Fine-Tuned DeBERTa for Opinion Mining

Aspect-Based Sentiment Analysis Using LoRA

Presented by:

Esraa Ammar
Farida Khaled
Muhammed Tarek
Muhammed Yasser
Madiha Saeid
Hania Ruby

Course: Natural Language Understanding

Supervisor: TA Doaa Abdullah

Faculty of Computers & Artificial Intelligence
Helwan University

Project Objective

Objective

The goal of this project is to apply a fine-tuned DeBERTa model using Low-Rank Adaptation (LoRA) to perform Aspect-Based Sentiment Analysis (ABSA) on customer reviews. The model extracts both sentiment polarity (positive, negative, neutral) and the specific product aspects mentioned in each review.

Motivation

Traditional sentiment analysis provides an overall sentiment but lacks granularity. Aspect-based sentiment analysis offers deeper insights by identifying which parts of a product are liked or disliked. This is crucial for businesses to improve their products based on targeted feedback.

Faculty of Computers & Artificial Intelligence
Helwan University

Dataset Information

Dataset Used: We used the Laptop Reviews ABSA Dataset from Kaggle, which contains annotated laptop product reviews for Aspect-Based Sentiment Analysis (ABSA).

Source: Kaggle - Laptop ABSA Dataset

Size:

- ~3,045 customer reviews
- Each sentence includes annotated aspect terms and associated sentiment labels

Structure:

- Text: Sentence from a product review
- Aspect Term: A specific feature mentioned (e.g., "battery life")
- Sentiment: Label indicating sentiment polarity (Positive, Negative, Neutral)

Preprocessing Steps

- Datastet was already Cleaned
- Tokenized sentences using DeBERTa tokenizer
- Extracted aspect term spans and encoded them
- Created input-label pairs for training

Preprocessing & Task Overview

We handled two subtasks using the same DeBERTa model backbone:

1. Feature Extraction (NER)

- Applied BIO tagging to detect features like brand, color, and product aspects.
- Custom tags: B-ASP, B-BRAND, B-FEATURE, etc.
- Tokenization handled with DeBERTa tokenizer, aligning labels at token level.
- Final format: input_ids, attention_mask, labels

2. Sentiment Classification

- Converted each review into:
- "Aspect: [aspect]. Sentence: [review]"
- Sentiment mapped to labels:
  - Positive → 2
  - Neutral → 1
  - Negative → 0
- Tokenized and prepared for classification task.

Base Model & LoRA Fine-Tuning

Base Model

- We used DeBERTaV3-base, a pre-trained transformer model from HuggingFace.
- Tasks:
  - Token Classification (NER)
  - Sequence Classification (Sentiment)

LoRA Fine-Tuning

- Applied Low-Rank Adaptation (LoRA) to reduce training cost while retaining high performance.
- Injected trainable rank-decomposed matrices into attention layers.
- Enabled fast adaptation with fewer parameters.

Evaluation & Results

NER Task (Token Classification)

- Accuracy improved significantly from 14% → 89%
- F1-Score for non-target class (O) increased from 0.27 → 0.95
- F1-Score for aspect terms (B-ASP) improved from 0.00 → 0.57
- Macro F1 Score rose from 0.04 → 0.40, indicating better overall label performance
- Eval Loss dropped drastically from 430.15 → 1.30

