Phase3 Logbook

Tarfah Al Ateeq	443200800
Luluh Alyahya	443200609
Norah Aljedai	443200841
Layan Alsaykhan	443200751
Deema Alkanhal	443202970

1. The role of language in tourist satisfaction: analyzing Google Maps reviews from international with local visitor

1. Objective

Analyze how the language of evaluations (Arabic or English) affects sentiment and content. Compare sentiments and key subject matters in evaluations to perceive variations in expectancies or pleasure tiers among nearby and global tourists. Offer insights into how language impacts traveler reviews and perceptions of destinations.

2. Methods and Workflow

1. Data Loading:

Imported a dataset (ProcessedData.csv) containing traveler reviews for evaluation.

2. Language Detection:

Used the **langdetect** library to perceive the language (Arabic or English) for every evaluate. Added a language column to keep detected languages.

Sentiment Analysis:

Utilized pre-skilled **BERT** fashions for each Arabic and English evaluations:

- Arabic Model: asafaya/bert-base-arabic
- English Model: nlptown/bert-base-multilingual-uncased-sentiment

Created sentiment evaluation pipelines with the use of Hugging Face`s Transformers library. A sentiment column was added to keep track of the outcomes of the sentiment evaluation.

3. Iterative Processing:

For each review in the dataset:

- Detected the language.
- Applied the respective sentiment evaluation version primarily based totally at the detected language.

- **Key Insights:**

Enabled evaluation of sentiment styles among nearby and global tourists.

3. Challenges Encountered

- Language Detection Failures:
  - Some evaluations failed language detection and had been labeled as "unknown.
  - " Mitigation: Implemented a default fallback for undetectable languages.

- Model Compatibility:

  Ensured accurate tokenizer and version pairing for every language to keep away from errors.

- Performance Bottlenecks:

  Processing the dataset iteratively the use of iterrows() caused capacity inefficiencies with large datasets.

4. **Outcomes**

The notebook effectively integrates multilingual sentiment evaluation pipelines. Data is now enriched with language and sentiment labels for similarly thematic and comparative evaluation.

5. **Future Enhancements**
   - Optimize sentiment evaluation the use of batch processing to address large datasets efficiently.
   - Explore opportunity language detection techniques for stepped forward accuracy.

2. Comparative Analysis of Tourist Reviews in Saudi Arabia

We performed a detailed analysis and comparison of **Cultural Heritage** and **Modern Attractions** tourist reviews in Saudi Arabia, identifying trends, satisfaction levels, and actionable insights.

Steps Taken and Rationale

Data Cleaning and Preprocessing:

Steps:

Loaded and explored the dataset for structure and content.
Identified and addressed missing values in key columns like text and city.
Removed invalid or redundant columns, including columns with excessive NaN values.
Rationale: Ensuring clean and structured data is critical for reliable analysis.

Challenge:

Missing or incomplete data in the city column.
Solution:

Dropped rows with NaN in critical columns like city or replaced missing text with empty strings for sentiment analysis.

Categorization of Reviews:

Steps:

Categorized reviews into two groups:
Cultural Heritage: Mosques, historical landmarks, and cultural sites.
Modern Attractions: Theme parks, malls, and entertainment facilities.
Used keywords and contextual information to classify reviews accurately.
Rationale: Segregating reviews allowed for a focused comparison between the two types of attractions.

Sentiment Analysis:

Steps:

Mapped sentiment labels (POSITIVE, NEGATIVE, NEUTRAL) to numeric scores (1, -1, 0) for analysis.
Applied BERT-based sentiment analysis for robust sentiment classification.
Calculated average sentiment scores for each attraction category and visualized results using bar charts.

Rationale: Numeric sentiment scores provided quantifiable insights into tourist satisfaction levels.

Challenge:

Sentiment analysis errors due to non-string values in the text column.
Solution:

Converted all text to string format and handled missing or invalid entries gracefully.

Negative Review Analysis:

Steps:

Filtered reviews labeled as NEGATIVE.
Analyzed themes in negative feedback for Modern Attractions.
Identified recurring complaints like:
Overpricing.
Facility or service-related dissatisfaction.
Rationale: Understanding negative feedback is essential for identifying actionable improvement areas.

City-Based Sentiment Analysis:

Steps:

Grouped reviews by city and calculated average sentiment scores for each category.
Merged all cities into the analysis, ensuring cities without reviews in a category were still included.
Visualized geographic trends using bar charts.
Rationale: Geographic trends help identify cities that excel or need improvement in tourism experiences.

Challenge:

Some cities lacked reviews in one or both categories.
Solution:

Included all cities in the analysis, assigning NaN for missing sentiment scores to maintain comprehensiveness.

Sentiment Distribution Analysis:

Steps:

Visualized sentiment distribution for both categories using histograms and bar charts.
Highlighted differences in variability and consistency between the two categories.
Rationale: Understanding sentiment variability provides insights into the consistency of tourist experiences.

