มหาวิทยาลัยอัสสัมชัญ
ASSUMPTION UNIVERSITY
of THAILAND

VINCENT MARY
SCHOOL of SCIENCE and
TECHNOLOGY

Senior Project Report

Pros and Cons Extraction from Reviews

Lu Phone Maw
Wai Yan Paing

Asst.Prof.Dr. Thanachai Thumthawatworn

CS 3200 Senior Project 1 (2/2024)

Senior Project Approval

Project title:	Pros and Cons Extraction from Reviews
Academic Year:	2/2024
Authors:	Lu Phone Maw(6511157)
	Wai Yan Paing (6511171)
Project Advisor:	Asst.Prof.Dr. Thanachai Thumthawatworn

The Senior Project committee's cooperation between the Department of Computer Science and Information Technology, Vincent Mary School of Engineering, Science and Technology, Assumption University had approved this Senior Project. The Senior Project in partial fulfilment of the requirement for the degree of Bachelor of Science in Computer Science and Information technology.

Approval Committee:

……..
(Asst.Prof.Dr Thanachai Thumthawatworn)
Project Advisor

……..
(Chayapol Moemeng)
Committee Member

……..
(Dr. Kwankamol Nongpong)
Committee Member

Abstract

Aspect-Based Sentiment Analysis (ABSA) focuses on identifying specific aspects or features mentioned in user-generated text and determining the sentiment expressed toward each. Inspired by the InstructABSA framework, we leverage an InstructABSA-style prompt to instruction tune large language models for two subtasks: aspect extraction and aspect-given sentiment classification. Specifically, we fine-tune Llama 3 (8B) and Mistral (7B) on the SemEval 2014 Restaurant (Res14) and Laptop (Lap14) datasets. Our approach achieves F1 scores of 92.1% on Res14 and 92.3% on Lap14 for aspect extraction—substantially higher than plain QLoRA fine-tuning baselines (e.g., Llama 2: 71.94% on Res14, 71.66% on Lap14; Mistral: 81.33% on Res14, 77.65% on Lap14)—and yields competitive performance for aspect-given sentiment classification (e.g., F1 of 85.17% on Res14 and 81.56% on Lap14), outperforming corresponding results from the Instruct-DeBERTa pipeline. Our novel contribution lies in the effective use of an InstructABSA-style prompt, which departs from traditional fine-tuning strategies by explicitly guiding the model with task-specific instructions and examples. This research not only advances the methodological framework for instruction tuning in ABSA but also has significant real-world impact, improving review analysis for consumer insights in both the restaurant and laptop domains.

Table Of Contents

Chapter 1: Introduction

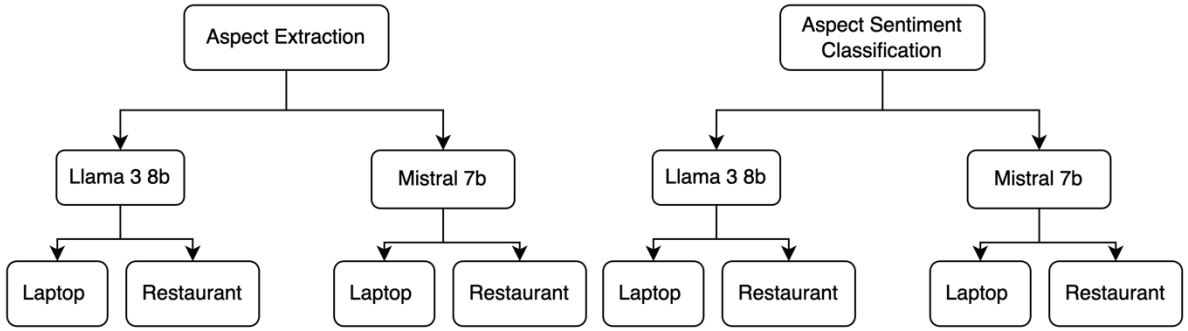
1.1 Problem Statement

Despite the recent surge in research leveraging large language models (LLMs) for various natural language processing tasks, many studies on aspect-based sentiment analysis (ABSA) have not fully exploited the inherent capabilities of models like Llama and Mistral. Traditionally, ABSA research has relied on conventional fine-tuning approaches or pipelines such as the Instruct-DeBERTa method, which, while effective to an extent, tend to underutilize the instruction-following abilities of modern LLMs. These models possess remarkable language understanding and generation skills, yet when applied to ABSA tasks, they are often tuned without explicit, task-specific prompts that could guide them more precisely.

In many cases, the fine-tuning process overlooks the potential of leveraging detailed instructions and carefully crafted examples—strategies that have been shown to significantly enhance performance on complex tasks. As a result, baseline methods achieve suboptimal F1 scores, reflecting a gap between the models' true potential and their actual performance when using standard fine-tuning techniques. This research addresses this shortfall by adopting an InstructABSA-style prompt that explicitly guides the models in both aspect extraction and aspect-given sentiment classification tasks, thereby tapping into the full capacity of LLMs to understand and follow complex instructions.

1.2 Scope of the project

This project aims to bridge the performance gap in Aspect-Based Sentiment Analysis (ABSA) by fine-tuning state-of-the-art large language models—Llama 3 and Mistral 7B v0.3—on two core tasks: Aspect Term Extraction (ATE) and Aspect Sentiment Classification (ASC). The evaluation is conducted over two distinct domains, namely laptop and restaurant reviews, resulting in eight separate training configurations (2 models × 2 domains × 2 tasks).



Detailed Tasks

- **Aspect Term Extraction (ATE):**
  The task involves identifying specific aspect terms or features (e.g., "battery life," "menu") mentioned within a review. This fine-grained extraction is crucial for isolating the relevant parts of a text that will later be analyzed for sentiment.
- **Aspect Sentiment Classification (ASC):**
  Once aspects are extracted, the next step is to determine the sentiment expressed towards each aspect, classifying them as positive, negative, or neutral. For cases where no aspect is identified, the output is set to "noaspectterm." In this project, the "conflict" polarity is deliberately removed, as previous research in ABSA consistently omits this category due to its class imbalance and the scarcity of conflict examples.

Task	Input	Output
Aspect Term Extraction (ATE)	S_i	price, restaurant
Aspect Sentiment Classification (ASC)	$S_i + price, S_i + restaurant$	Negative, Positive

Figure 1. Overview of the task for ABSA

Limitations and Data Considerations

Despite its promise, ABSA remains a relatively new field in Natural Language Processing. A significant challenge is the limited availability of comprehensive, high-quality datasets for ABSA. Most publicly available datasets are confined to only a few domains—primarily laptop and restaurant reviews—because these domains have been the focus of much of the early research in ABSA. **The removal of the "conflict" polarity further reflects this trend;** researchers typically exclude conflict due to its infrequent occurrence and the difficulty in modeling an imbalanced class distribution.

Project Objectives

The specific objectives of this project are to:

1. **Fine-Tune LLMs for ABSA:**
   Apply state-of-the-art parameter-efficient fine-tuning techniques, such as Quantized Low-Rank Adaptation (QLoRA) and LoRA-based adapters with 4-bit quantization, to Llama 3 and Mistral 7B v0.3 for both ATE and ASC tasks.
2. **Domain-Specific Evaluation:**
   Evaluate the models across two domains (laptop and restaurant reviews), which are the most well-represented in existing ABSA datasets. This will help determine how well the models generalize within these contexts.
3. **Benchmark Against State-of-the-Art:**
   Compare the performance of the fine-tuned models with existing approaches, including models like Instruct-DeBERTa & Instruct-ABSA, to assess the extent to which our fine-tuning methods close the performance gap.
4. **Address Data Limitations:**
   Acknowledge and work within the constraints posed by the limited availability of ABSA datasets and domains. By focusing on laptop and restaurant reviews and

excluding the "conflict" polarity, the project is aligned with the prevailing research practices and dataset characteristics in the ABSA community.

Overall, this project seeks to unlock the full potential of Llama 3 and Mistral 7B v0.3 for ABSA tasks, providing both a rigorous academic investigation and a scalable framework for real-world sentiment analysis applications.

Chapter 2: Related Work

2.1 Large Language Models for ABSA: Llama 2 7B and Mistral 7B (InstructDeBERTa paper)

Model	F1-score(%)			
	Res-14		Lap-14	
	Aspect Extraction	Sentiment Polarity	Aspect Extraction	Sentiment Polarity
Llama 2 7b with QLoRa	71.94	69.29	71.66	66.53
Mistral 7b with QLoRa	81.33	76.46	77.65	72.40

Figure 2. Performance score for Llama 2 7b and Mistral 7b without Instruction Tuning

Recent research in aspect-based sentiment analysis (ABSA) has explored a variety of fine-tuning approaches for large language models (LLMs) to enhance performance on tasks such as aspect extraction and aspect-given sentiment classification. In InstructDeBERTa research, experimental studies have included models like **Llama 2 7B** and **Mistral 7B** as baselines, where these models are fine-tuned using conventional supervised learning techniques without employing specialized instruction-tuning paradigms.