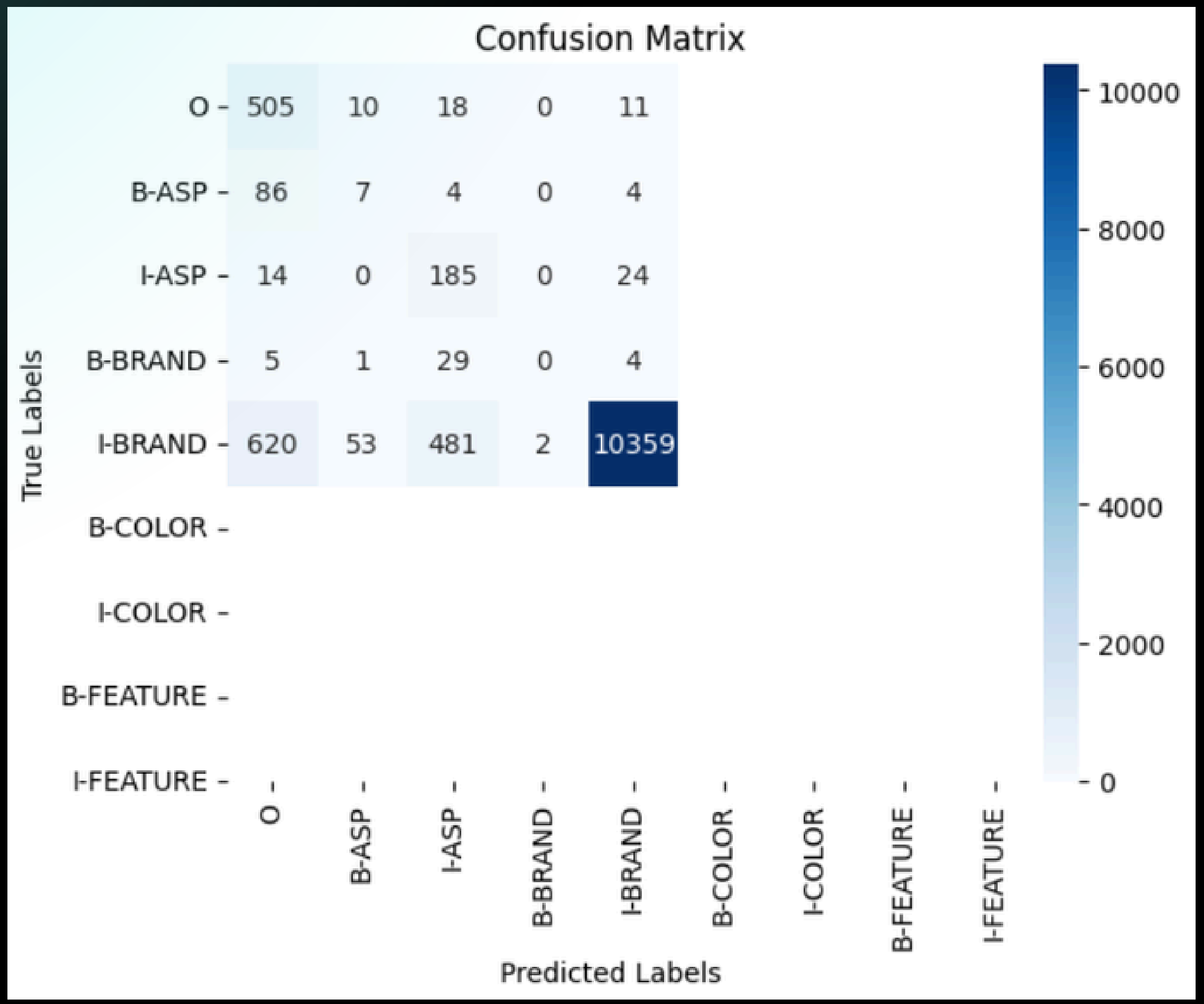
Sentiment Task (Sequence Classification)

- Accuracy reached 88%, showing strong overall sentiment detection
- F1-Score for Positive sentiment: 0.91
- F1-Score for Negative sentiment: 0.92
- F1-Score for Neutral sentiment: 0.00 (model struggled with this class)
- Macro F1 Score: 0.61 – indicates imbalance in class performance
- Training Loss: 0.51
- Eval Loss: 0.44