Summary and Recommendations

Steps:

Summarized findings from sentiment analysis, negative feedback, and city-based trends.
Drafted actionable recommendations to improve Modern Attractions and leverage the strengths of Cultural Heritage.
Rationale: Aligning findings with actionable insights is crucial for guiding future tourism strategies.

Challenges Faced and Solutions

Data Quality:

Issue:

Missing or incomplete data in key columns (text, city).
Solution:

Replaced missing values or dropped rows as necessary, ensuring minimal data loss.
Sentiment Analysis Errors:

Issue:

Errors arose when processing non-string values in the text column.
Solution:

Converted all text to string format and handled missing entries by replacing them with empty strings.
City Representation:

Issue:

Some cities lacked reviews in one or both categories.
Solution:

Ensured all cities were included in the analysis, assigning NaN for missing data.
Broad Sentiment Range in Modern Attractions:

Issue:

High variability in Modern Attractions sentiment made trends harder to interpret.
Solution:

Used visualizations (e.g., box plots, histograms) to clarify trends and extremes.
Decisions Made

Categorization:

Decided to classify reviews based on keywords and categories provided in the dataset.
Handling Missing Data:

Chose to drop or replace missing values based on their relevance to the analysis.
Sentiment Analysis Tool:

Used BERT for sentiment classification due to its ability to handle complex linguistic structures.
Visualization:

Opted for bar charts and histograms to effectively communicate trends and insights.

Outcome

The analysis successfully:

- Highlighted higher and more consistent satisfaction for **Cultural Heritage** over **Modern Attractions**.
- Identified overpricing, accessibility issues, and value-for-money concerns as key areas for improvement in Modern Attractions.
- Showcased strong performance in cities like **Jeddah** and **Riyadh**, while identifying areas like **Neom** for further development.
- Provided actionable recommendations to enhance tourist experiences and leverage Saudi Arabia’s cultural assets

3. Impact of Tourist Volume on Service Quality Perceptions in Saudi Arabia's Tourist Sites

In this phase, the goal was to predict the **service quality** perceptions of tourist sites based on **tourist volume** using various machine learning models. The models were compared and evaluated using multiple performance metrics, such as **Mean Squared Error (MSE)** and **R-squared**. The following steps were taken, along with the reasoning and decisions behind each.

- Data Preparation

Objective: Prepare the data for model training and testing.

1. **Data Selection**:
    a. We decided to use **totalscore** (or stars) as the **target variable** because they directly represent **service quality perceptions** for tourist sites. These scores were chosen because they are the most reliable measures for the service quality that we are trying to predict.
    b. The **features** (independent variables) were selected to represent **tourist volume** and **location**. We included **city-related features** (city_Riyadh, city_Jeddah, city_Dhahran) and **category-related features** (categories_Animal park, categories_Resort hotel, categories_Tourist attraction, etc.). These features were selected because:
        i. They capture **tourist volume** (city and category-based tourist flows) which directly influences the service quality at tourist sites.
        ii. Tourist volume in certain cities and categories has been shown to impact service quality, as more tourists could lead to higher demand, potentially affecting the quality of services provided.
2. **Data Preprocessing**:
    a. **Imputation**: Missing values in both the features and target variable were handled using **mean imputation** (SimpleImputer(strategy='mean')). This approach was chosen because the dataset had missing values in some columns, and imputation with the mean value is a common and safe method to handle missing numerical data.
    b. **Train-Test Split**: The data was split into **80% training** and **20% testing**. This split ratio was chosen to ensure enough data for training while also maintaining a separate test set to evaluate the model's generalization ability.

Challenges:

- **Missing Values**: A challenge we faced was missing values in the dataset, particularly in some categorical columns. This was resolved by using imputation.
- **Feature Engineering**: Some columns may have needed further transformation (e.g., one-hot encoding for categorical variables), but for simplicity and given the nature of the data, we used the columns directly.

- Model Selection and Training

Objective: Train three models to predict the service quality: **Linear Regression**, **Decision Tree**, and **Random Forest**.

1. **Model Choice**:
    a. **Linear Regression**: This model was chosen as the baseline due to its simplicity and ability to predict continuous values. It was a natural first choice to assess the linear relationship between tourist volume and service quality.
    b. **Decision Tree**: A **Decision Tree** was chosen because it can handle non-linear relationships in the data. It's also easy to interpret and can be useful when there are categorical variables.
    c. **Random Forest**: The **Random Forest** model was chosen due to its ability to aggregate predictions from multiple decision trees, which improves performance and reduces overfitting. It's typically more robust and provides better predictions than a single decision tree.
2. **Model Training**:
    a. All three models were trained using the same training data (X_train and y_train). The models were then evaluated on the test data (X_test).
    b. **Random Forest** performed well out of the box due to its ensemble nature, while **Linear Regression** and **Decision Tree** were used for comparison.