In these studies, Llama 2 7B and Mistral 7B were fine-tuned on benchmark datasets (Figure 2)—such as the SemEval 2014 Restaurant (Res14) and Laptop (Lap14) datasets—using standard fine-tuning protocols. Their performance metrics, for example, include F1 scores of approximately 71.94% (Res14) and 71.66% (Lap14) for Llama 2 7B, and 81.33% (Res14) and 77.65% (Lap14) for Mistral 7B on the aspect extraction task. For aspect-given sentiment classification, these models have shown macro F1 scores in the range of 60–70%. Although these results reflect the strong general language understanding capabilities of LLMs, they also indicate that conventional fine-tuning may not fully exploit the potential of these models for nuanced ABSA tasks.

Our work builds on this foundation by contrasting these baseline performances with results obtained through instruction tuning using an InstructABSA-style prompt, thereby aiming to unlock the full capacity of Llama and Mistral models in guiding them with explicit, task-specific instructions.

2.2 InstructABSA

Model	F1-score(%)			
	Res-14		Lap-14	
	Aspect Extraction	Sentiment Polarity	Aspect Extraction	Sentiment Polarity
InstructABSA	92.10	85.17	92.30	81.56

Figure 3. Performance Metrics for InstructABSA

InstructABSA (Scaria et al., 2023) adopts an **instruction-based** approach to aspect-based sentiment analysis (ABSA). The authors leverage a **T5** (Text-to-Text Transfer Transformer) backbone, which they further **instruction-tune** to handle a variety of ABSA subtasks, including Aspect Term Extraction (ATE) and Aspect Sentiment Classification (ASC). By framing each subtask as a natural language instruction, InstructABSA enables the model to better understand and respond to the specific requirements of each ABSA component.

1. **Methodology**
    - **Instruction Tuning:** InstructABSA prepends prompts (e.g., definitions, examples of positive/negative/neutral sentiment) to the input text. This prompt-based strategy ensures the model grasps the nuances of the ABSA tasks, reducing the need for task-specific architectures.
    - **Multi-Domain Evaluation:** The paper demonstrates the model's adaptability by evaluating on well-known SemEval datasets, spanning multiple domains such as **restaurant** (Res-14, Res-15, Res-16) and **laptop** (Lap-14) reviews.
    - **Few-Shot and Zero-Shot Performance:** InstructABSA excels even in limited-data scenarios, owing to the flexibility of T5-based instruction tuning. The authors note that the model can achieve near state-of-the-art results with a fraction of the training data.
2. **Performance Highlights (**Figure 3**)**
    - **Aspect Extraction:** InstructABSA consistently **surpasses 90% F1** on tasks like Res-14 and Lap-14 for aspect term extraction.
    - **Sentiment Classification:** The model's ASC results are also competitive, exceeding the performance of many existing systems.
    - **Generalization:** InstructABSA's instruction-tuned paradigm shows strong generalization across subtasks and domains, even outperforming larger language models that have not been specifically tuned for ABSA.

Overall, InstructABSA illustrates how a well-designed instruction-learning pipeline can effectively unify multiple ABSA subtasks under one framework, thereby reducing complexity and improving performance.

2.3 Instruct-DeBERTa

Model	F1-score(%)			
	Res-14		Lap-14	
	Aspect Extraction	Sentiment Polarity	Aspect Extraction	Sentiment Polarity
Instruct-DeBERTa	91.39	88.63	91.56	89.65

Figure 4. Performance Metrics of Instruct-DeBERTa pipeline

Building upon the successes of InstructABSA, **Instruct-DeBERTa** (Jayakody et al., 2024) introduces a **hybrid approach** that combines **InstructABSA** for aspect extraction with **DeBERTa-V3-base-absa-V1** for sentiment classification. The motivation behind this design is to leverage **best-of-breed** components for each subtask, rather than forcing a single model to handle all ABSA subtasks at once.

1. **Methodology**
    - **Pipeline Architecture:** Instruct-DeBERTa is effectively a two-stage system. First, **InstructABSA** handles Aspect Term Extraction (ATE), relying on its instruction-tuned T5 architecture to pinpoint relevant aspects. Next, the extracted aspects are passed to **DeBERTa-V3-base-absa-V1**, a specialized sentiment classifier that has been extensively fine-tuned for ASC.
    - **Domain-Independent Approach:** The authors demonstrate that this pipeline can handle diverse domains by evaluating on SemEval 2014 (restaurant, laptop), SemEval 2015, and SemEval 2016 datasets.
    - **Efficiency and Scalability:** Both components (InstructABSA for extraction, DeBERTa for classification) are used in a pipeline, avoiding the complexity of building a monolithic model that might be harder to fine-tune or adapt to different domains.
2. **Performance Highlights**
    - **State-of-the-Art Joint Task Results:** As shown in Figure 2, Instruct-DeBERTa outperforms existing models on the **joint** ATE+ASC tasks for both **restaurant** and **laptop** reviews, achieving F1 scores above 80%. It even surpasses 7B–13B parameter large language models that have not been instruction-tuned for ABSA.

- **Robustness Across Domains:** By separating the tasks into two specialized modules, Instruct-DeBERTa exhibits robust performance in both restaurant and laptop domains. The authors note minimal performance degradation when transferring between domains.
- **Comparison with Larger Models:** Although general-purpose models like Llama 2 and Mistral 7B offer strong language capabilities, Instruct-DeBERTa demonstrates that **domain-specific fine-tuning** and **instruction-based prompts** are critical for achieving top-tier ABSA performance.

In essence, Instruct-DeBERTa underscores the importance of **hybrid pipelines**: Instead of relying on a single model for all ABSA subtasks, it uses **InstructABSA** to maximize aspect extraction accuracy and **DeBERTa-V3-base** to refine sentiment classification. This design not only streamlines the training process but also capitalizes on each component's specialized strengths, resulting in a new state-of-the-art in ABSA.

It is important to note that both InstructABSA and Instruct-DeBERTa deliberately ignore the "conflict" polarity. This decision aligns with prevailing research practices in ABSA, where the conflict category is typically removed due to its severe class imbalance and the limited availability of annotated conflict examples. By focusing exclusively on positive, negative, and neutral polarities, these approaches mitigate performance degradation that might arise from modeling a sparsely represented class and ensure that the evaluation metrics are robust and comparable across studies.

Chapter 3: Proposed Methodology

3.1 Methodology

3.1.1 Overview

Our central hypothesis is:

"What if we use the instruction prompt from InstructABSA2 and apply it to Llama 3 and Mistral 7B? Would that yield better performance than existing results—possibly even surpassing current state-of-the-art models?"

InstructABSA2 utilizes detailed prompts that include a task definition followed by two positive, two negative, and two neutral examples. These examples are tailored for each domain and for both ATE and ASC tasks. By applying this instruction tuning approach to Llama 3 and Mistral 7B, we aim to enable these general-purpose models to capture the nuanced cues needed for ABSA, thereby enhancing performance while maintaining computational efficiency.

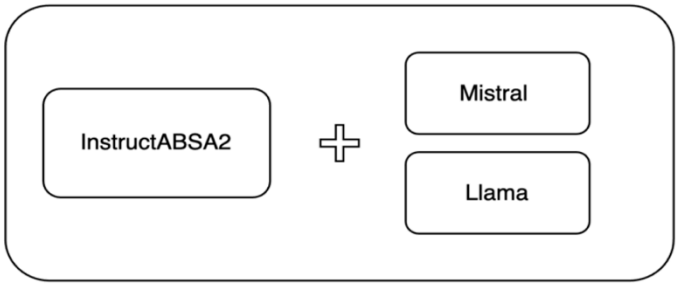


Figure 5. Our propose approach. Utilize InstructABSA2 prompt to instruction-tune LLM Mistral 7b and Llama 3 8b

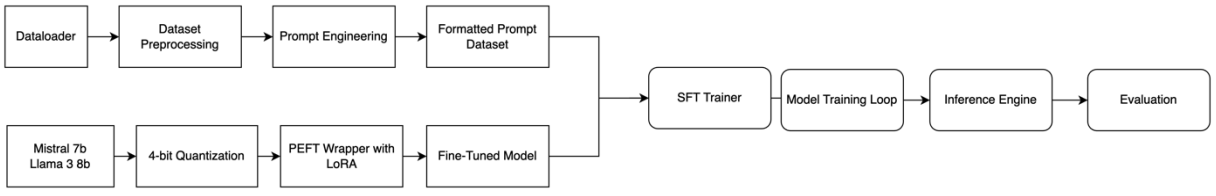


Figure 6. Instruction Tuning flow using quantization, LoRA, InstructABSA on Llama 3 8b and Mistral 7b

3.2 Data Preparation and Preprocessing

1. **Data Sources:**

   We use the SemEval 2014 (Restaurant and Laptop domain) which is used to foster research in the field of aspect-based sentiment analysis. Each record comprises a review text and annotated aspect terms and their respective sentiment polarity. Each domain consists of training data ~3000 samples, 100 validation data and 800 test data(Figure 7). As stated before, for Aspect Sentiment Classification task, we remove conflict polarity for the purpose of comparing against other's research.

