Phase 2 Document: Data Wrangling and Analysis

Introduction

Phase 2 of our project is dedicated to data wrangling and analysis, critical steps in preparing the raw dataset for building a personalized content discovery engine. This phase involves employing various data manipulation techniques using Python to clean, transform, and explore the dataset. Additionally, we assume a scenario where the project aims to recommend personalized content to users based on their preferences and interactions, enhancing user engagement and satisfaction.

Objectives:

1. Cleanse the dataset by addressing inconsistencies, errors, and missing values to ensure data integrity.
2. Explore the dataset's characteristics through exploratory data analysis (EDA) to understand distributions and correlations.
3. Engineer relevant features to enhance model performance for accurate content recommendations.
4. Document the data wrangling process comprehensively, ensuring transparency and reproducibility.

Dataset Description

The dataset comprises user interaction data collected from a digital platform, including information about user profiles, content items, and user interactions such as ratings, views, and purchases. Each row in the dataset represents a user's interaction with a specific content item, forming the foundation for personalized content recommendations.

Data Wrangling Techniques

1. Data Description

- **Head :** Displaying the first few rows of the dataset to get an initial overview.
- **Tail :** Examining the last few rows of the dataset to ensure completeness.
- **Info :** Obtaining information about the dataset structure, data types, and memory usage.
- **Describe :** Generating descriptive statistics for numerical features to understand their distributions and central tendencies.

Code:

```python
 Sample code for data description
print(data.head())
print(data.tail())
print(data.info())
print(data.describe())
```

Output Screenshot

2. Null Data Handling

- **Null Data Identification** : Identifying missing values in the dataset.
- **Null Data Imputation :** Filling missing values with appropriate strategies.
- **Null Data Removal :** Eliminating rows or columns with excessive missing values.

Code:

```python
 Sample code for null data handling
print(data.isnull().sum())
data = data.dropna()  Drop rows with missing values
```

Output Screenshot

3. Data Validation

- **Data Integrity Check :** Verifying data consistency and integrity to eliminate errors.
- **Data Consistency Verification** : Ensuring data consistency across different columns or datasets.

Code:

```python
 Sample code for data validation
 Check for unique values in a column
print(data['column_name'].unique())
```

Output Screenshot

4. Data Reshaping

- **Reshaping Rows and Columns** : Transforming the dataset into a suitable format for analysis.
- **Transposing Data** : Converting rows into columns and vice versa as needed.

Code:

```python
 Sample code for data reshaping
 Transpose the dataset
transposed_data = data.T
```

Output Screenshot

5. Data Merging

- **Combining Datasets :** Merging multiple datasets or data sources to enrich the information available for analysis.
- **Joining Data :** Joining datasets based on common columns or keys.

Code:

```python
 Sample code for data merging
merged_data = pd.merge(data1, data2, on='common_column')
```

Output Screenshot

6. Data Aggregation

- **Grouping Data :** Grouping dataset rows based on specific criteria.
- **Aggregating Data :** Computing summary statistics for grouped data.

Code:

```python
 Sample code for data aggregation
grouped_data = data.groupby('category_column')
aggregated_data = grouped_data.agg({'numerical_column': 'mean'})
```

Output Screenshot

Data Analysis Techniques

7. Exploratory Data Analysis (EDA)

- **Univariate Analysis :** Analyzing individual variables to understand their distributions and characteristics.
- **Bivariate Analysis :** Investigating relationships between pairs of variables to identify correlations and dependencies.

- **Multivariate Analysis** : Exploring interactions among multiple variables to uncover complex patterns and trends.

Code:

```python
 Sample code for exploratory data analysis
import seaborn as sns
import matplotlib.pyplot as plt

 Univariate analysis - Histogram
sns.histplot(data['numerical_column'], bins=20)
plt.show()

 Bivariate analysis - Scatter plot
sns.scatterplot(data['feature1'], data['feature2'])
plt.show()

 Multivariate analysis - Pair plot
sns.pairplot(data)
plt.show()
```

Output Screenshot

8. Feature Engineering

- **Creating User Profiles :** Aggregating user interaction data to construct comprehensive user profiles capturing preferences and behaviors.

- **Temporal Analysis :** Incorporating temporal features such as time of day or day of week to capture temporal trends in user behavior.

- **Content Embeddings :** Generating embeddings for content items to represent their characteristics and relationships.

Code:

```python
 Sample code for feature engineering
 Creating user profiles
user_profiles = data.groupby('user_id').agg({'interaction_column': 'mean'})

 Temporal analysis
data['timestamp'] = pd.to_datetime(data['timestamp'])
data['hour_of_day'] = data['timestamp'].dt.hour

 Content embeddings
 Code for generating embeddings using techniques like word2vec or doc2vec
```

Output Screenshot

Assumed Scenario

- **Scenario :** The project aims to recommend personalized content to users based on their historical interactions and preferences.

- **Objective :** Enhance user engagement and satisfaction by delivering relevant and tailored content recommendations.

- **Target Audience :** Digital platform users seeking personalized content recommendations across various domains.

Conclusion

Phase 2 of the project focuses on data wrangling and analysis to prepare the dataset for building a personalized content discovery engine. By employing Python-based data manipulation techniques and assuming a scenario focused on personalized content recommendations, we aim to transform raw data into actionable insights for enhancing user experience and engagement on digital platforms.

(sample_code)

[Output Screenshot]