Result Visualization

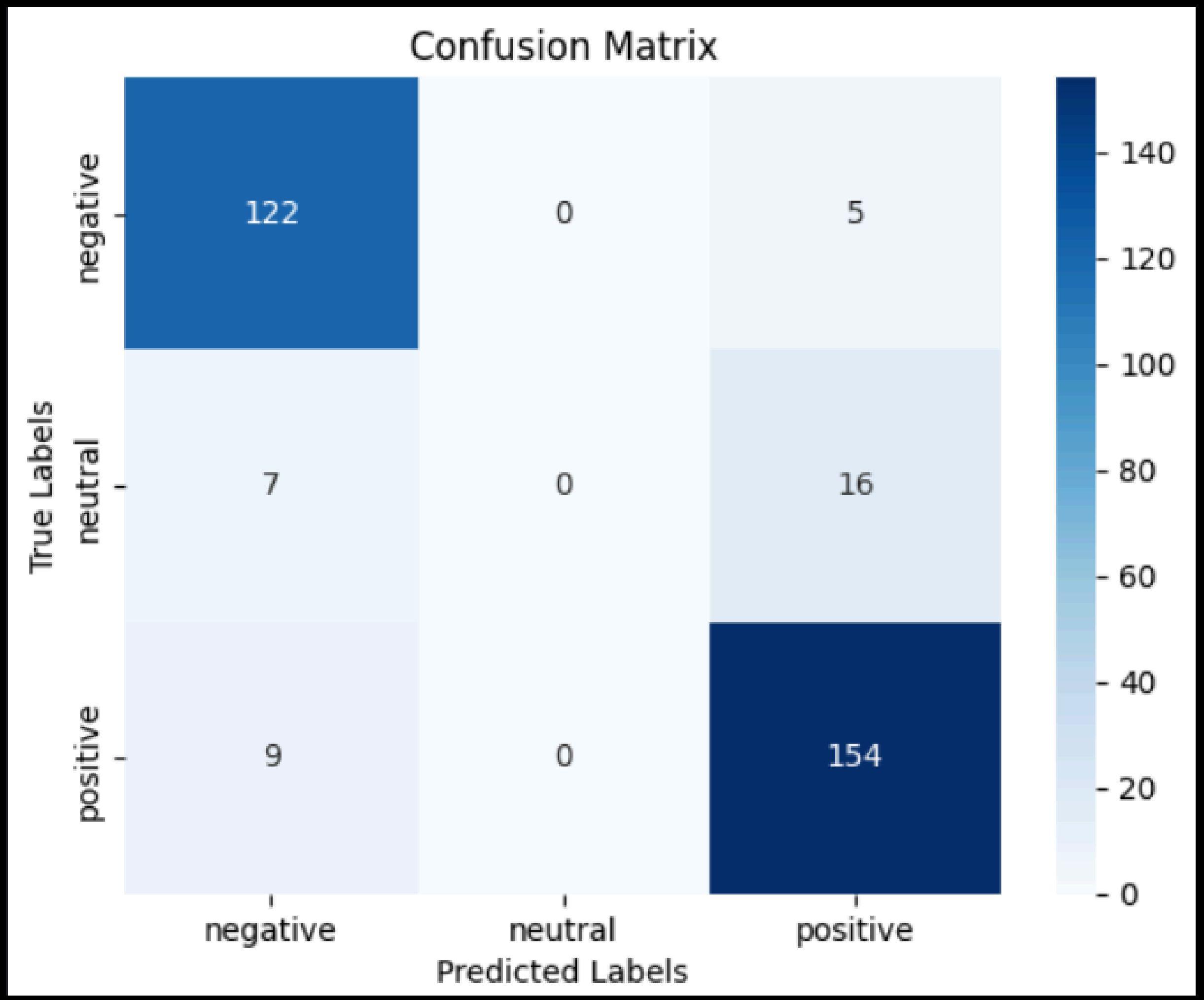
NER Task – Token Classification

Confusion Matrix (LoRA Model)

