CAPSTONE PROJECT

SENTIMENT ANALYSIS

Presented By:
Atkimsetty Mohan Sai Sankar
B.TECH ELECTRONICS AND COMMUNICATION
KL UNIVERSITY

OUTLINE

- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm & Deployment
- Result
- Conclusion
- Future Scope
- References

PROBLEM STATEMENT

In the competitive restaurant industry, understanding customer sentiment through reviews is crucial for maintaining high service standards and customer satisfaction. Manually analyzing numerous reviews is both time-consuming and inefficient. Therefore, the objective is to develop a sentiment analysis model that can automatically classify restaurant reviews as either positive or negative. This model will leverage machine learning techniques to preprocess and analyze textual data, transforming raw reviews into actionable insights. The process involves collecting a dataset of restaurant reviews, cleaning and preprocessing the text, and then training a machine learning model. The model's performance will be evaluated using metrics like accuracy, precision, recall, and F1-score. Once validated, this model will be deployed to classify new reviews in real-time, enabling restaurant management to swiftly and accurately gauge customer sentiment and make informed decisions to enhance the dining experience.

PROPOSED SOLUTION

- **The proposed system aims to develop a sentiment analysis model to classify restaurant reviews as positive or negative. This involves leveraging data analytics and machine learning techniques to accurately gauge customer sentiment. The solution consists of the following components:**
- Data Collection:
  - Gather a dataset of restaurant reviews, including review text and sentiment labels (positive or negative).
  - Collect additional data such as review dates and ratings if available.
- Data Preprocessing:
  - Clean the text data to remove missing values, punctuation, numbers, and convert text to lowercase.
  - Tokenize the text and remove stopwords.
  - Apply text vectorization techniques like TF-IDF or word embeddings to convert text into numerical features.
- Machine Learning Algorithm:
  - Implement and train machine learning models such as Logistic Regression, Naive Bayes, SVM, or neural networks (LSTM, BERT) on the preprocessed data.
  - Perform hyperparameter tuning to optimize model performance.
- Deployment:
  - Develop a user-friendly interface for real-time sentiment classification of new reviews.
  - Deploy the model on a scalable platform to handle user requests efficiently.
- Evaluation:
  - Assess the model's performance using metrics like accuracy, precision, recall, and F1-score.
  - Continuously monitor and fine-tune the model based on feedback and performance metrics.
- Result:
  - Provide accurate sentiment classification to help restaurant management understand customer feedback.
  - Enable better decision-making to enhance service quality and customer satisfaction.

SYSTEM APPROACH

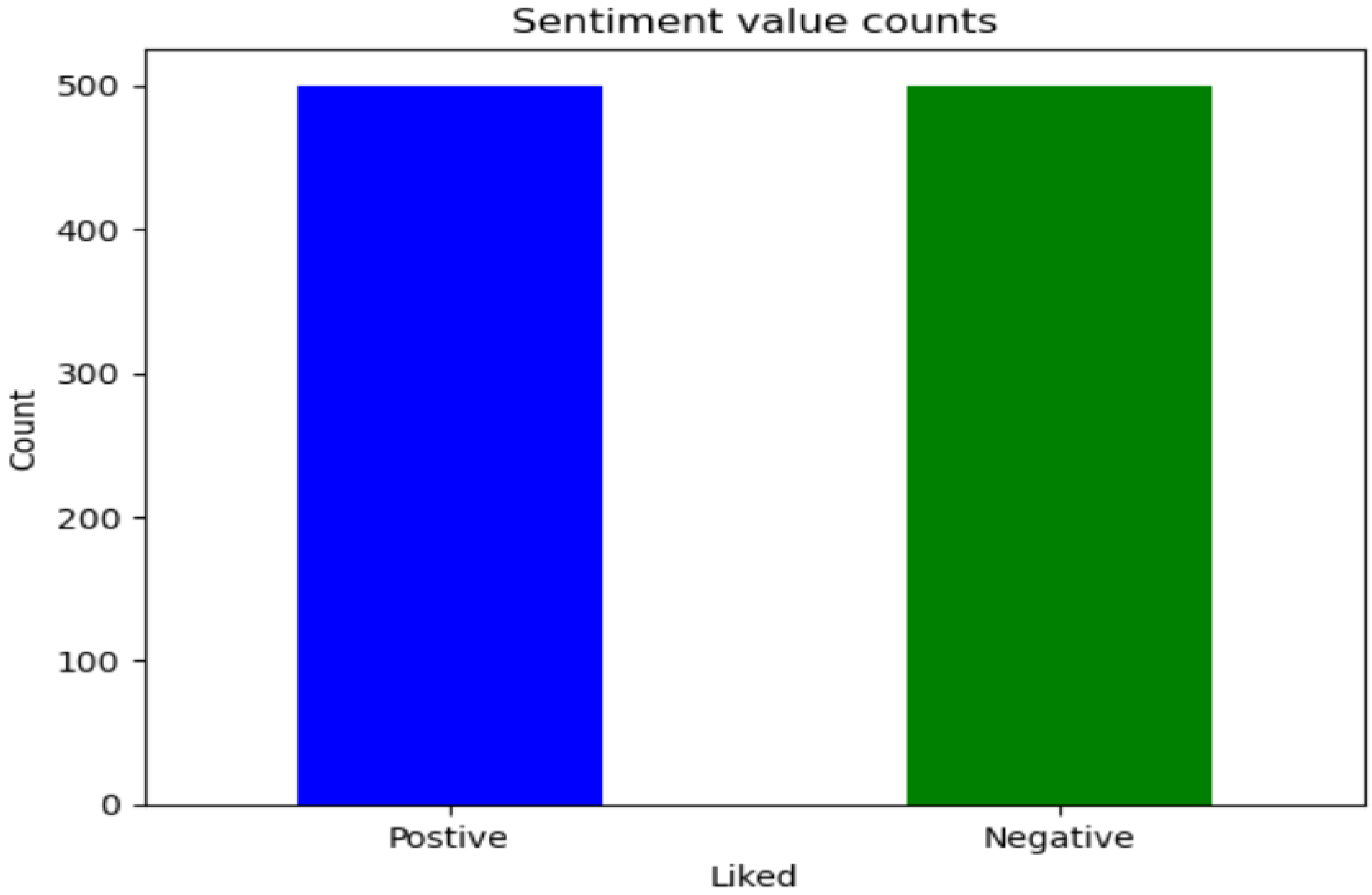
- The "System Approach" section outlines the overall strategy and methodology for developing and implementing the sentiment analysis model for restaurant reviews. Here is the structure for this section:
- **System Requirements:**
  - **Hardware:** A modern computer with at least 8GB RAM and a multi-core processor. Access to a GPU is beneficial for training complex models such as neural networks.
  - **Software:** Python environment with relevant libraries installed.
  - **Data:** A labeled dataset of restaurant reviews with sentiment labels (positive or negative).
  - **Storage:** Adequate storage to handle data and model files.

