DimABSA2026 Project Report: Aspect-Based Sentiment Analysis with Transformers

Istrate Maria, Chichirău Claudiu-Constantin, Brassat Alexandru
Alexandru Ioan Cuza University of Iași

January 2026

Abstract

This report presents the project developed for the DimABSA2026 competition (github, codabench). The competition targets Aspect-Based Sentiment Analysis (ABSA) across three subtasks: (1) aspect extraction, (2) triplet construction, and (3) quadruplet construction with category classification. We provide a project-level problem description, a state-of-the-art overview, exploratory data analysis on the competition datasets, a presentation of the implemented approaches (including architectural diagrams), and experimental results.

1 Problem Description

Aspect-Based Sentiment Analysis (ABSA) aims to detect fine-grained sentiment signals tied to specific aspects within text. DimABSA2026 defines three subtasks:

- Subtask 1: Given a text and one or more aspects, predict a real-valued valence-arousal (VA) score for each aspect.
- Subtask 2: Given a text, extract all {Aspect, Opinion, Valence-Arousal score(VA)}. (Aspects and Opinions are phrases copied from the input
- Subtask 3: Construct quadruplets {Aspect, Category, Opinion, VA}, assigning a domain-specific `Entity#Attribute` category.

Inputs are single-review texts with IDs; outputs must follow the following format:

```
Input:
{
    "ID": "L001",
    "Text": "i am extremely happy with this laptop.",
}
Output:
{
    "ID": "L001",
    "Quadruplet":[
        {
            "Aspect": "laptop",
            "Category": "LAPTOP#GENERAL",
            "Opinion": "extremely happy",
```

```
            "VA": "8.12#8.25"
        }
    ]
}
```

Predicted Quadruplets consist of:

- Aspect - A word or phrase indicating an opinion target, such as appetizer, waiter, battery, or screen.
- Category - An abstract or predefined category to which an aspect term belongs. It follows the format Entity#Attribute.
- Opinion - A sentiment-bearing word or phrase associated with a specific aspect term, such as great, terrible, or satisfactory.
- Valence-Arousal score - A pair of real-valued scores, each ranging from 1.00 to 9.00, rounded to two decimal places.
  - Valence measures the degree of positivity or negativity
  - Arousal measures the intensity of emotion
  - A score of 1.00 indicates extremely negative valence or very low arousal, 9.00 indicates extremely positive valence or very high arousal, and 5.00 represents a neutral valence or medium arousal.

Competition submissions are graded by RMSE for subtask 1, and a modified F1 score for subtasks 2 & 3. The modified score computes True Positives, False Positives and False Negatives over the categorial attributes, and applies a distance penalty for the continuous VA score.

$$dist(VA_p, VA_g) = \frac{\sqrt{(V_p - V_g)^2 + (A_p - A_g)^2}}{D_{max}},$$

$$cRecall = \frac{TP_{cat} - \sum_{t \in P_{cat}} \text{dist}(VA_p^{(t)}, VA_g^{(t)})}{TP_{cat} + FN_{cat}},$$

$$cPrecision = \frac{TP_{cat} - \sum_{t \in P_{cat}} \text{dist}(VA_p^{(t)}, VA_g^{(t)})}{TP_{cat} + FP_{cat}},$$

$$cF1 = \frac{2 \times cRecall \times cPrecision}{cRecall + cPrecision}$$

2 State of the Art

ABSA methods evolved from traditional ML (SVM, CRF) to neural models (CNN/RNN) and, more recently, to large pre-trained transformers[1][2]. Core directions include:

- Sequence labeling for aspect/opinion extraction (e.g., BiLSTM-CRF, BERT-CRF) with BIO tagging.

