preprocessing categorical data. A Flask-based web app was implemented for real-time predictions, making the system user-friendly and practical. The project also considers future improvements, such as using advanced models or integrating a database. Limitations like dependence on data size and the simplicity of linear models are acknowledged**.**

6. CONCLUSION

The project successfully demonstrates how machine learning can be applied to predict the tip amount a customer might give in a restaurant setting. By using a Linear Regression model along with a simple and user-friendly Flask web interface, the system accurately forecasts tip values based on key customer and order details such as total bill, gender, smoker status, day, time, and group size. The integration of data preprocessing, model prediction, and real-time output display shows the practical potential of predictive analytics in the hospitality industry. This solution can help restaurants make smarter decisions, improve customer experience, and optimize staff performance. Overall, the project proves that even basic machine learning models, when implemented effectively, can bring meaningful insights and automation to real-world business operations.

7. FUTURE SCOPE:

The current system can be enhanced in several ways to improve its accuracy, usability, and scalability. In future versions, more advanced machine learning algorithms like Random Forest, XGBoost, or Neural Networks can be implemented to handle complex relationships in data and improve prediction performance. A larger and more diverse dataset can be used to train the model, making it more robust across different customer behaviors and restaurant types. Integration with a real-time database can enable storage of past transactions and predictions, allowing continuous learning and performance tracking. The system can also be extended to a mobile application for easier access by restaurant staff. Furthermore, additional features like customer loyalty, food category, or server ID can be added to improve prediction accuracy. Finally, the tool can be integrated into restaurant billing systems to automate tip suggestions at the point of sale, making it more practical for real-world deployment.

8. ACKNOWLEDGEMENT:

Miss. M. Tarani working as an Assistant Professor in Master of Computer Applications (MCA) in Sanketika Vidya Parishad Engineering College, Visakhapatnam, Andhra Pradesh. With 1 year experience as Automation tester in Stigentech IT services private limited, and member in IAENG, accredited by NAAC with her areas of interests in C, Java, Data Structures, Web Technologies, Python, Software Engineering.

Adari Jayanth is pursuing his final semester MCA in Sanketika Vidya Parishad Engineering College, accredited with A grade by NAAC, affiliated by Andhra University and approved by AICTE. With interest in Machine Learning, Adari Jayanth has taken up his PG project on “WAITERS TIP PREDICTION USING MACHINE LEARNING” and published the paper in connection to the project under the guidance of MAMIDI TARANI, Assistant Professor, Master of Computer

REFRENCES

[1] Restaurant tip prediction using linear regression
https://www.academia.edu/79865309/Restaurant_tip_prediction_using_linear_regression?utm_source=chatgpt.com
[2] Restaurant Tipping Linear Regression Model (SSRN)
https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3719232&utm_source=chatgpt.com
[3] Predicting waiters prompt service by analyzing restaurants rating
https://www.academia.edu/95580162/Predicting_waiters_prompt_service_by_analyzing_restaurants_rating_and_other_factors_using_machine_learning?utm_source=chatgpt.com
[4] Predicting waiter’s prompt service…
https://www.researchgate.net/publication/367613974
[5] Waiter Tips Prediction with Machine Learning
https://amanxai.com/2022/02/01/waiter-tips-prediction-with-machine-learning/?utm_source=chatgpt.com
[6] Waiter’s Tip Prediction using Machine Learning

https://www.geeksforgeeks.org/machine-learning/waiters-tip-prediction-using-machine-learning/?utm_source=chatgpt.com
[7] ML model to predict Waiter's Tip
https://iq.opengenus.org/ml-model-to-predict-waiters-tip/?utm_source=chatgpt.com
[8] Waiter Tips Analysis & Prediction Using ML
https://www.kaggle.com/code/pthakral1998/waiter-tips-analysis-prediction-using-ml?utm_source=chatgpt.com
[9] Prediction of tipping waiters for serving food
https://www.kaggle.com/code/amirmotefaker/prediction-of-tipping-waiters-for-serving-food?utm_source=chatgpt.com
[10] Waiter Tips Prediction Model
https://github.com/raghavendranhp/waiter_tip_prediction?utm_source=chatgpt.com
[11] Waiter-s-Tip-Prediction by ArtsyInd
https://github.com/ArtsyInd/Waiter-s-Tip-Prediction?utm_source=chatgpt.com
[12] iseptianto/Waiter-Tips-Prediction-with-Machine-Learning
https://github.com/iseptianto/Waiter-Tips-Prediction-with-Machine Learning?utm_source=chatgpt.com
[13] Afia1102/Predicting-Waiter-Tips
https://github.com/Afia1102/Predicting-Waiter-Tips-A-Python-Based-Machine-Learning-Approach-using-Linear-Regression?utm_source=chatgpt.com
[14] shubhamets/Waiter-Tip-Analysis-and-future-Prediction
https://github.com/shubhamets/Waiter-Tip-Analysis-and-future-Prediction?utm_source=chatgpt.com
[15] Food Delivery Time Prediction in Indian Cities Using ML Models
https://arxiv.org/abs/2503.15177?utm_source=chatgpt.com