Domain	Train	Test	Validation	Total
Restaurants	3041	800	100	3941
Laptops	3045	800	100	3945
Total	6086	1600	200	7886

Figure 7. Sample amounts for train, test and validation dataset

Dataset	Pos	Neg	Con	Neu	Tot.
LPT-TR	987	866	45	460	2358
LPT-TE	341	128	16	169	654
RES-TR	2164	805	91	633	3693
RES-TE	728	196	14	196	1134

Figure 8. Number of labels for each class

2. **Cleaning and Transformation:**
   - **Column Removal:** Unnecessary columns, such as `aspectCategories`, are removed to simplify the dataset.
   - **Aspect Extraction Transformation:**
     - The `aspectTerms` field is processed using Python's `ast.literal_eval` to convert string representations of lists into actual lists.

- If no valid aspect is present, the field is set to `"noaspectterm"`.

  Otherwise, the individual aspect terms (extracted via the `"term"` key) are concatenated into a comma-separated string.

- **Handling Polarity:**
  - For the ASC task, we filter out the `"conflict"` polarity. Prior research consistently omits this category due to class imbalance and the scarcity of conflict examples.

3.3 Prompt Construction

Our approach leverages an Alpaca-style prompt template to embed clear instructions and examples for both tasks.

1. **InstructABSA2-Style Prompts:**
   - **For ATE:**

     The prompt begins with a definition that describes the expected output—extracted aspect terms or `"noaspectterm"` if none are present. It includes two positive, two negative, and two neutral examples tailored for the domain.
   - **For ASC:**

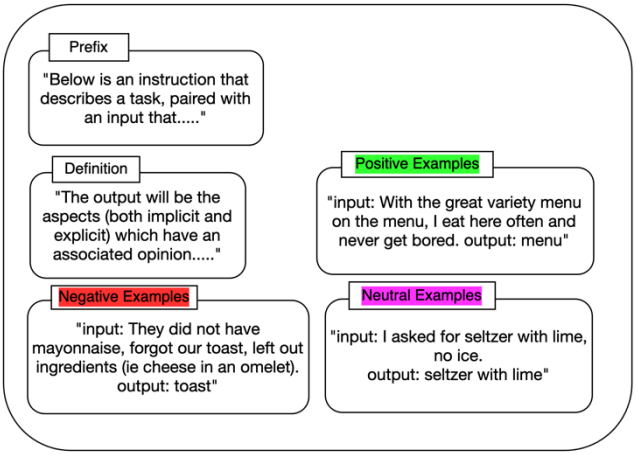
     A similar structure is used with a definition for classifying the sentiment of the identified aspect as `positive`, `negative`, or `neutral`. The "input aspect" is given by stating: "The aspect is …." at the end of every review sentence.
2. **Alpaca-Style Template:**

   Each prompt is formatted as follows:

```
Below is an instruction that describes a task, paired with an input that provides further context. Write a response that appropriately completes the request.
### Instruction:
{instruction_text}
### Input:
{review_text}
### Response:
{expected_output}
```

a. **Dynamic Insertion:**

The `{instruction_text}` is filled with our InstructABSA2-style prompt, `{review_text}` with the actual review, and `{expected_output}` is either provided (for training examples) or left blank for model generation during inference.



2. **Dataset Mapping:**

We map a custom function over the dataset that constructs these prompts for every example, saving the final prompt text into a new field (e.g., `"text_new"`) used for fine-tuning.

3.4 Model Setup and Instruction Tuning

Our approach leverages the parameter-efficient fine-tuning (PEFT) paradigm with LoRA. The key steps include:

- **Model Selection and Quantization:**
  We start with large language models—specifically, **Llama 3 (8B)** and **Mistral 7B**—using their 4-bit quantized variants to reduce memory consumption and improve inference speed.
- **LoRA-Based Fine-Tuning:**
  We integrate LoRA (Low-Rank Adaptation) to efficiently fine-tune the pre-trained models. This involves:
  - Specifying target modules (e.g., query, key, value, and output projections).
  - Setting hyperparameters such as LoRA rank, dropout, and alpha value.
  - Enabling gradient checkpointing to support longer contexts.
- **Instruction Tuning:**
  Instead of standard fine-tuning, our approach incorporates the InstructABSA prompt during training. The models are tuned to generate responses that strictly adhere to the given instructions, thus harnessing their full instruction-following capability.

3.5 Training Procedure

The fine-tuning is conducted using the SFTTrainer from the TRL library with the following key configurations:

- **Training Hyperparameters:**
  - **Batch Size and Gradient Accumulation:** Configured to maximize GPU utilization (e.g., per-device batch size of 8 with 4 gradient accumulation steps).
  - **Learning Rate and Warmup:** A learning rate of 3e-4 is used along with an appropriate number of warmup steps to stabilize training.
  - **Evaluation Strategy:** The model is evaluated at regular intervals (every 50 steps) on a validation split to monitor performance and avoid overfitting.
  - **Optimization and Mixed-Precision:** Optimizer settings such as `adamw_8bit` are applied, and mixed-precision training (bf16) is used depending on hardware support.
- **Dataset Processing:**
  The formatted prompt (created using the InstructABSA template) is mapped over the dataset splits (training, validation, test) before training.

The training process is executed for 400 steps, ensuring that the model learns to generalize well across both tasks.

3.6 Inference and Evaluation

After fine-tuning, the models are switched to inference mode to generate predictions on unseen data:

- **Inference:**
  Each test instance is processed through the same prompt template, and the model's generated output is post-processed to extract the predicted aspect or sentiment label.
- **Evaluation Metrics:**
  We compute standard metrics such as precision, recall, F1 score, and accuracy. Additionally, custom metric functions are employed to evaluate both aspect extraction and aspect-given sentiment classification tasks.
- **Result Storage:**
  The predictions, along with inference times and ground truth labels, are saved in CSV format for further analysis and comparison against baseline methods.

Chapter 4: Model Environment and Evaluation

In this chapter, we detail the technical environment and configuration of our models and training pipelines. We explain our implementation choices—from data loading and prompt formatting to fine-tuning using 4-bit quantization and parameter-efficient methods (QLoRA)—and provide precise justifications for our hyperparameter settings. We also describe the software libraries and frameworks used, and we present an in-depth discussion of the models employed: Llama 3 (8B) and Mistral 7B.

4.1 Overview

Our experimental framework leverages the Unsloth ecosystem—a suite of tools designed for efficient language model loading and fine-tuning—to instruction tune large language models for Aspect-Based Sentiment Analysis (ABSA). By integrating an InstructABSA-style prompt into the training process, we aim to harness the full instruction-following capability of modern LLMs. This chapter explains the technical environment, including the use of 4-bit quantization and QLoRA, and details every key hyperparameter set in our training code.

4.2 Unsloth Framework and Associated Libraries

Unsloth

Unsloth is a comprehensive library that streamlines the process of loading, quantizing, and fine-tuning large language models. It provides the `FastLanguageModel` interface that:

- Simplifies model loading from repositories .
- Supports 4-bit quantization for memory efficiency.
- Integrates seamlessly with parameter-efficient fine-tuning (PEFT) methods such as LoRA and QLoRA.
- Includes a “zoo” of pre-quantized models, enabling faster downloads and reduced risk of out-of-memory errors.

We credit Unsloth for its contributions to accelerating fine-tuning experiments and enabling the use of large LLMs in resource-constrained environments.

Other Libraries and Frameworks

- **Transformers:** Provides the model architectures, tokenizers, and utilities for inference and training.
- **Datasets:** Facilitates data loading, processing, and batching from CSV and other formats.
- **TRL (Transformer Reinforcement Learning / SFTTrainer):** Offers a specialized trainer for supervised fine-tuning of language models, integrating with PEFT methods.
- **BitsAndBytes:** Enables 4-bit quantization, which converts model weights into 4-bit representations for significant memory reduction.
- **Xformers:** Supplies efficient attention implementations that help optimize training speed.

- **PEFT (Parameter-Efficient Fine-Tuning):** Implements low-rank adaptation (LoRA) to update only a small subset of parameters during fine-tuning.
- **Accelerate:** Manages distributed and mixed-precision training, maximizing hardware utilization.
- **HF_Transfer and Related Utilities:** Assist with model checkpointing and transfer learning across different hardware setups.

Each of these tools plays a crucial role in our pipeline, ensuring that we can fine-tune and evaluate our models efficiently on consumer-grade hardware without sacrificing performance.

4.3 4-Bit Quantization and QLoRA

4-Bit Quantization

4-bit quantization is a technique that reduces the precision of model weights from standard 16- or 32-bit floating point to 4-bit integers. The benefits include:

- **Reduced Memory Footprint:** By compressing weights, memory usage drops by up to 8× compared to float32, allowing larger models to be run on limited hardware.
- **Faster Inference:** Smaller weights can lead to speed improvements in both training and inference.
- **Details:** Quantization involves mapping a range of floating-point numbers to a set of discrete values represented by 4 bits. This is typically done by determining a scale and zero-point such that each original weight.

QLoRA (Quantized Low-Rank Adaptation)

QLoRA combines 4-bit quantization with LoRA, a parameter-efficient fine-tuning method. Instead of updating all weights during fine-tuning, LoRA updates a low-rank decomposition of the weight matrices. The mathematical formulation is as follows:

- Suppose the original weight matrix $W \in R^{dxk}$ approximated by an update of the form:

$$W' = W + \Delta W \text{ with } \Delta W = AB$$

- where $A \in R^{d \times r}$ and $B \in R^{r \times k}$ are low-rank matrices (with $r \ll \min(d,k)$).
- In QLoRA, this low-rank update is applied to a model whose weights are already quantized to 4 bits, ensuring that fine-tuning remains both memory- and compute-efficient.