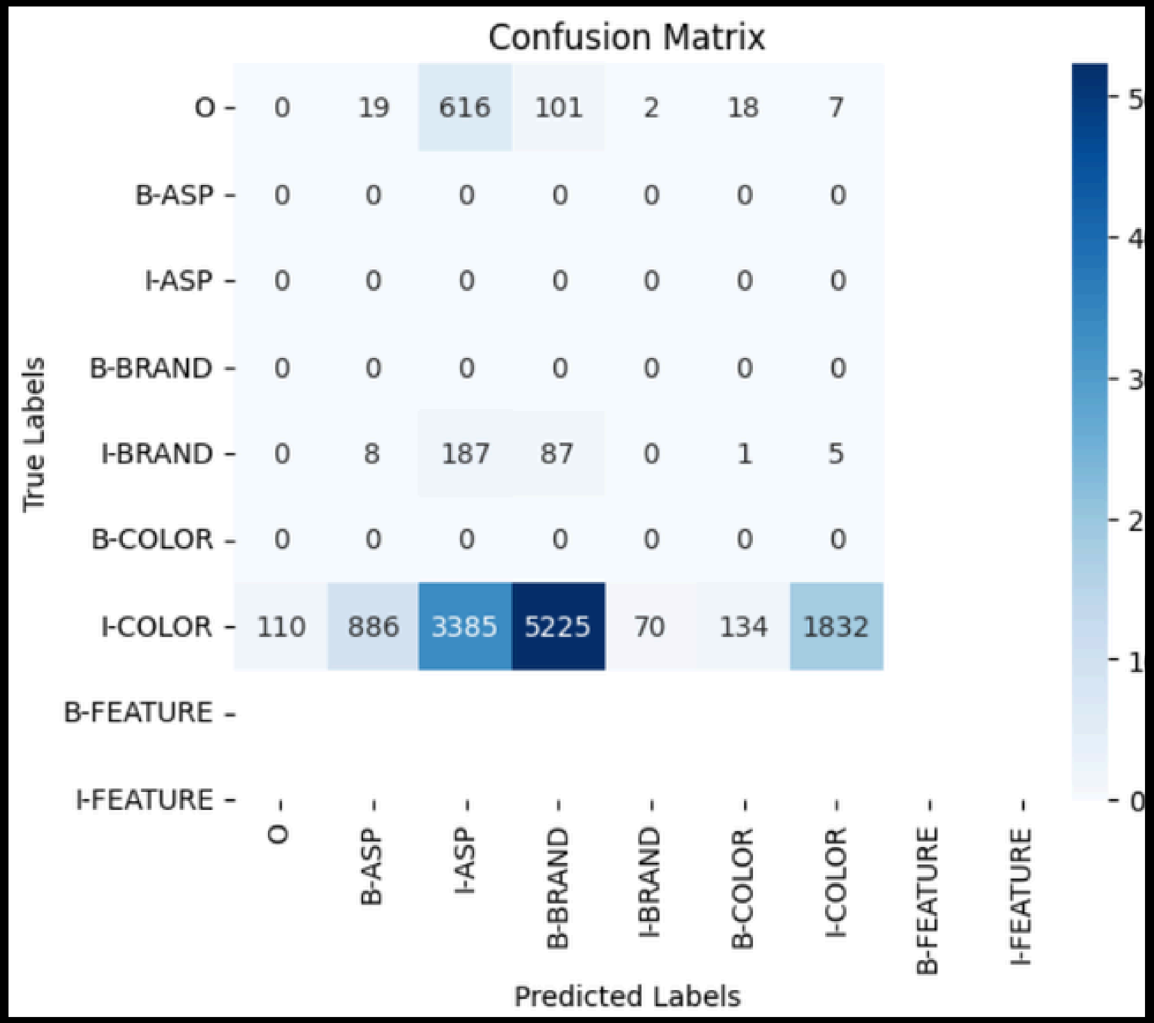


Sentiment Classification

Confusion Matrix (LoRA Model)



Base Model



Structured Output

NER Task Output Example

```
{
  "tokens": ["The", "Dell", "XPS", "is", "fast", "and", "lightweight"],
  "tags": ["O", "B-BRAND", "I-BRAND", "O", "B-FEATURE", "O", "I-
FEATURE"]
}
```

Entities identified:

→ Brand: Dell XPS

→ Feature: fast and lightweight

Sentiment Task Output Example

```
{
  "input": "Aspect: battery. Sentence: The battery life is disappointing.",
  "predicted_sentiment": "negative"
}
```

- Extracted Aspect: battery
- Model sentiment: negative

Conclusion & Key Takeaways

Project Summary

- Applied LoRA-based fine-tuning on a DeBERTa model for:
  - Aspect Term Extraction (Token Classification)
  - Sentiment Analysis (Sequence Classification)

Performance Improvement

- Accuracy jumped from 14% to 89% on token classification after fine-tuning
- Significant improvements across precision, recall, and F1-score

Handling Class Imbalance

- Addressed imbalance using custom class weighting in loss function:
  - Computed weights based on observed label distribution
  - Integrated via a custom Trainer class with CrossEntropyLoss(weight=...)
  - Resulted in better learning for underrepresented classes

Thank You