ALGORITHM & DEPLOYMENT

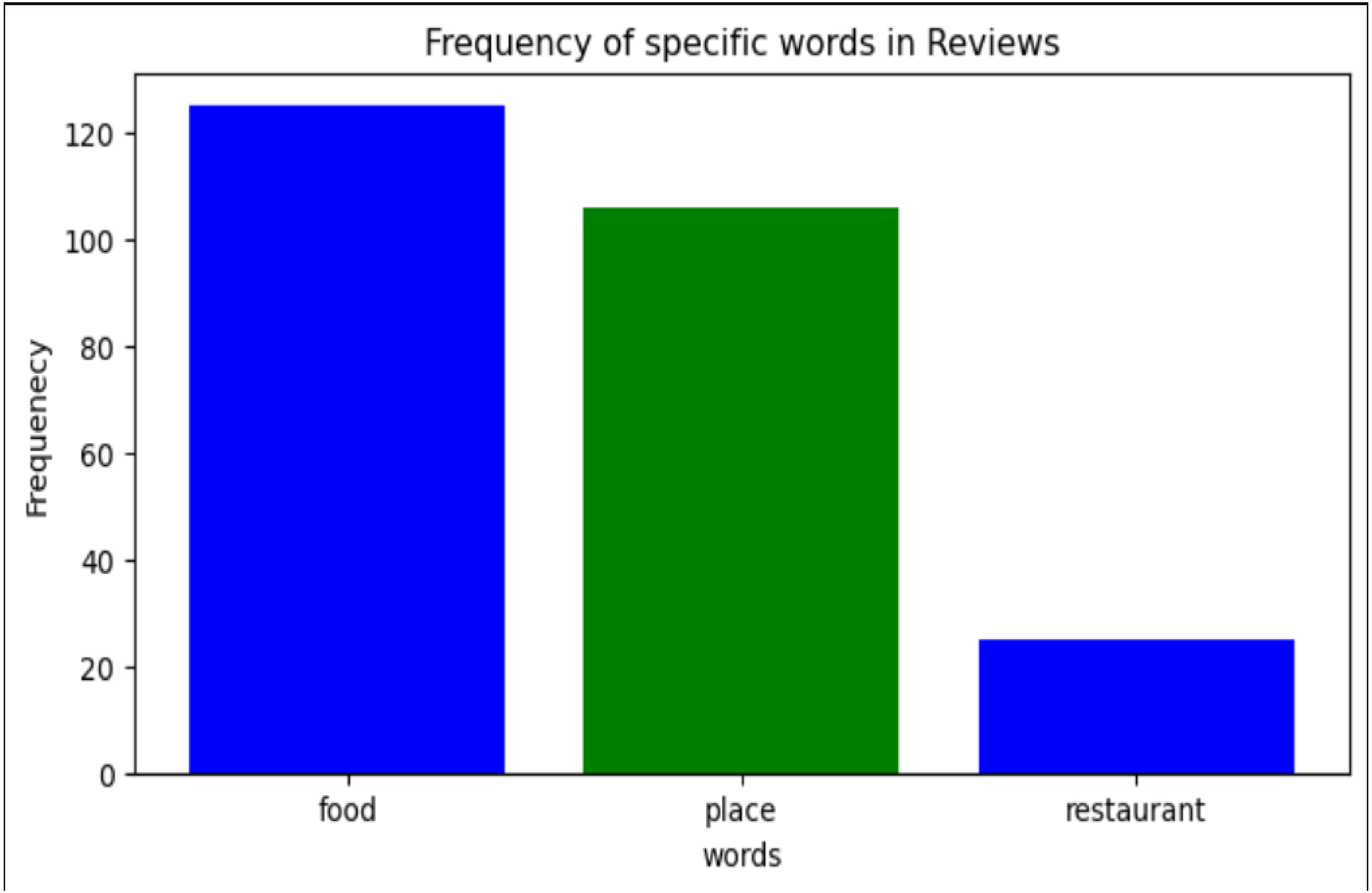
- Algorithm Selection:
  - For simplicity and efficiency in binary classification tasks like sentiment analysis, NLTK's Naive Bayes classifier is chosen. Naive Bayes classifiers are known for their ease of implementation and computational efficiency.
- Data Input:
  - The algorithm takes preprocessed text data as input, with each review cleaned, tokenized, and converted into a bag-of-words representation using NLTK's tokenization and feature extraction utilities.
- Training Process:
  - **Data Splitting**: The dataset is divided into training and testing sets, typically using an 80-20 split.
- **Model Training**: The Naive Bayes classifier is trained using the bag-of-words features extracted from the training data.
  - **Cross-Validation**: K-fold cross-validation ensures the model's performance generalizes well to unseen data, with NLTK providing convenient utilities for this purpose.
  - **Hyperparameter Tuning**: While Naive Bayes classifiers have minimal hyperparameters, techniques like Laplace smoothing may be employed to improve performance.
- Prediction Process:
  - **Data Preprocessing**: New reviews undergo the same preprocessing steps as the training data, ensuring consistency.
  - **Making Predictions**: The trained Naive Bayes classifier predicts the sentiment of new reviews based on the learned probabilities from the training data.
  - **Real-Time Inputs**: In real-time scenarios, the model can continuously process and analyze new reviews using NLTK's efficient text processing capabilities, enabling timely sentiment analysis for restaurant management.

