**AIT 526-DL1 Natural Language Processing**

Team 4:

Meghna Kattekola (G01463794)

Ramya Sri Thallapally (G01454183)

Srija Anasuri (G01446546)

**Professor Lindi Liao, PhD**

**George Mason University**

**04-24-2024**

**Sentiment Analysis and Recommendations for Hotels**

**Project Description:**

This project “Sentiment Analysis and Recommendations for Hotels” extracts insights with sentiment analysis that is analyzing guest reviews across multiple languages using aspect-based sentiment analysis. Giving personalized hotel recommendations i.e., tailoring hotel suggestions based on user preferences and travel needs.

This project utilizes advanced technologies, leverages NLTK, Natural Language Processing, and Machine Learning algorithms to improve analysis and recommendation accuracy. It also supports informed decision-making such as assisting travelers in making quick, informed accommodation choices. Finally helping the hoteliers enhance guest satisfaction and operational success.

The project aims to streamline the hotel selection process globally, improving both guest satisfaction and hotel management efficiency, leading to enhanced travel experiences.

**Outline of solution:**

Aspect-Based Sentiment Analysis:

Extracted sentiments for various aspects from guest reviews.

Provided insights into the hotel's strengths and weaknesses.

Recommendation System:

Utilized sentiment analysis results and user input for personalized recommendations.

Recommended hotels aligned with user preferences and sentiment analysis outcomes.

While giving results we have provided an example of how our works.

Multilingual Support:

Delivered personalized hotel suggestions based on language-specific inputs.

We have demonstrated this using example in multiples languages of English, Spanish, French

with same description to show that same output of hotel recommendations

**Examples of input/output:**

**Aspect-based Sentimental Analysis:**

* Location: Mixed sentiments but generally positive.
* Staff: Mixed sentiments, with potential service issues.
* Cleanliness: More negative sentiment, indicating areas for improvement.
* Amenities: Dissatisfaction with amenities requiring attention.
* Service: Widespread dissatisfaction, highlighting the need for improvement.
* Actionable Insights: Focus on staff training, service quality, cleanliness, and amenities to enhance guest satisfaction.

Actionable Insights: Focus on staff training, service quality, cleanliness, and amenities to enhance guest satisfaction.

Displaying aspect-based sentiment analysis results for positive reviews:

Sentiment Analysis Results for Negative Reviews:

Aspect: location

Positive: 4170, Negative: 3875, Neutral: 2671

Aspect: staff

Positive: 11871, Negative: 17881, Neutral: 4341

Aspect: cleanliness

Positive: 461, Negative: 670, Neutral: 353

Aspect: amenities

Positive: 338, Negative: 458, Neutral: 154

Aspect: service

Positive: 5000, Negative: 9072, Neutral: 3009

Displaying aspect-based sentiment analysis results for negative reviews:

Sentiment Analysis Results for Positive Reviews:

Aspect: location

Positive: 156680, Negative: 1030, Neutral: 31230

Aspect: staff

Positive: 177507, Negative: 1230, Neutral: 8558

Aspect: cleanliness

Positive: 3548, Negative: 43, Neutral: 1630

Aspect: amenities

Positive: 2486, Negative: 20, Neutral: 192

Aspect: service

Positive: 22258, Negative: 392, Neutral: 1995

**Recommendation System:**

Based on this recommendation system, we provided an example for the United Kingdom, France, and the Netherlands, which we demonstrated by entering the location, description, and tags. We have the findings of the top hotels in Amsterdam, which include the hotel name, average rating score, and hotel address.

#Example1

location = "UK"

description = "I am going for fun"

tags = "Family"

recommended\_hotels = recommend\_hotel(location, description, tags)

print(recommended\_hotels)

Output:

Hotel\_Name Average\_Score \

0 Haymarket Hotel 9.6

1 41 9.6

2 Milestone Hotel Kensington 9.5

3 Charlotte Street Hotel 9.5

4 The Soho Hotel 9.5

Hotel\_Address

0 1 Suffolk Place Westminster Borough London SW1...

