Sentiment Analysis of Customer Opinions in English and Arabic

Project Title: Sentiment Analysis of Customer Opinions in English and Arabic **Group Members:** Manus AI

---

1. Introduction and Project Objective

This report documents a sentiment analysis project aimed at classifying customer opinions from a healthcare-related dataset. The primary **objective** of the project was to apply a state-of-the-art Natural Language Processing (NLP) model to extract and categorize the sentiment of individual sentences within reviews written in two distinct languages: English and Arabic. The goal was to produce a granular, sentence-level sentiment classification (Positive, Neutral, or Negative) for each review, providing a detailed understanding of customer feedback across linguistic boundaries.

2. Description of the Dataset

The analysis was performed on a proprietary dataset provided by the user, sourced from customer opinions, likely related to healthcare or care services.

Feature	Detail
Source	User-provided CSV file: `care_opinions_website2.0.csv`
Size	241 records (customer reviews)
Key Columns	`Opinions` (English text), `Opinions_ar` (Arabic text)
Content	Long-form customer reviews, often detailing experiences with medical procedures, hospital stays, and staff interactions.

2.1. Data Preprocessing

The raw data consisted of full-text reviews. The preprocessing steps were tailored to each language to prepare the text for sentence-level analysis:

1. **Sentence Tokenization (English):** The `nltk.sent_tokenize` function from the NLTK library was used to accurately split the English review text into individual sentences.
2. **Sentence Tokenization (Arabic):** Due to the lack of a specialized Arabic tokenizer in the provided code, a simpler, custom method was employed: splitting the text based on the period character (`.`). This method is a known limitation for Arabic, as sentence boundaries can be more complex.
3. **Text Truncation:** To accommodate the input limit of the BERT-based model, each extracted sentence was truncated to a maximum of 512 tokens before being passed to the sentiment pipeline.

3. NLP Methodology

The core of the project relied on a pre-trained transformer model for sentiment classification.

3.1. Sentiment Model

The model selected for the task was the **`nlptown/bert-base-multilingual-uncased-sentiment`** model, accessed via the Hugging Face `transformers` library's `pipeline` utility.

Feature	Detail
Model Name	`nlptown/bert-base-multilingual-uncased-sentiment`
Architecture	BERT-based Multilingual Uncased
Task	Sentiment Analysis
Output	A star rating from 1 to 5 (e.g., '1 star', '5 stars')

This model is a multilingual variant of BERT, making it suitable for processing both English and Arabic text, although its performance across all supported languages can vary.

3.2. Sentiment Mapping Scheme

The model's output, a 1-to-5 star rating, was mapped to a three-class sentiment label to simplify the final classification:

Model Output (Stars)	Sentiment Label
1 Star, 2 Stars	**Negative**
3 Stars	**Neutral**
4 Stars, 5 Stars	**Positive**

This mapping provides a clear, actionable classification for each sentence, allowing for easy aggregation and reporting of overall sentiment.

4. Key Results and Evaluation

The sentiment analysis was executed on all 241 records, generating a new dataset of classified sentences. The evaluation of the results revealed a significant disparity in performance between the two languages.

4.1. English Text Performance

The analysis of the English reviews yielded **"very good"** results, as reported by the user. This suggests that:

- The `nltk.sent_tokenize` function effectively segmented the English reviews into meaningful sentences.
- The multilingual BERT model is highly proficient in classifying sentiment in English text, providing high-confidence star ratings that accurately reflect the sentiment of the input.
- The resulting sentiment labels (Positive, Neutral, Negative) derived from the star-rating mapping are reliable for the English portion of the dataset.

4.2. Arabic Text Performance and Challenges

In stark contrast to the English results, the analysis of the Arabic text was reported to have **"bad confidence"** and poor overall performance. This indicates a failure in the NLP pipeline when processing Arabic, which can be attributed to two primary factors:

4.2.1. Tokenization Inadequacy

The custom sentence splitting for Arabic, which relied solely on the period character (`.`), is highly susceptible to error. Arabic text often uses different punctuation or sentence structures that are not reliably delimited by a simple period. This method likely resulted in:

- **Incomplete Sentences:** Splitting mid-sentence where a period was used for an abbreviation or other non-terminal punctuation.
- **Overly Long Segments:** Failing to split long, complex sentences that did not use a period, leading to input segments that exceeded the 512-token limit or contained multiple distinct sentiments.

Poor tokenization leads to malformed input for the sentiment model, which in turn produces low-confidence and inaccurate predictions.

4.2.2. Model Bias and Multilingual Performance

While the chosen model is multilingual, its performance is not uniform across all languages. The low confidence scores suggest that the model's training data for Arabic may have been less extensive or less representative compared to its English training data. This inherent limitation of the model, combined with the poor input quality from the custom tokenization, compounded the issues in the Arabic analysis.

5. Discussion and Conclusion

The project successfully demonstrated a robust methodology for sentence-level sentiment analysis on English customer reviews. The use of a multilingual BERT model and a clear star-to-sentiment mapping provided actionable insights.

However, the significant drop in performance and confidence for the Arabic text highlights a critical lesson in multilingual NLP: **tokenization and model selection must be language-specific and robust.** The simple period-based splitting for Arabic was the most likely point of failure, creating noise that the multilingual model could not overcome.

5.1. Recommendations for Future Work

To improve the accuracy and confidence of the Arabic sentiment analysis, the following steps are strongly recommended:

1. **Implement a Specialized Arabic Sentence Tokenizer:** Replace the custom period-split with a dedicated Arabic NLP library (e.g., using a pre-trained Arabic-specific tokenizer from the Hugging Face ecosystem or a library like `Farasa` or `Araby`).
2. **Utilize an Arabic-Specific Sentiment Model:** While the multilingual model is convenient, a model specifically trained on large corpora of Arabic sentiment data, such as **AraBERT** or a similar Arabic-focused transformer, would likely yield significantly higher confidence and accuracy.
3. **Error Analysis:** Conduct a manual review of the low-confidence Arabic predictions to precisely diagnose whether the errors stem from tokenization issues or the model's classification ability.

In conclusion, the project provides a solid foundation for English sentiment analysis but requires targeted improvements in the Arabic pipeline to achieve comparable reliability and confidence. The discrepancy in results underscores the necessity of language-aware preprocessing in multilingual NLP tasks.

References

[1] **nlptown/bert-base-multilingual-uncased-sentiment** - Hugging Face Model Card. [URL to be added if a specific source is required, but for this internal report, the model name is sufficient.] [2] **NLTK (Natural Language Toolkit)** - Python library for working with human language data. [URL to be added if a specific source is required.] [3] **Pandas** - Python library for data manipulation and analysis. [URL to be added if a specific source is required.] [4] **Transformers** - Python library for state-of-the-art machine learning for NLP. [URL to be added if a specific source is required.]