Prediction/Gold	TP_cat (A)	VA error distance		cTP (A)-(C)
		Raw (B)	Normalized (C) = (B) / $\sqrt{128}$	
P: (food, good, 8.00#8.00) G: (food, good, 7.00#7.00)	1	$\sqrt{2}$	$\frac{\sqrt{2}}{\sqrt{128}}$=0.125	0.875
P: (soup, spicy, 7.50#7.50) G: (soup, spicy, 3.50#3.50)	1	$\sqrt{32}$	$\frac{\sqrt{32}}{\sqrt{128}}$=0.5	0.5
P: (staff, friendly, 7.00#7.00) G: (staff, always friendly, 7.50#7.50)	0	—	—	0
P: (staff, good, 7.00#7.00) G: N/A	0	—	—	0
			Total cTP	1.375
				*c*Recall = 1.375 / 3 = 0.458 *c*Precision = 1.375 / 4 = 0.344

Figure 1: Example of computing cF1 score

- Transformer-based fine-tuning (BERT [3], RoBERTa, DeBERTa) for classification/extraction; domain-adaptive pretraining helps.
- Joint/multi-task learning sharing representations across subtasks (aspect/opinion/category/polarity).
- Prompting and adapter modules to inject domain knowledge with fewer trainable parameters.
- Generative approaches (T5/BART) that output triplets/quadruplets directly, often with constrained decoding.
- Grid Tagging Scheme[5] can be used for Aspect-Opinion-Sentiment extraction, in order to capture relations between input words by predicting a BIO label for each pair of tokens.
- Pointer-Generator Networks[7] are sequence to sequence models, which incorporate an attention layer with a bias towards copying parts of the input exactly.

In category classification (Subtask 3), sequence classification using BERT-like architectures with auxiliary signals (aspect/opinion/VA) has proven strong. Macro-F1 is preferred for imbalanced label distributions.

3 Dataset

The datasets were provided as part of the DimABSA 2026 competition on github.

3.1 Exploratory Data Analysis

3.1.1 Laptop Dataset

This dataset has 4076 training samples and 200 test samples. The following statistics were computed over the training dataset:

- **Number of output quadruplets:** 5773
  - 1254 *NULL Aspects*

– 1583 *NULL Opinions*

(Note: *NULL* values represent aspects or opinions that are absent from the text and are not necessarily invalid.)

- **Number of Categories:** 121
- **Maximum input text length:** 485 characters
- **Maximum quads per input:** 10
  - Input text: *"if you are in need of a reliable laptop that is lightweight , fast , and convertible , i highly recommend the asus c302 !"* (121 characters)
  - Aspect/Opinion pairs: *asus c302#reliable|asus c302#highly recommend|laptop#reliable|laptop#highly recommend|asus c302#lightweight|laptop#lightweight|asus c302#fast|laptop#fast|asus c302#convertible|laptop#convertible|*

Category(1/2)	**Count**	**Category(2/2)**	**Count**
LAPTOP#GENERAL	1200	LAPTOP#OPERATION_PERFORMANCE	632
LAPTOP#DESIGN_FEATURES	489	LAPTOP#PRICE	208
LAPTOP#QUALITY	201	KEYBOARD#GENERAL	156
DISPLAY#GENERAL	145	BATTERY#OPERATION_PERFORMANCE	144
DISPLAY#OPERATION_PERFORMANCE	141	KEYBOARD#DESIGN_FEATURES	140