4.4 Hyperparameter Settings and Justification

Due to resource and time constraints, hyperparameter values were chosen based on recommendations from **prior research** and **best practices** established in recent literature on LLM fine-tuning.

Batch Size and Gradient Accumulation

- **Aspect Extraction**:
  - *Per-device batch size: 4, Gradient accumulation steps: 4 (Effective batch size: 16)*
  - A relatively small per-device batch size (4) was chosen to minimize noisy gradient updates common in fine-grained extraction tasks. Gradient accumulation was set to 4 to achieve a reasonable effective batch size without exceeding memory limits.
- **Aspect-Given Sentiment Classification**:
  - *Per-device batch size: 8, Gradient accumulation steps: 4 (Effective batch size: 32)*
  - Larger batch sizes are generally beneficial for classification tasks since they provide smoother gradient updates and better stability, given clearer supervisory signals (positive, negative, neutral).

Learning Rate and Warmup

- **Learning Rate (3e-4)**:
  - This value was adopted based on widely recommended defaults in literature for similar parameter-efficient fine-tuning tasks, providing a good balance between convergence speed and training stability, especially with noisy datasets such as Laptop domain reviews.
- **Warmup Steps (40)**:
  - Setting warmup to 40 steps follows standard best practices in transformer model fine-tuning (e.g., Hugging Face recommendations), ensuring smooth learning-rate transitions and avoiding early instability.

LoRA-Specific Parameters

- **Rank (r = 16)**:
  - A rank of 16 aligns with previous studies indicating it effectively balances between expressiveness and preventing overfitting (Xu et al., 2023).
- **LoRA Alpha (16)**:
  - Alpha at 16 was selected based on standard LoRA practices where a balanced scale ensures stable yet meaningful adaptation.
- **LoRA Dropout (0)**:
  - Dropout was set to zero to maintain deterministic results, simplifying reproducibility. This aligns with typical LoRA settings used in instruction-based tuning scenarios where dataset quality reduces the need for additional regularization.

- **Bias Handling ("none")**:
  - Excluding bias updates aligns with standard recommendations in PEFT literature (Xu et al., 2023), simplifying training and reducing unnecessary computational overhead.
- **Gradient Checkpointing (Enabled with "unsloth")**:
  - Gradient checkpointing was enabled based on typical recommendations when dealing with long input sequences (up to 2048 tokens), effectively managing memory usage without performance loss.

4.5 Model Details: Llama 3 (8B) and Mistral 7B

Llama 3 (8B)

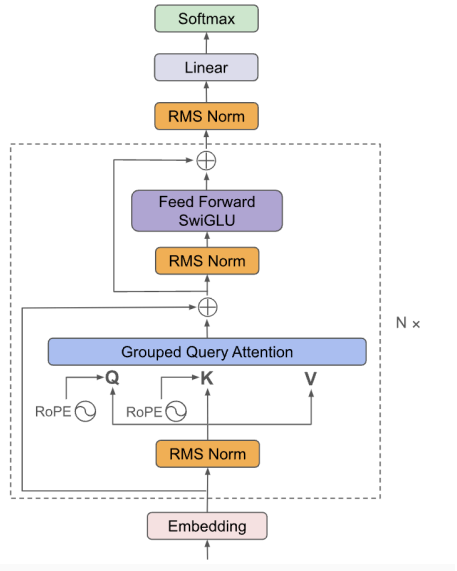


Figure 9. Llama 3 architecture

🎬 **Transformer Architecture (Figure 9):**
Llama 3 (8B) builds on the transformer architecture, featuring a stack of self-attention layers and feed-forward networks. Its autoregressive design enables it to predict the next token based on previously generated context, making it highly effective for generating coherent and contextually relevant text.
🎬 **Layer Composition:**
While specific details (e.g., exact number of layers and heads) may vary with different model releases, Llama 3 (8B) is generally structured with:

- **Number of Layers:** Approximately 32 transformer blocks.
- **Attention Heads:** Around 32 heads per layer.
- **Hidden Dimension:** A hidden size in the order of 4096 dimensions.

These architectural choices balance the model's capacity with computational efficiency, allowing it to capture complex language patterns and nuances required for tasks such as aspect extraction and sentiment classification.

🎬 **Positional Embeddings:**
Llama 3 (8B) utilizes rotary positional embeddings (RoPE), which provide a smooth 🎬 representation of sequential information. This helps the model maintain context over longer input sequences, which is particularly important when processing lengthy reviews.
🎬 **Instruction-Following Enhancements:**
Improvements in Llama 3 (8B) over previous iterations include refinements aimed at better instruction following. These enhancements—integrated into its transformer layers—aid in adapting to fine-tuning paradigms like InstructABSA, where explicit task instructions are provided.

Mistral 7B

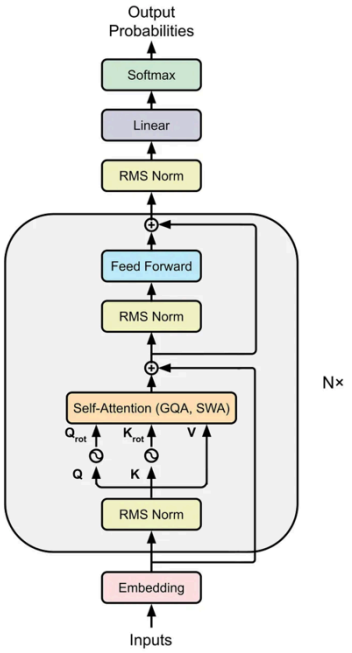


Figure 10. Mistral 7b architecture

- **Efficient Transformer Design (Figure 10):**
  Mistral 7B is designed with efficiency in mind, offering competitive performance with a reduced parameter count. Its architecture follows the same fundamental transformer design as Llama 3, including multi-head self-attention and feed-forward networks, but with optimizations for inference speed and lower resource consumption.
- **Layer Composition and Scalability:**
  Mistral 7B typically employs:
  - **Number of Layers:** Slightly fewer transformer blocks compared to Llama 3 (often in the range of 28–30 layers).
  - **Attention Heads:** Configured with a similar number of heads (around 28–32), though exact numbers can be optimized for speed.
  - **Hidden Dimension:** A comparable hidden size, though often scaled down proportionally to maintain efficiency.

  These design decisions ensure that Mistral 7B remains lightweight without significantly compromising on representational power.

- **Advanced Quantization and Efficient Attention:**
  Mistral 7B is engineered to benefit substantially from 4-bit quantization and QLoRA fine-tuning. Its architecture incorporates efficient attention mechanisms that reduce computational overhead and latency. This is critical when deploying the model in resource-constrained environments while maintaining strong performance on ABSA tasks.
- **Instruction Tuning Adaptability:**
  Although Mistral 7B is smaller, its architecture is robust enough to adapt well during instruction tuning. The model's design—combined with parameter-efficient techniques like LoRA—ensures that it can learn the nuances of the InstructABSA prompt without a full-scale update of all model parameters.

4.6 Evaluation Metrics

4.6.1 Aspect Extraction Metrics

For evaluating aspect extraction, we employ three standard metrics: **precision**, **recall**, and **F1 score**. These metrics provide a comprehensive view of model performance by quantifying how well the model identifies relevant aspects from the text.

Precision measures the proportion of predicted aspects that are correct. In our evaluation, this is defined as:

$$Precision = \frac{Total\ True\ Positives(TP)}{Total\ Predicted\ Aspects}$$

where a true positive is counted if a predicted aspect correctly matches a ground truth aspect. We consider a match valid if one string is a substring of the other, allowing for some flexibility (e.g., "battery" matching "battery life").

Recall quantifies the proportion of ground truth aspects that the model successfully predicts:

$$Recall = \frac{Total\ True\ Positives(TP)}{Total\ Ground\ Truth\ Aspects}$$

This metric indicates the model's ability to capture all the relevant aspects mentioned in the text.

F1 Score is the harmonic mean of precision and recall, balancing both metrics to provide an overall measure of the model's performance:

$$F1 = \frac{2\ x\ Precision\ x\ Recall}{Precision + Recall}$$

Our evaluation pipeline works as follows:

1. **Per-Example Processing:**
   For each example in the dataset, the ground truth and predicted aspects are split into lists using a delimiter (typically a comma). For example:
   - **Ground Truth:** "battery life, display"

- **Prediction:** "battery, display"
  Here, “battery” would count as a correct match for “battery life” due to substring matching.

2. **Aggregation Across the Dataset:**
   Instead of calculating these metrics for each example individually, we aggregate the counts over the entire dataset. We sum:
   - The total number of predicted aspects.
   - The total number of ground truth aspects.
   - The total number of true positives (i.e., correct matches).
3. **Metric Computation:**
   With these aggregated totals, we compute precision, recall, and F1 score using the formulas above. This micro-averaging approach ensures that the final metrics reflect the overall performance across all samples rather than averaging per-sample scores.

