Text Sentiment Classification

Nur Farah Husna binti Junaidi (2024148067)

1 Introduction

This project explores the design of a simple AI pipeline for classifying short English sentences into either positive or negative sentiment. In this project, I compared two methods:

- A naïve keyword-based baseline
- A pre-trained transformer model accessed through the HuggingFace `pipeline` API.

Sentiment classification is an ideal task for exploring AI concepts because it is easy to understand, widely used in many practical systems, and requires only small computational resources. Through this project, I was able to experience the complete workflow of building an AI system, from defining the task, designing a pipeline, evaluating results, and reflecting on the model behavior.

2 Task Definition

- **Task description:** Classifying short English sentences as either `positive` or `negative` sentiment.
- **Motivation:** Sentiment analysis is a known, useful task widely applied in real-world applications such as in customer reviews and user feedback analysis in the industry. It is also a common NLP downstream task, making it simple yet meaningful setting to compare a rule-based approach with a modern transformer model.
- **Input / Output:** The input is a natural, English language sentence, while the output is a sentiment labeled as either `positive` or `negative`.
- **Success criteria:** The model should achieve high accuracy and correctly classify subtle or context-dependent sentiment expressions.

3 Methods

This section includes both the naïve baseline and the improved AI pipeline.

3.1 Naïve Baseline

- **Method description:** The baseline uses a small set of predefined positive and negative keywords. If a sentence contains any positive word (e.g., “great”, “happy”, “satisfied”), it is labeled `positive`. Conversely, if it contains negative keywords (e.g., “bad”, “sad”, “terrible”), it is labeled `negative`. If the number of positive and negative keywords is the same, or if no keywords appear, the classifier defaults to `negative`.
- **Why naïve:** The method cannot handle negation (e.g., “not good”), sarcasm, where the sentence expresses different sentiment than what it appear on the surface, as well as subtle

emotional phrasing. It also fails when a sentence contains mixed emotions or no obvious keywords.

- **Likely failure modes:**
  - Sentences with no sentiment keywords.
  - Sentences with complex structure (e.g.,"I was tired, but the day felt rewarding").
  - Implicit emotional expressions (e.g.,"The presentation went better than expected").

3.2 AI Pipeline

- **Models used:** I used the HuggingFace `sentiment-analysis` pipeline [1] together with the pretrained model `distilbert-base-uncased-finetuned-sst-2-english` [2].
- **Pipeline stages:**
  1. Preprocessing and tokenization (handled automatically).
  2. Embedding sentences using DistilBERT transformer layers.
  3. Classification into positive or negative sentiment.
  4. Probability-based label assignment.
- **Design choices and justification:** DistilBERT is lightweight, efficient, and well-suited for small projects while still delivering strong performance on sentiment tasks. Using pretrained models allows us to build an effective pipeline without fine-tuning.

4 Experiments

4.1 Datasets

- **Source:** A custom dataset of 40 manually written sentences representing daily emotional expressions.
- **Total examples:** 40 (20 positive, 20 negative).
- **Train/Test split:** No training required; all examples used for evaluation.
- **Preprocessing steps:** Lowercasing for baseline matching. The transformer pipeline handles tokenization automatically.

4.2 Metrics

In this project, accuracy is used as the main metric. Since the dataset is balanced and the task is binary classification, accuracy is appropriate and easy to interpret.

4.3 Results

Method	Accuracy
Baseline	0.80
AI Pipeline	1.00

Qualitative Error Examples.

- "The presentation went better than I thought."
  True: positive, Baseline: negative, AI: positive

- "Even though I was tired, the day felt rewarding."
  True: positive, Baseline: negative, AI: positive
- "I feel anxious and unhappy about this."
  True: negative, Baseline: positive, AI: negative

5 Reflection and Limitations

The experiment demonstrates the gap between simple rule-based systems and pretrained transformer models. The baseline worked reasonably well for explicit sentiment expressions but failed on complex or subtle emotional statements. In contrast, the AI pipeline achieved perfect accuracy on the dataset and handled context such as contrastive phrasing and emotional implication more effectively.

However, since the dataset is small, the reported accuracy likely overestimates real-world performance. In this case, a larger, diverse dataset might provide more realistic evaluation, likely reducing accuracy and expose additional weakness.

In addition, accuracy was an appropriate metric for this balanced dataset, but it does not reveal detailed failure patterns, so additional metrics such as F1-score would be more informative in more complex settings.

If I have more time, I would expand the dataset, experiment with a three-way sentiment classification (positive, neutral, and negative), and attempt lightweight fine-tuning to analyze how performance changes under different conditions.

References

[1] Thomas Wolf et al. Transformers: State-of-the-art natural language processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, 2020.

[2] HuggingFace. Distilbert base uncased finetuned sst-2. https://huggingface.co/distilbert-base-uncased-finetuned-sst-2-english, 2019.