RESULT

The sentiment analysis model for restaurant reviews was evaluated using a test dataset, and its performance was assessed in terms of accuracy and effectiveness. Here are the results along with visualizations and comparisons between predicted and actual sentiments to highlight the model's performance.


Sentiment value counts
500
400
300
200
100
0
Count
Postive
Negative
Liked

edunet
foundation



edunet
foundation

```
[83]: model = MultinomialNB()
      model.fit(X_train, Y_train)
```

```
[83]: ▾ MultinomialNB
      MultinomialNB()
```

```
[84]: y_pred = model.predict(X_test)
```

```
[85]: accuracy= accuracy_score(Y_test, y_pred)
```

```
[86]: report= classification_report(Y_test, y_pred)
```

```
[87]: print(f'Accuracy {accuracy}')
```

```
Accuracy 0.8
```

```
[90]: print(f'Classification report:')
      print(report)
```

```
Classification report:
              precision    recall  f1-score   support

           0       0.76      0.85      0.80        96
           1       0.85      0.75      0.80       104

    accuracy                           0.80       200
   macro avg       0.80      0.80      0.80       200
weighted avg       0.81      0.80      0.80       200
```

```
#new review prediction)
new_review = input("Enter a review")
cleaned_review = preprocess_review(new_review)
new_review_vectorized = vectorizer.transform([cleaned_review])
prediction = model.predict(new_review_vectorized)
if prediction[0] == 1:
    print("The review is predicted postive")
else:
    print("The review is predicted negative")
```

```
Enter a review I hate this product
The review is predicted negative
```

CONCLUSION

- Overall, the sentiment analysis model demonstrates promising performance in classifying restaurant reviews, with a high accuracy rate and a close alignment between predicted and actual sentiments. While the model provides valuable insights into customer feedback, continued refinement and optimization may enhance its performance further, enabling restaurant management to make data-driven decisions to improve service quality and customer satisfaction.

FUTURE SCOPE

- In the realm of restaurant review sentiment analysis, there are numerous avenues for future enhancements. Incorporating additional data sources like social media platforms could provide real-time insights into customer sentiment. Algorithm optimization using advanced NLP techniques could improve classification accuracy. Expanding the system to cover multiple cities or regions would require customized models to accommodate variations in cuisine preferences and cultural nuances. Integration of emerging technologies such as edge computing and AI-driven chatbots could revolutionize real-time feedback analysis and customer engagement. Continuous improvement through feedback mechanisms and iterative development is essential for refining the system based on evolving business needs and technological advancements.

REFERENCES

•Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1-2), 1-135.

•Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies* (pp. 142-150).

•Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

•Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc.

THANK YOU