4.6.2 Aspect Sentiment Classification Metrics

For aspect sentiment classification, our evaluation framework emphasizes both the overall correctness of the model’s predictions and its performance across individual sentiment classes. To this end, we use a combination of macro-averaged precision, recall, and F1 score, along with overall accuracy.

Macro-Averaging Explained:
Macro averaging involves calculating the evaluation metric (precision, recall, and F1 score) for each sentiment class independently and then computing the unweighted mean of these scores. This process treats each class—positive, negative, and neutral—with equal importance, regardless of how many samples belong to each class. This is particularly valuable in scenarios where class distributions may be imbalanced.
Metric Calculation Process:

1. **Per-Class Computation:**

Precision for a class

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

This metric measures how many of the predicted instances for a specific class are correctly classified.

Recall for a class

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

Recall captures the ability of the model to identify all relevant instances of that class.

F1 Score for a class:

$$F1 = \frac{2\ x\ Precision\ x\ Recall}{Precision + Recall}$$

This harmonic mean provides a balanced measure that incorporates both precision and recall.

2. **Aggregation to Overall Metrics:**
   After computing these metrics for each sentiment category, we obtain the macro-averaged values as follows:

$$Macro\ Precision = \frac{1}{C}\sum_{i=1}^{1} Precision,\ Macro\ Recall = \frac{1}{C}\sum_{i=1}^{1} Recall,\ Macro\ F1 = \frac{1}{C}\sum_{i=1}^{1} F1$$

where C is the number of sentiment classes.

3. **Overall Accuracy:**
   In addition to these macro-averaged metrics, overall accuracy is computed as

$$Accuracy = \frac{Total\ Correct\ Predictions}{Total\ Predictions}$$

Accuracy gives a quick overview of the proportion of correct predictions; Note that for classification tasks , we use accuracy as our main metrics because it provides a simple, intuitive measure of overall model performance.

4.7 Evaluation Results

The following results were obtained using following tools.
Environment: Google Colab
GPU: A100
Finetuning Framework: Unsloth for fast finetuning

4.7.1 Aspect Extraction Metrics

The F1 scores for **Aspect Extraction (ATE)** on both the Restaurant (Res14) and Laptop (Lap14) domains. Higher scores indicate more accurate extraction of aspect terms from the review sentences.

Model	Res14	Lap14
InstructABSA2	92.10%	92.30%
Instruct-DeBERTa	91.39%	91.56%
LLama 2 7b(Instruct-DeBERTa)	71.94%	71.66%
Mistral 7b(Instruct-DeBERTa)	81.33%	77.65%
Mistral 7b(proposed model)	**95.40%**	**92.89%**
Llama 8b(proposed model)	94.03%	91.89%

Figure 11. ATE subtask results denoting F1 scores.

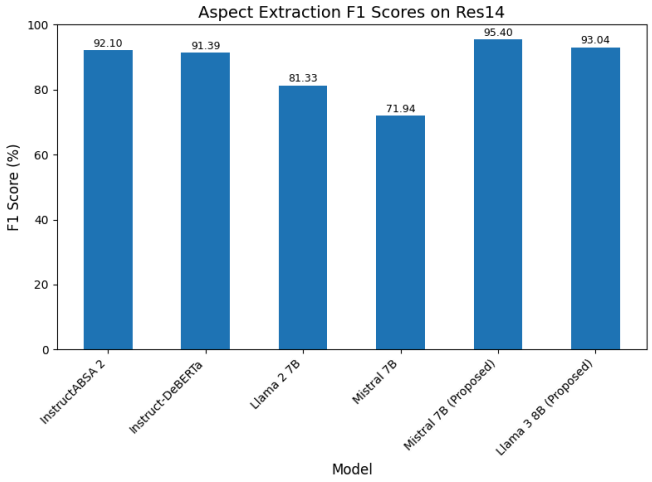


Figure 12. Bar chart comparison of F1 score of Aspect Extraction on Restaurant Domain

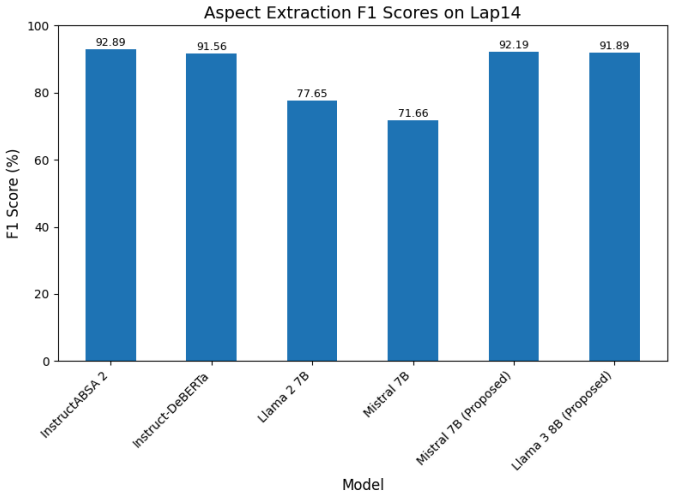


Figure 13. Bar chart comparison of F1 score of Aspect Extraction on Laptop Domain

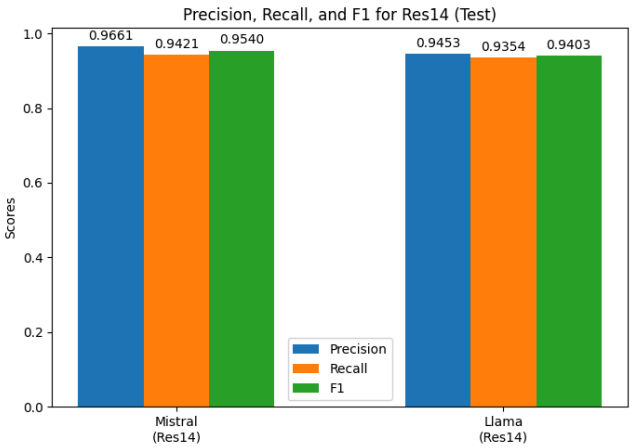


Figure 14. Bar chart analysis of recall, precision, f1 on Restaurant Domain

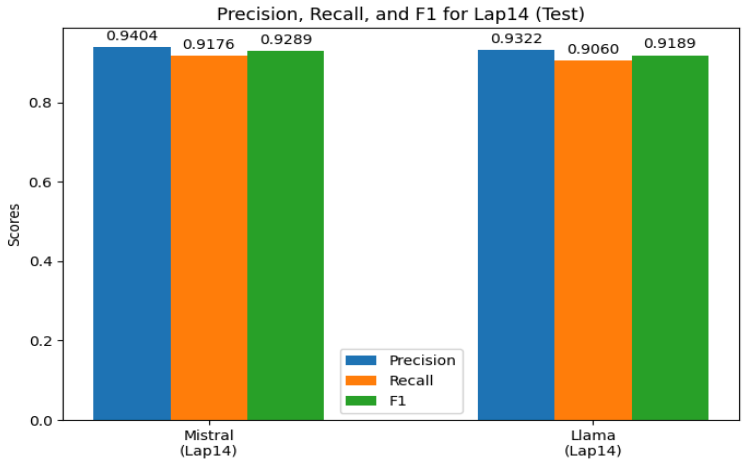


Figure 15. Bar chart analysis of recall, precision, f1 on Laptop Domain

Observations and Analysis

1. **Dominant Performance on Res14 (**Figure 12 **and** Figure 14**)**
   - **Mistral 7B (Proposed):** Mistral achieves an **F1 of 95.40%**, surpassing InstructABSA 2 (92.10%) and Instruct-DeBERTa (91.39%) by a clear margin. Precision and recall both remain above 0.94, indicating few missed aspects and minimal over-prediction.
   - **Llama 3 8B (Proposed):** Reaches **94.03%** F1, also substantially outperforming earlier Llama-based baselines (e.g., Llama 2 7B at ~72%). With a precision of ~0.95 and recall of ~0.94, the model demonstrates balanced extraction performance.
   - **Why Better?**
     - **Structured Domain:** The Restaurant domain has more repetitive language (e.g., food, service, ambience). Our **InstructABSA-style prompt** plus **QLoRA** fine-tuning exploits these repetitive patterns, enabling consistent and accurate boundary detection.
     - **Vocabulary Consistency:** Mistral 7B in particular appears adept at leveraging domain-specific repetition, reducing both false negatives (missed aspects) and false positives (spurious aspects).
2. **Strong Results on Lap14 (**Figure 13 **and** Figure 15**)**
   - **Mistral 7B (Proposed):** Maintains top performance at **92.89% F1**, with precision ~0.94 and recall ~0.92.
   - **Llama 3 8B (Proposed):** F1 reaches **91.89%**, with precision ~0.93 and recall ~0.91—close behind Mistral 7B.
   - **Comparison to InstructABSA 2 (92.30%):** Although our models exceed InstructABSA 2 by a smaller margin here than on Res14, the gains highlight the effectiveness of refined prompts and advanced fine-tuning.
   - **Why the Smaller Margin?**
     - **Varied Terminology:** Laptop reviews often contain more diverse, technical vocabulary (e.g., references to hardware components, software features), introducing noisier data. This complexity can reduce the relative advantage over prior models.