Challenges:

- **Overfitting with Decision Tree**: Decision Trees tend to **overfit** when the data is noisy or too complex. We observed this behavior during evaluation, where the model's performance on the training set was much better than on the test set.
- **Model Complexity**: Deciding the right complexity for the **Decision Tree** (like the depth of the tree) was challenging, but we kept it simple initially to avoid overfitting.

- Model Evaluation and Comparison

Objective: Evaluate and compare the models based on their performance metrics.

1. **Metrics**:
    a. **R-squared (R^2)** was used to measure how well the model explains the variance in service quality.
    b. **Mean Squared Error (MSE)** was used to evaluate the accuracy of the predictions. A lower MSE indicates better model performance.
2. **Results**:
    a. **Linear Regression** had a relatively low R^2 value and higher MSE compared to the other models, indicating that it wasn't the best fit for this data.
    b. **Decision Tree** showed better performance but exhibited signs of overfitting. Its performance on the test set wasn't as good as on the training set.
    c. **Random Forest** had the best performance, with a high R^2 and low MSE, showing that it could handle the complexity of the data and generalize well to the test set.
3. **Comparison**:
    a. The **Random Forest** model outperformed both **Linear Regression** and **Decision Tree** in terms of both **R^2** and **MSE**, making it the model of choice for predicting service quality.
    b. Visualizing the **Actual vs Predicted** scatter plots helped in understanding that **Random Forest** provided predictions closer to the actual values, while the other models (especially **Linear Regression**) showed more spread and errors in predictions.

Challenges:

- **Model Interpretation**: While Random Forest outperformed the other models, it is an ensemble model and not as interpretable as a **Decision Tree** or **Linear Regression**. This is a trade-off for better performance.
- **Tuning the Decision Tree**: Finding the right depth and complexity of the Decision Tree to avoid overfitting was a bit tricky. This could be improved by using cross-validation or hyperparameter tuning.

- Conclusion and Insights

1. **Best Performing Model**:
    a. Based on the **R^2** and **MSE**, the **Random Forest** model was the best performing model, with the most accurate predictions and the least error.
2. **Feature Importance**:
    a. The **Random Forest** model also provided insights into which features (e.g., city or category-related features) were most important for predicting service quality, which could inform further business decisions or policy recommendations for tourist sites.

Challenges:

- **Understanding the Impact of Features**: While **Random Forest** gave feature importance, interpreting the results was not as straightforward as with **Linear Regression** or **Decision Tree**, where the relationships between features and target are easier to understand.

4. Predicting Spending Based on Length of Stay and Key Attributes

1. Objective

To develop and evaluate machine learning models for predicting tourist spending using features such as length of stay, purpose of visit, and other attributes.

2. Steps and Details

Step 1: Data Loading and Preprocessing

- Loaded the dataset containing tourist spending data.
- Performed data cleaning:
    - Handled missing values.
    - Ensured data types were consistent for analysis.
- Conducted exploratory data analysis (EDA) to understand the dataset's distribution, correlations, and key features.

Step 2: Feature Selection and Engineering

- Selected relevant features based on correlation analysis and domain knowledge.
- Normalized/standardized numerical features to improve model performance.
- Encoded categorical features using one-hot encoding.

Step 3: Splitting Data

- Split the data into training and testing sets (e.g., 80% training, 20% testing) to evaluate model performance.

Step 4: Model Development

- Implemented three machine learning models:
    - **Linear Regression**: A simple baseline model to understand linear relationships.
    - **Random Forest**: An ensemble model that reduces overfitting and improves accuracy.
    - **Gradient Boosting**: A robust iterative model designed to handle complex relationships effectively.

Step 5: Model Evaluation

- Evaluated each model using:
  - **Mean Absolute Error (MAE):** To measure average prediction error.
  - **Root Mean Squared Error (RMSE):** To penalize larger prediction errors.
- Gradient Boosting demonstrated the lowest MAE and RMSE, making it the most accurate model.

Step 6: Final Model Selection

- Selected **Gradient Boosting** as the final model due to:
  - Best performance (lowest MAE and RMSE).
  - Ability to capture non-linear relationships.
  - Robustness and adaptability to diverse datasets.

Step 7: Saving the Final Model

- Saved the trained Gradient Boosting model as `gradient_boosting_model.pkl` for deployment or future use.

3. Key Findings

- Gradient Boosting consistently outperformed other models in accuracy, robustness, and generalization.
- Random Forest provided reliable results but was slightly less accurate than Gradient Boosting.
- Linear Regression struggled with the dataset due to its inability to capture complex relationships.

4. Future Improvements

- Experiment with hyperparameter tuning for further optimization of Gradient Boosting.
- Explore other advanced ensemble models, such as XGBoost or LightGBM, for comparison.
- Consider additional feature engineering to capture more relevant patterns in the data.

5. Conclusion

Gradient Boosting was chosen as the best model due to its exceptional performance in both MAE and RMSE metrics, its ability to model non-linear relationships, and its robustness across datasets. The saved model is ready for deployment or integration into broader analytics workflows.