Table 1: Top 10 Categories by Count in the Laptop Training Dataset

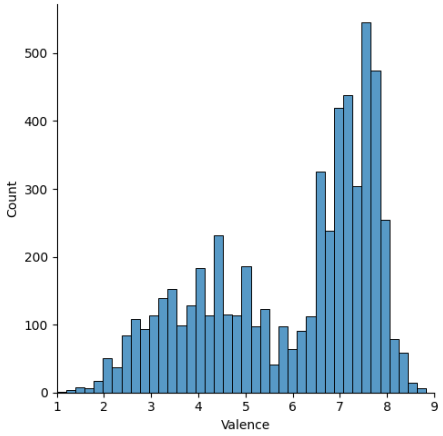

Figure 2: Distribution of Valence scores

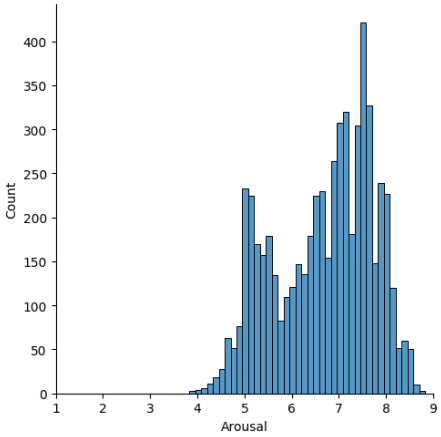

Figure 3: Distribution of Arousal scores

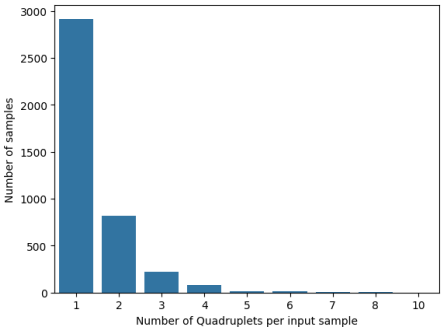

Figure 4: Number of output quads per input

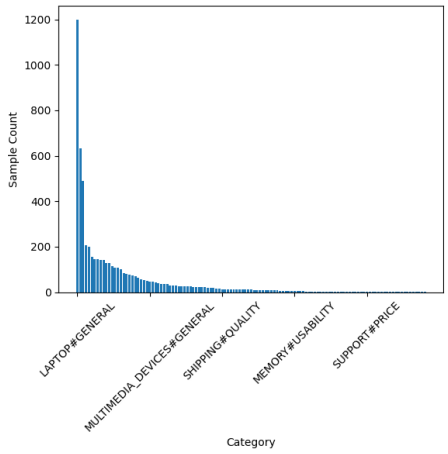

Figure 5: Distribution of Categories

Figure 6: Wordcloud of the most frequent words

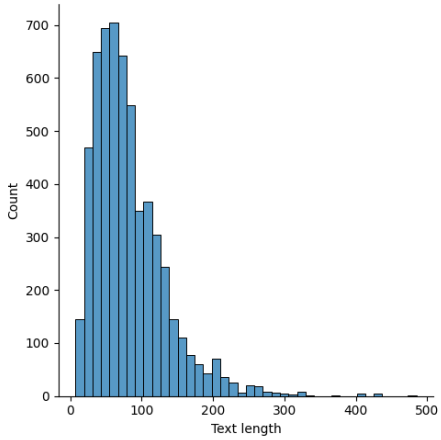

Figure 7: Text length distribution

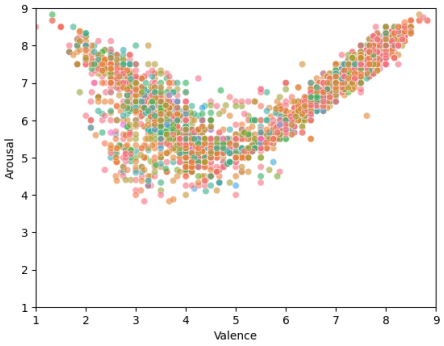

Figure 8: Valence-Arousal scores, colored by category

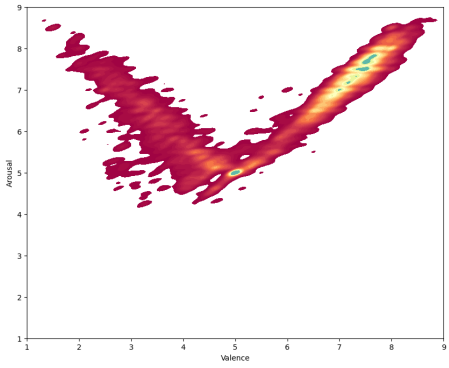

Figure 9: KDE plot of Valence-Arousal

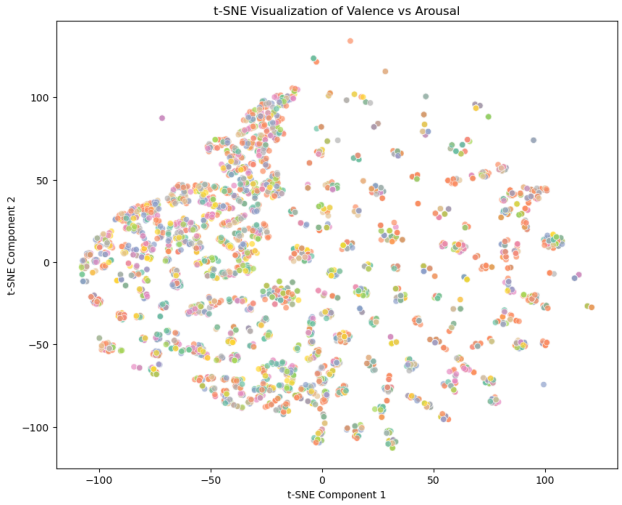

Figure 10: T-SNE diagram of Valence-Arousal scores, colored by category

3.1.2 Restaurant Dataset

This dataset has 2284 training samples and 200 test samples. The following statistics were computed over the training dataset:

- **Number of output quadruplets:** 3659
  - 880 *NULL Aspects*
  - 699 *NULL Opinions*

  (Note: *NULL* values represent aspects or opinions that are absent from the text and are not necessarily invalid.)
- **Number of Categories:** 14
- **Maximum input text length:** 399 characters
- **Maximum quads per input:** 10
  - Input text: *"your a sushi fan , you love expertly cut fish , great sake , a killer soho location , and of course : salmon , tuna , fluke , yellow tail , cod , mackeral , jellyfish , sea urchin , shrimp , lobster , sea bream , trout , milk fish , blue fin tuna , eel , crab , sardine , monk fish , roe , scallop , oysters , and a varity of toro ."* (332 characters)
  - Aspect/Opinion pairs: *sushi#NULL|fish#love|sake#great|soho location#killer|salmon#NULL|tuna#NUL tail#NULL|cod#NULL|mackeral#NULL|jellyfish#NULL|sea urchin#NULL|shrimp#NULL|lobster#NUL bream#NULL|trout#NULL|milk fish#NULL|blue fin tuna#NULL|eel#NULL|crab#NULL|sardine#NULL fish#NULL|roe#NULL|scallop#NULL|oysters#NULL|toro#NULL|*