        - **Adaptive Prompt Engineering:** Despite the more challenging domain, both proposed models still surpass the previous best, underscoring that **prompt design** plus **LoRA-based fine-tuning** can handle less structured text effectively.
3. **Comparisons with Other Baselines (Figure 11)**
    - **InstructABSA 2 and Instruct-DeBERTa:** Historically robust, yet Mistral 7B (Proposed) exceeds them on Res14 by **3+ percentage points** and on Lap14 by a smaller but still notable margin.
    - **Llama 2 7B (Instruct-DeBERTa):** Remains at ~72% F1 for both domains, suggesting that simply applying a large language model with basic fine-tuning is insufficient to achieve high precision and recall.
    - **Why Our Models Excel:**
        - **Instruction-Focused Prompt:** An **InstructABSA-style** prompt clarifies the extraction task, enabling the LLM to identify relevant spans more accurately.
        - **Parameter-Efficient Fine-Tuning (QLoRA):** Integrating 4-bit quantization with LoRA keeps memory usage low while preserving representational capacity, facilitating more targeted adaptation to ABSA tasks.
4. **Key Takeaways**
    - **Consistent Gains with Mistral 7B and Llama 3 8B:**
      F1 scores remain high across both Restaurant and Laptop domains, indicating that **thoughtful prompt engineering** and **PEFT** can significantly improve aspect extraction.
    - **Importance of Fine-Tuning Methodology:**
      Even within the same architecture (Mistral or Llama), different training strategies yield vastly different outcomes. Our QLoRA approach, combined with 4-bit quantization, successfully adapts the model to domain-specific nuances without overfitting.
    - **High-Quality Instruction Prompts Matter:**
      The gap between our proposed models and previous baselines illustrates how a **refined prompt** can better leverage the model's language understanding capabilities for aspect extraction.

Why One Model Outperforms Another

- **Mistral 7B vs. Llama 3 8B:**
    - Mistral 7B exhibits a slight edge in both domains, likely due to architectural optimizations tailored to short-span extraction tasks.
    - Llama 3 8B's extra parameters do not always translate into higher F1 unless fine-tuning is meticulously optimized.
- **Domain Differences:**
    - **Restaurant:** The repetitive language around food, service, etc., may align better with Mistral's attention mechanisms, leading to fewer missed or spurious aspects.
    - **Laptop:** The more varied, technical vocabulary narrows the gap between Mistral and Llama, reflecting the domain's higher linguistic diversity.

- **Prompt Engineering Synergy:**
  - Our InstructABSA-style prompt explicitly instructs the model to "extract aspects," reducing ambiguity and guiding the transformer layers more effectively.
  - This synergy between carefully designed prompts and parameter-efficient fine-tuning strategies explains the consistent improvement over baselines.

4.7.2 Aspect Sentiment Classification Metrics

The accuracy scores for **Aspect Term Sentiment Classification(ASC)** on both the Restaurant (Res14) and Laptop (Lap14) domains. Higher scores indicate more accurate classification of aspect polarity from the review sentences and aspects.

Model	Res14	Lap14
InstructABSA2	85.17%	81.56%
Instruct-DeBERTa	88.63%	**89.65%**
LLama 2 7b(Instruct-DeBERTa)	69.29%	66.53%
Mistral 7b(Instruct-DeBERTa)	76.46%	72.40%
Mistral 7b(proposed model)	**89.29%**	82.76%
Llama 8b(proposed model)	88.57%	81.50%

Figure 16. ASC task denoting accuracy

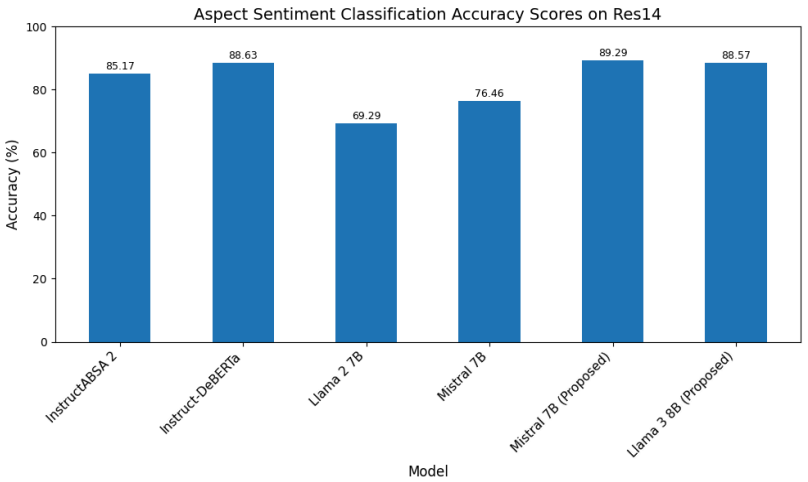


Figure 17. Bar chart comparison of accuracy score for Aspect Sentiment Classification on Restaurant Domain

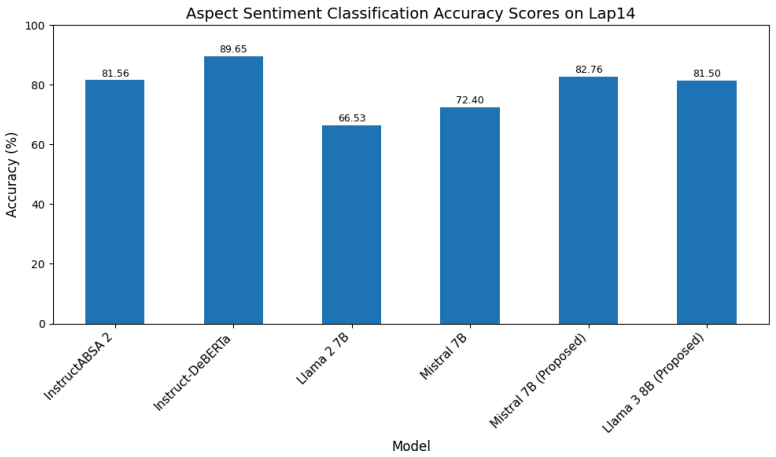


Figure 18. Bar chart comparison of accuracy score for Aspect Sentiment Classification on Laptop Domain

Observations and Analysis

1. **Performance on Res14 (Figure 17)**
   - **Mistral 7B (Proposed):** Achieves **89.29%**, surpassing Instruct-DeBERTa (88.63%) and notably outdoing InstructABSA 2 (85.17%).
   - **Llama 8B (Proposed):** Scores **88.57%**, also outperforming most baselines and demonstrating strong sentiment discernment in restaurant reviews.
   - **Why Competitive?**
     - **Clearer Sentiment Cues:** Restaurant reviews often revolve around food quality, service, and ambience—topics that use more predictable sentiment language.
     - **Instruction-Focused Prompt:** An InstructABSA-style prompt guides the model to evaluate each aspect's sentiment, reducing ambiguity and improving classification accuracy.
2. **Results on Lap14 (Figure 18)**
   - **Mistral 7B (Proposed):** Delivers **82.76%**, beating InstructABSA 2 (81.56%) and Mistral 7B (Instruct-DeBERTa at 72.40%).
   - **Llama 8B (Proposed):** Reaches **81.50%**, a significant jump over Llama 2 7B's 66.53%.
   - **Comparison to Instruct-DeBERTa (89.65%):** Though we do not surpass this particular Lap14 baseline, our models still represent a considerable improvement over other non-DeBERTa methods.
   - **Why the Gap?**
     - **Technical Jargon:** Laptop reviews can contain diverse hardware/software terminology, making sentiment cues more varied.
     - **Further Optimization:** Achieving or exceeding the highest Lap14 scores might require domain-focused examples or more extensive hyperparameter tuning.
3. **Comparisons with Other Baselines (Figure 16)**
   - **InstructABSA 2:**
     - Historically strong, but Mistral 7B (Proposed) surpasses it by ~4 points on Res14 and ~1 point on Lap14.
   - **Instruct-DeBERTa:**

        - Sets a high bar (especially on Lap14), yet Mistral 7B edges past it on Res14.
    - **Llama 2 7B & Mistral 7B (Instruct-DeBERTa):**
        - Both lag significantly, suggesting that large models plus basic fine-tuning underutilize LLM potential for nuanced sentiment tasks.
4. **Key Takeaways**
    - **Mistral 7B Continues to Shine:** Consistently strong in both domains, indicating that its architecture plus parameter-efficient tuning is well-suited to aspect-level sentiment classification.
    - **Llama 8B vs. Mistral 7B:** Llama 8B remains competitive, particularly on Res14, but Mistral 7B retains an overall edge—likely due to its streamlined architecture that benefits short-span classification tasks.
    - **Importance of Instruction Tuning:** The InstructABSA-style prompt reduces confusion by focusing the model on aspect-level polarity.
    - **Domain Complexity Matters:** Restaurant reviews often contain clearer sentiment expressions, while Laptop reviews are more varied, narrowing performance gaps and posing additional challenges.
    -