1 41 Buckingham Palace Road Westminster Borough ...

2 1 Kensington Court Kensington and Chelsea Lond...

3 15 17 Charlotte Street Hotel Westminster Borou...

4 4 Richmond Mews Westminster Borough London W1D...

#Example2

location = "France"

description = "I am going on a Business Trip"

tags = "Suite"

recommended\_hotels = recommend\_hotel(location, description, tags)

print(recommended\_hotels)

Output:

Hotel\_Name Average\_Score \

0 Ritz Paris 9.8

1 H tel de La Tamise Esprit de France 9.6

2 Hotel The Peninsula Paris 9.5

3 Le Narcisse Blanc Spa 9.5

4 Nolinski Paris 9.4

Hotel\_Address

0 15 Place Vend me 1st arr 75001 Paris France

1 4 rue d Alger 1st arr 75001 Paris France

2 19 avenue Kleber 16th arr 75116 Paris France

3 19 Boulevard De La Tour Maubourg 7th arr 75007...

4 16 Avenue de l Opera 1st arr 75001 Paris France

#Example3

location = "netherlands"

description = "I am going for fun"

tags = "Solo traveler"

recommended\_hotels = recommend\_hotel(location, description, tags)

print(recommended\_hotels)

Output:

Hotel\_Name Average\_Score \

0 Waldorf Astoria Amsterdam 9.5

1 Pillows Anna van den Vondel Amsterdam 9.4

2 The Toren 9.4

3 The Hoxton Amsterdam 9.3

4 Ambassade Hotel 9.3

Hotel\_Address

0 Herengracht 542 556 Amsterdam City Center 1017...

1 Anna van den Vondelstraat 6 Oud West 1054 GZ A...

2 Keizersgracht 164 Amsterdam City Center 1015 C...

3 Herengracht 255 Amsterdam City Center 1016 BJ ...

4 Herengracht 341 Amsterdam City Center 1016 AZ ...

**Multilingual Support:**

Then, we worked on multilingual support, enabling the users to enter in specific language and still get the same recommendations. We have implemented in the French and Spanish languages, respectively. This demonstrates that the code accurately detects the language of the description and processes it accordingly.

#Example with English description

location = "netherlands"

description = "I am going for fun"

tags = "Solo traveler"

recommended\_hotels = recommend\_hotel(location, description, tags)

print(recommended\_hotels)

Output:

Hotel\_Name Average\_Score \

0 Waldorf Astoria Amsterdam 9.5

1 The Toren 9.4

2 Pillows Anna van den Vondel Amsterdam 9.4

3 Canal House 9.3

4 Luxury Suites Amsterdam 9.3

Hotel\_Address

0 Herengracht 542 556 Amsterdam City Center 1017...

1 Keizersgracht 164 Amsterdam City Center 1015 C...

2 Anna van den Vondelstraat 6 Oud West 1054 GZ A...

3 Keizersgracht 148 Amsterdam City Center 1015 C...

4 Oudeschans 75 Amsterdam City Center 1011 KW Am...

#Example with French description

description\_fr = "Je pars pour m'amuser"

recommended\_hotels\_fr = recommend\_hotel(location, description\_fr, tags)

print("\nRecommended Hotels (French):")

print(recommended\_hotels\_fr)

Output:

Recommended Hotels (French):

Hotel\_Name Average\_Score \

0 Waldorf Astoria Amsterdam 9.5

1 The Toren 9.4

2 Pillows Anna van den Vondel Amsterdam 9.4

3 Canal House 9.3

4 Luxury Suites Amsterdam 9.3

Hotel\_Address

0 Herengracht 542 556 Amsterdam City Center 1017...

1 Keizersgracht 164 Amsterdam City Center 1015 C...

2 Anna van den Vondelstraat 6 Oud West 1054 GZ A...

3 Keizersgracht 148 Amsterdam City Center 1015 C...

4 Oudeschans 75 Amsterdam City Center 1011 KW Am...

#Example with Spanish description

description\_es = "Voy a divertirme"

recommended\_hotels\_es = recommend\_hotel(location, description\_es, tags)

print("\nRecommended Hotels (Spanish):")

print(recommended\_hotels\_es)