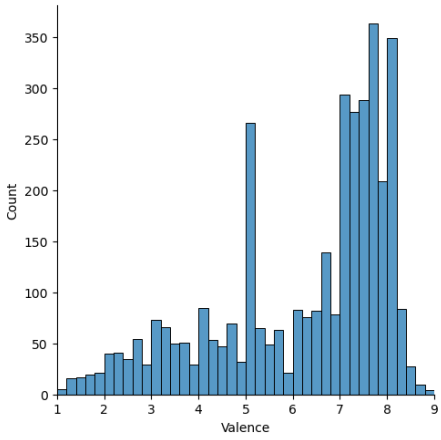

Figure 11: Distribution of Valence scores

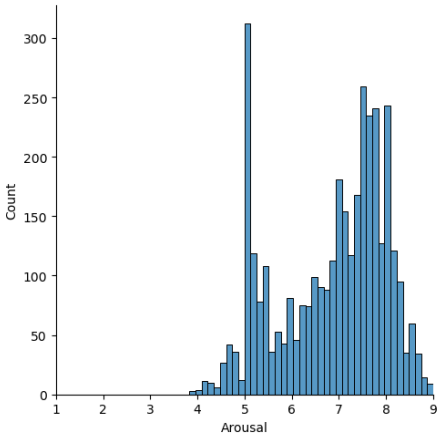

Figure 12: Distribution of Arousal scores

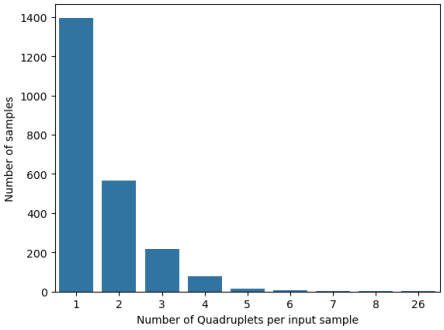

Figure 13: Number of output quads per input

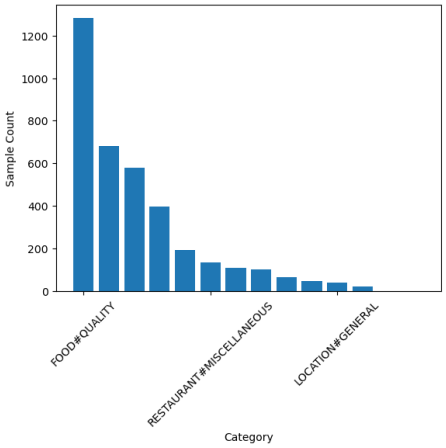

Figure 14: Distribution of Categories

Figure 15: Wordcloud of the most frequent words

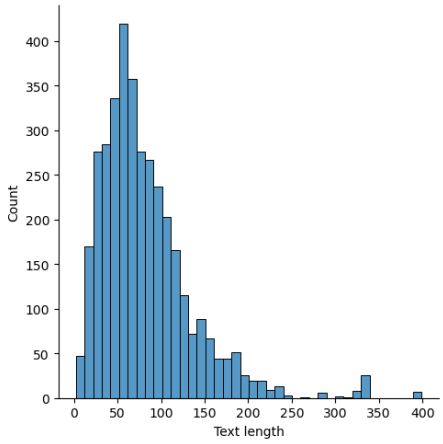

Figure 16: Text length distribution

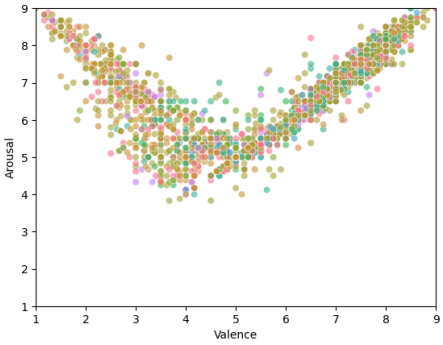

Figure 17: Valence-Arousal scores, colored by category

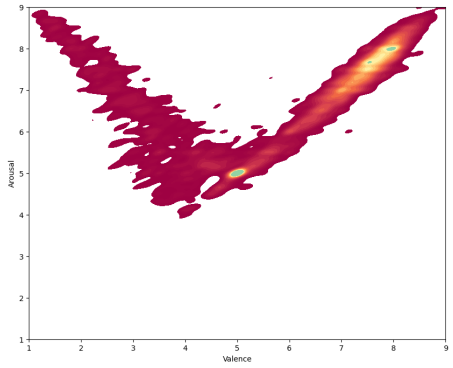

Figure 18: KDE plot of Valence-Arousal

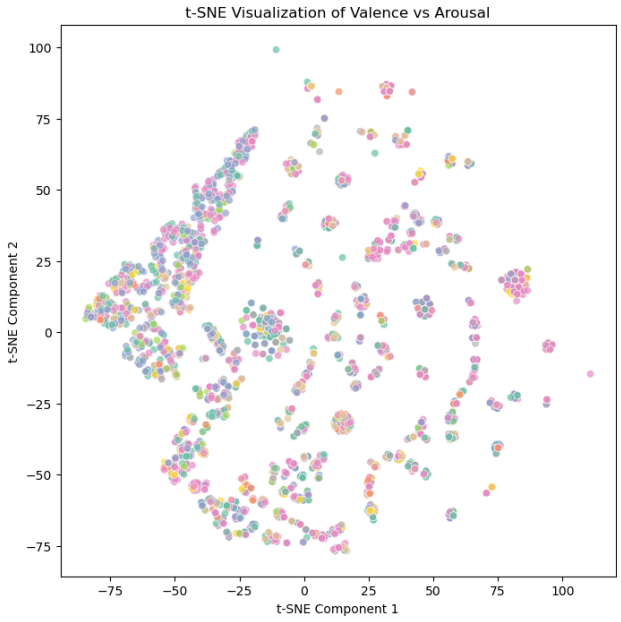

Figure 19: T-SNE diagram of Valence-Arousal scores, colored by category

Category(1/2)	Count	Category(2/2)	Count
FOOD#QUALITY	120012	RESTAURANT#MISCELLANEOUS	135
SERVICE#GENERAL	680	FOOD#PRICES	111
RESTAURANT#GENERAL	579	RESTAURANT#PRICES	101
AMBIENCE#GENERAL	398	DRINKS#QUALITY	66
FOOD#STYLE_OPTIONS	192	DRINKS#STYLE_OPTIONS	47

Table 2: Top 10 Categories by Count in the Restaurant Training Dataset

4 Project-Level Architecture and Approaches

Figure 20 presents the high-level pipeline spanning all subtasks. The implemented solution focuses on Subtask 3, while Subtasks 1 and 2 are outlined as potential modules.