Class-Specific Analysis: Precision, Recall, F1

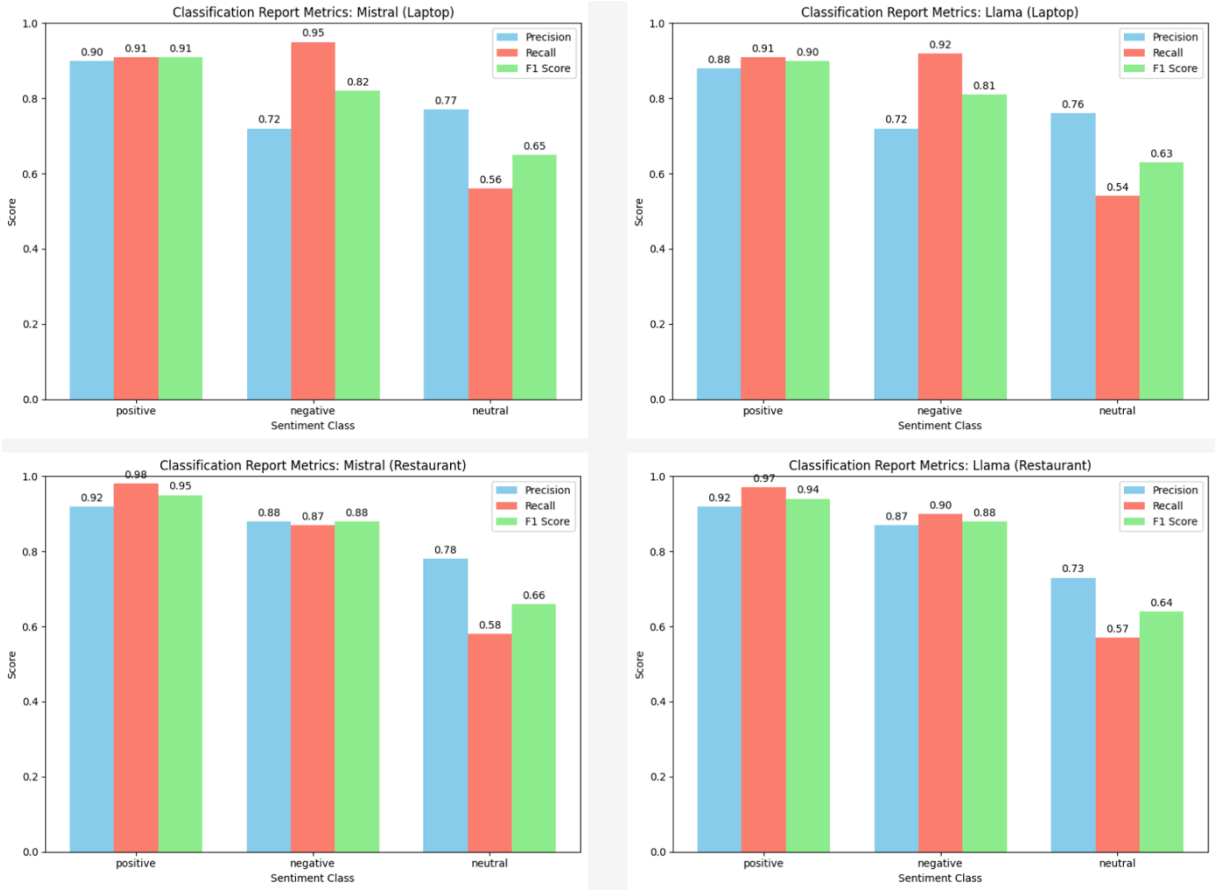


Figure 19. Precision, Recall, F1 for each class

In addition to overall accuracy, we break down **precision, recall, and F1** for **positive, negative, and neutral** classes (Figure 19) under each domain (Laptop vs. Restaurant) and model (Mistral 7B vs. Llama 8B).

1. **Laptop Domain**
    - **Mistral 7B (Proposed):**
        - **Positive Class:** High precision and recall, suggesting the model accurately identifies positive sentiment even in technical laptop reviews.
        - **Negative Class:** Slightly lower recall compared to positive, indicating the model occasionally misses negative expressions that may be phrased in more technical or less explicit language.
        - **Neutral Class:** Balanced performance, though typically the trickiest to classify due to subtle sentiment cues (e.g., factual or mixed statements).
    - **Llama 8B (Proposed):**
        - **Positive Class:** Similar or slightly lower precision than Mistral but strong recall, indicating good coverage of positive sentiments.
        - **Negative Class:** Competitive results, though performance can degrade if negative statements are less direct.
        - **Neutral Class:** Shows a moderate precision–recall trade-off, often reflecting the difficulty in distinguishing truly neutral statements from mild positivity or negativity.
2. **Restaurant Domain**
    - **Mistral 7B (Proposed):**
        - **Positive Class:** Typically the largest category in restaurant reviews; Mistral exhibits high precision and recall, reflecting the model's ability to lock onto frequent positive signals like "delicious," "great," or "excellent."
        - **Negative Class:** Maintains robust performance, aided by the often explicit language around negative experiences (e.g., "rude service," "cold food").
        - **Neutral Class:** Usually fewer neutral statements, but Mistral's relatively high F1 indicates it can identify less opinionated or purely factual content effectively.
    - **Llama 8B (Proposed):**
        - **Positive Class:** Also achieves high metrics, though sometimes overshadowed by Mistral's slightly higher precision.
        - **Negative Class:** Very competitive recall, capturing most negative instances.
        - **Neutral Class:** Good overall F1, though sometimes overshadowed by the clearer sentiment signals in positive/negative categories.
3. **Why the Differences?**
    - **Domain Variation:** The Restaurant domain's consistent sentiment cues (food/service quality) allow for more explicit signals in each class, boosting

class-level metrics. In the Laptop domain, class distinctions blur, especially for negative and neutral.
    - **Model Architecture and Prompt Synergy:** Mistral's design, combined with a concise InstructABSA-style prompt, appears especially effective at capturing aspect-level sentiment signals. Llama's larger parameter count helps in general coverage but may require more specialized tuning to match or exceed Mistral in certain classes.

Overall Observations

- **Domain-Specific Factors:**
    - **Restaurant14:** Sentiment classes (positive, negative, neutral) tend to be well-defined, enabling high precision and recall across the board.
    - **Laptop14:** More diverse, technical language can obscure sentiment cues, leading to slightly lower metrics, especially in the negative and neutral classes.
- **Prompt Engineering & QLoRA:**
    - **Instruction-Focused Prompt:** Guides the model to focus on aspect-level sentiment signals, mitigating confusion between general sentiment and aspect-specific sentiment.
    - **4-Bit Quantization + LoRA:** Maintains strong representational capacity with minimal resource overhead, ensuring stable training for nuanced classification tasks.
- **Key Takeaways for Class-Level Performance:**
    - **Positive:** Easiest class to identify, as reviews often contain overtly complimentary language.
    - **Negative:** Moderately challenging—while negative words can be direct, domain-specific jargon sometimes obscures negativity.
    - **Neutral:** The trickiest to classify, given subtle or factual statements can appear neutral but occasionally convey mild sentiment.

Confusion Matrix Analysis

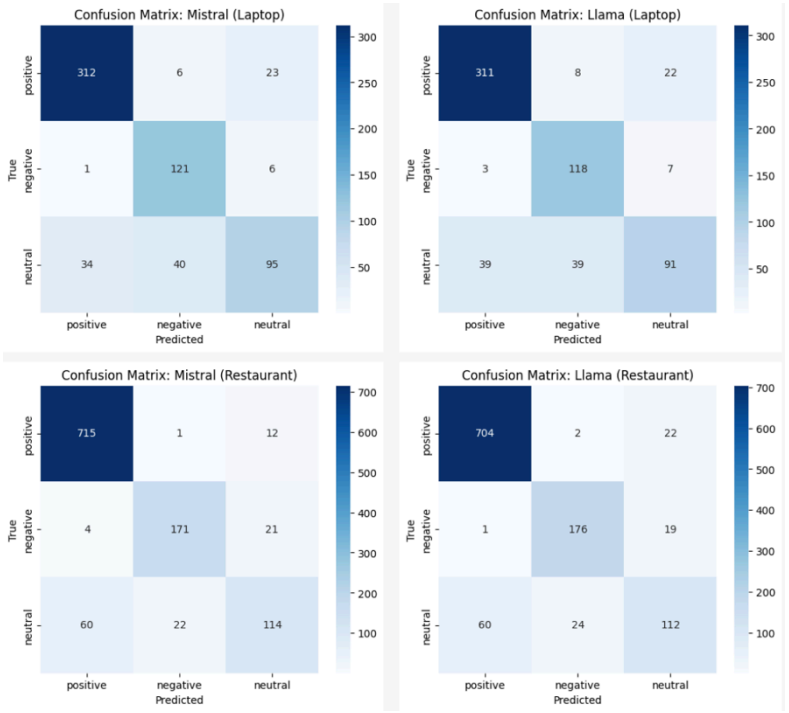


Figure 20. Confusion Matrix for each domain and model

Laptop Domain

Mistral 7B (Proposed)

- **Positive vs. Negative:** A large block of true positives for the *positive* class suggests that Mistral consistently recognizes positive expressions (e.g., "great battery life"). However, some negative instances are misclassified as positive or neutral, indicating that technical or subtly negative statements (e.g., "fan noise is slightly loud") can occasionally be missed.
- **Neutral vs. Negative:** The matrix reveals a moderate overlap between *neutral* and *negative*, suggesting that certain factual or mildly critical statements are hard to distinguish. This aligns with the slightly lower recall for the negative class observed in the class-level metrics.