Output:

Recommended Hotels (Spanish):

Hotel\_Name Average\_Score \

0 Waldorf Astoria Amsterdam 9.5

1 The Toren 9.4

2 Pillows Anna van den Vondel Amsterdam 9.4

3 Canal House 9.3

4 Luxury Suites Amsterdam 9.3

Hotel\_Address

0 Herengracht 542 556 Amsterdam City Center 1017...

1 Keizersgracht 164 Amsterdam City Center 1015 C...

2 Anna van den Vondelstraat 6 Oud West 1054 GZ A...

3 Keizersgracht 148 Amsterdam City Center 1015 C...

4 Oudeschans 75 Amsterdam City Center 1011 KW Am...

Overall, by leveraging both aspect-based sentiment analysis and a recommendation system, the solution successfully provided personalized hotel recommendations tailored to the user's preferences and aligned with the sentiment analysis results. This integration enhanced the user experience by offering relevant and satisfactory hotel options.

**Setting up and Running the System:**

* Dataset: Download the Hotel Reviews dataset from the link provided in the code.
* Dependencies: Install necessary dependencies using pip install commands provided in the code (e.g., nltk, spacy, wordcloud).
* Models: Download and install spaCy language models for English, French, and Spanish using python -m spacy download.
* Running the Program: Copy and paste the provided code into a Python environment and execute it.
* Input: Provide location, description, and tags as input parameters to the recommend\_hotel function.
* Output: View the top recommended hotels with their average scores and addresses.

By following these steps, users can set up and run the system to perform sentiment analysis, aspect-based sentiment analysis, and hotel recommendation based on user input.

**Running on Jupyter Notebook:**

* Install Jupyter Notebook: If you have not already, you can install Jupyter Notebook using pip:
* Copy code
* pip install notebook
* Launch Jupyter Notebook: Open your terminal or command prompt and type:
* Copy code
* jupyter notebook
* This will open a new tab in your default web browser.
* Create a New Notebook: Click on the "New" button on the right-hand side and select "Python 3" to create a new notebook.
* Copy and Paste Code: Copy the provided code into the cells of the notebook.
* Run Cells: Run each cell by clicking on it and pressing Shift + Enter or by clicking the "Run" button in the toolbar.

**Running on Visual Studio Code:**

* Install Visual Studio Code: Download and install Visual Studio Code from their official website.
* Create Workspace: Once Visual Studio Code is installed, create a new workspace by selecting "File"> "Open Folder" and choosing a directory to work in.
* Open Notebook: If you have a Jupyter notebook file (.ipynb), you can open it directly in Visual Studio Code. Otherwise, create a new Python file (.py) and copy the code into it.
* Install Required Libraries: Open a terminal in Visual Studio Code (Terminal > New Terminal) and install the required libraries using pip:
* Copy code
* pip install pandas nltk wordcloud matplotlib spacy langdetect numpy
* Run Cells: If you are working with a Jupyter notebook, you can run each cell individually by clicking on it and pressing Shift + Enter or by clicking the "Run Cell" button. If you are working with a Python script, you can run the entire script by pressing F5 or clicking "Run Python File in Terminal" from the context menu.

**Dataset link:**

We have taken the dataset from Kaggle. Here is the link:

<https://www.kaggle.com/datasets/jiashenliu/515k-hotel-reviews-data-in-europe?resource=download>

**Acknowledgements:**

We thank Professor Lindi Liao, PhD, for her guidance and support throughout this project's development. Her expertise in Natural Language Processing (NLP) and sentiment analysis has been instrumental in shaping our approach and achieving meaningful results.

Professor: Dr. Lindi Liao Duoduo ([dliao2@gmu.edu](mailto:dliao2@gmu.edu))

Thank you,

Meghna Kattekola (mkatteko@gmu.edu)

Ramya Sri Thallapally (rthallap@gmu.edu)

Srija Anasuri(sanasuri@gmu.edu)