**Text Analysis (Extra Model)**

**1. Data Loading and Preprocessing**

The code starts by loading Yelp's dataset files containing user tips and check-ins.

Text Processing:

A custom function text\_process is defined to preprocess the text data:

Punctuation is removed, and text is tokenized.

Stopwords (common words that add little meaning) are filtered out to enhance the quality of the text data used for modeling.

**2. Data Transformation**

The compliment\_count feature is categorized into star ratings using the pd.cut function.

The text data is processed through the text\_process function, and the processed text is stored in a new column called processed\_text.

**3. Text Analysis**

A frequency count of words is generated using the Counter from the collections module, identifying the most common words in the tips.

A bar plot is created to visually represent the frequency of these words, which can provide insights into what users commonly mention in their reviews.

**4. Model Training and Evaluation**

The data is split into training and testing sets using train\_test\_split, with 80% for training and 20% for testing.

Vectorization: The text data is converted into numerical vectors using CountVectorizer.

A Complement Naive Bayes model is then trained on the vectorized text data.

Predictions are made on the test set, and model performance is evaluated using accuracy and a classification report, which includes precision, recall, and F1-score for each class (star rating).

**5. Performance Metrics**

The accuracy of the model is printed along with a detailed classification report, providing insights into how well the model performed on each class.

6. Additional Data Analysis

A histogram of the lengths of the business\_id column is generated, giving insights into the distribution of business ID lengths.

Similar frequency analysis is performed on the business\_id field, with results visualized in a bar plot.

**Summary Report**

Text Preprocessing

Methods Used:

Removal of punctuation and stopwords.

Tokenization to prepare text for analysis.

Impact: Improved model performance by reducing noise in the text data.

Model Performance

Model: Complement Naive Bayes

Accuracy: The model achieved an accuracy of approximately X%X \%X% (replace with actual accuracy from your results).

Classification Report:

Provides metrics for each star rating category, revealing which categories were predicted well and which were confused.

Insights from Analysis

The most common words in reviews may highlight user sentiments and focus areas for businesses.

Understanding the distribution of star ratings can guide businesses on areas needing improvement based on customer feedback.

**Conclusion**

This analysis effectively showcases how sentiment analysis can be performed on Yelp data using Naive Bayes classification. The results can be invaluable for businesses aiming to improve customer satisfaction by understanding feedback trends. The combination of text preprocessing, model evaluation, and visualization provides a robust framework for text classification tasks.