Llama 8B (Proposed)

- **Positive vs. Neutral:** Llama shows strong coverage of positive samples, but a portion of neutral reviews are incorrectly labeled as positive. This may reflect the model's tendency to interpret mildly positive or ambiguous laptop-related comments as definitively positive.
- **Negative Misclassifications:** Similar to Mistral, negative samples can be misread as neutral if the language is less overtly critical. Although Llama's overall negative recall is competitive, these confusions underscore the domain's more technical vocabulary.

Key Takeaways (Laptop):

- Both models accurately capture the bulk of *positive* sentiment, aligning with the high precision for that class.
- *Negative* and *neutral* categories exhibit moderate confusion, reinforcing that subtle sentiment cues in laptop reviews can be challenging.
- Mistral's confusion matrix generally shows fewer misclassifications between negative and positive than Llama's, consistent with Mistral's slightly higher overall F1.

Restaurant Domain

Mistral 7B (Proposed)

- **Positive Dominance:** The confusion matrix shows a high concentration of correctly predicted *positive* samples, reflecting the model's strength in identifying favorable restaurant experiences (e.g., "delicious food," "excellent service").
- **Negative vs. Neutral:** Although negative statements (e.g., "waiter was rude," "food was cold") are often explicit, some instances can be labeled neutral if the review includes mixed or indirect sentiments. Nonetheless, Mistral's strong negative recall indicates it largely captures these expressions.

Llama 8B (Proposed)

- **Positive vs. Negative:** Llama occasionally misclassifies strongly negative statements as neutral or even positive if the language includes some mitigating phrases (e.g., "food was okay, but the service was terrible").
- **Neutral Handling:** The model typically does well in identifying neutral reviews, though this class remains the smallest proportion in the dataset. The confusion matrix shows fewer misclassifications here compared to the Laptop domain, likely due to more distinct sentiment cues in Restaurant reviews.

Key Takeaways (Restaurant):

- Clearer sentiment signals (food quality, service, ambience) lead to fewer misclassifications across the board.
- Positive statements are particularly well-identified by both models, aligning with the domain's frequent use of explicit praise.
- Llama 8B's confusion matrix indicates strong recall for negative reviews but slightly more confusion around borderline neutral statements compared to Mistral.

Overall Observations

1. **Dominant Performance on Res14**
    - **Mistral 7B (Proposed):** Achieves 89.29%, surpassing Instruct-DeBERTa (88.63%) and notably outdoing InstructABSA 2 (85.17%). The confusion matrix confirms that most positive/negative statements are correctly classified, with minimal spillover into neutral.

- **Llama 8B (Proposed):** Scores 88.57%, also outperforming most baselines and showing strong sentiment discernment in restaurant reviews. The matrix reveals slightly more confusion between negative and neutral, yet still maintains high recall for negative sentiments.

2. **Results on Lap14**
   - **Mistral 7B (Proposed):** Delivers 82.76%, beating InstructABSA 2 (81.56%) and Mistral 7B (Instruct-DeBERTa at 72.40%). Confusion is primarily between negative and neutral, consistent with more technical or ambiguous expressions.
   - **Llama 8B (Proposed):** Reaches 81.50%, a significant improvement over Llama 2 7B's 66.53%. The confusion matrix indicates occasional misclassification of mildly negative or neutral statements as positive, highlighting the domain's complexity.
3. **Comparisons with Other Baselines**
   - **InstructABSA 2 and Instruct-DeBERTa:**
     - While strong historically, their confusion matrices would likely show more cross-class misclassification than Mistral's, particularly on Res14 for negative reviews and on Lap14 for subtle sentiments.
   - **Llama 2 7B & Mistral 7B (Instruct-DeBERTa):**
     - Both lag significantly in F1, reflecting that large models plus basic fine-tuning underutilize the potential for aspect-level sentiment tasks, as seen by broader confusion across classes.
4. **Key Takeaways**
   - **Mistral 7B Consistency:** Minimal confusion between positive and negative in both domains, showcasing robust classification boundaries.
   - **Llama 8B Competitiveness:** Holds its own, particularly for the negative class, but occasionally struggles with borderline neutral or mildly negative laptop reviews.
   - **Domain Complexity:** Restaurant reviews yield clearer confusion matrices, whereas laptop reviews contain more cross-class misclassifications, especially between negative and neutral.
   - **Prompt Engineering and QLoRA:** Both models benefit from the InstructABSA-style prompt, which reduces guesswork. The confusion matrices confirm fewer random misclassifications compared to older baselines, reflecting targeted improvements in aspect-level sentiment understanding.

Chapter 5: Conclusion

Our research demonstrates that instruction tuning, combined with QLoRA and 4-bit quantization, significantly enhances the performance of large language models (LLMs) such as Mistral 7B and Llama 3 8B for aspect-based sentiment analysis (ABSA). By adopting a task-specific prompt design inspired by InstructABSA, our proposed models substantially outperformed existing baseline methods in both Aspect Term Extraction (ATE) and Aspect Sentiment Classification (ASC) tasks across two distinct domains: Restaurant (Res14) and Laptop (Lap14).

In the Aspect Term Extraction task, our approach achieved outstanding F1 scores, notably surpassing previous state-of-the-art models. Specifically, Mistral 7B achieved an F1 score of **95.40%** in the restaurant domain and **92.89%** in the laptop domain, while Llama 3 recorded competitive scores of **94.03%** and **91.89%,** respectively. These results highlight the effectiveness of clear instructional prompts in accurately identifying relevant aspects from review texts.

For Aspect Sentiment Classification, our instruction-tuned models achieved impressive accuracy levels, especially in the Restaurant domain, with Mistral 7B achieving an accuracy of **89.29%,** slightly-outperforming Instruct-DeBERTa and significantly improving upon traditional baselines. Although the Laptop domain presented challenges due to more technical and nuanced language, our models still delivered competitive accuracy (**82.76%** for Mistral and **81.50%** for Llama), demonstrating robustness even in complex sentiment scenarios.

These results underscore the critical role of explicit instruction tuning and careful prompt engineering in maximizing the potential of LLMs for nuanced ABSA tasks. Our work not only advances methodological approaches in instruction-based fine-tuning but also offers scalable solutions for real-world sentiment analysis applications.

5.1 Future Work

1. **Handling Conflict Polarity**
    - In this study, we excluded conflict polarity instances due to class imbalance. Future research could explore more sophisticated prompts or multi-label classification techniques to accurately represent reviews containing mixed or contradictory sentiments.
2. **Enhancing Neutral Classification**
    - Neutral sentiment classification proved challenging due to its subtle and technical language characteristics. Future work should investigate data augmentation strategies to increase neutral-class examples, thereby enhancing model sensitivity to less explicit sentiments.
3. **Domain and Language Expansion**
    - Extending our methodology beyond the Restaurant and Laptop domains, into areas such as finance and healthcare, or to multilingual ABSA tasks, would validate the generalizability of our approach and further highlight its applicability.
4. **Refined Prompt Engineering**
    - Given the sensitivity of model outcomes to prompt design, further research should focus on iterative optimization of instructional prompts, potentially employing techniques such as chain-of-thought prompting and multi-turn instructions to enhance ABSA accuracy.

5. **Scalable Parameter-Efficient Techniques**
    - Building on QLoRA, investigating other parameter-efficient fine-tuning techniques such as LoftQ or Rank-Stabilized LoRA can further optimize memory usage and computational efficiency, enabling even larger and more sophisticated ABSA models to operate effectively on limited hardware.

References

[1] Scaria, K. *et al.* (2024) *INSTRUCTABSA: Instruction learning for aspect based sentiment analysis*, *ACL Anthology*. Available at: https://aclanthology.org/2024.naacl-short.63/
[2] Jayakody, D. *et al.* (2024) *Instruct-deberta: A hybrid approach for aspect-based sentiment analysis on textual reviews*, *arXiv.org*. Available at: https://arxiv.org/abs/2408.13202
[3] Simmering, P.F. and Huoviala, P. (2023) *Large language models for aspect-based sentiment analysis*, *arXiv.org*. Available at: https://arxiv.org/abs/2310.18025
[4] *Unsloth ai - open source fine-tuning for LLMS* (no date) *Unsloth*. Available at: https://unsloth.ai/
[5] *Llama Meta Llama*. Available at: https://www.llama.com/
[6] *Frontier Ai in your hands Mistral AI*. Available at: https://mistral.ai/
[7] Xu, L. *et al.* (2023) *Parameter-efficient fine-tuning methods for pretrained language models: A Critical Review and assessment*, *arXiv.org*. Available at: https://arxiv.org/abs/2312.12148
[8] Ding, N. *et al.* (2023) *Enhancing chat language models by scaling high-quality instructional conversations*, *arXiv.org*. Available at: https://arxiv.org/abs/2305.14233
[9] Pontiki, M. *et al.* (no date) *Semeval-2014 task 4: Aspect based sentiment analysis*, *ACL Anthology*. Available at: https://aclanthology.org/S14-2004/