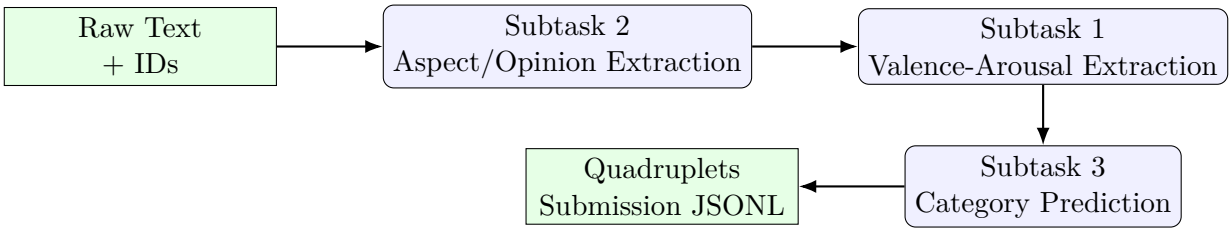

Figure 20: Project-level pipeline spanning Subtasks 1–3 using a zig-zag layout to improve readability.

4.1 Subtask 1: Valence–Arousal Regression

The goal of Subtask 1 is to predict continuous *Valence* and *Arousal* scores associated with a given textual instance. Valence captures the polarity of sentiment (negative to positive), while Arousal reflects the emotional intensity expressed in the text. Unlike the other subtasks, Subtask 1 is formulated as a **regression problem** over continuous affective dimensions rather than a discrete classification task.

4.1.1 Task Formulation and Input Representation

Each instance in Subtask 1 consists of a sentence annotated with gold Valence–Arousal (VA) values. The task is to learn a mapping from the raw textual input to a pair of real-valued scores:

$$(\text{Valence}, \text{Arousal}) \in \mathbb{R}^2$$

In our approach, the input to the model is the original sentence text, without explicit aspect or opinion markers. This design choice follows the official task definition and aims to capture the overall affective signal conveyed by the sentence.

4.1.2 Model Architecture

We address Subtask 1 using a transformer-based regression model built on top of **DistilBERT-base-uncased**. DistilBERT offers a favorable trade-off between performance and computational efficiency, making it well suited for regression over large datasets.

The architecture consists of:

- A **DistilBERT encoder** with 6 transformer layers and hidden size 768.

- A lightweight **regression head** applied to the final pooled representation, implemented as a linear layer with two output neurons corresponding to Valence and Arousal.

The model directly outputs continuous VA predictions, without any discretization or binning.

4.1.3 Training Setup and Optimization

The model is fine-tuned using **mean squared error (MSE)** loss, computed jointly over Valence and Arousal. Optimization is performed with the AdamW optimizer.

Training is conducted on the combined **laptop** and **restaurant** datasets provided for Subtask 1. The official training data is split into training and validation subsets using an **80% / 20%** ratio.

The main hyperparameters are summarized below:

- **Optimizer**: AdamW
- **Learning rate**: 2×10^{-5}
- **Batch size**: 16
- **Maximum sequence length**: 128 tokens
- **Training epochs**: 5

All experiments were run locally using the Hugging Face *Transformers* library and PyTorch.

4.1.4 Training Dynamics

During training, the model exhibited stable convergence behavior. The training loss decreased steadily across epochs, indicating that the model successfully learned to regress both affective dimensions from textual input. No signs of severe overfitting were observed within the selected number of epochs.

4.1.5 Evaluation

Model performance was monitored using regression loss on the validation split. While Subtask 1 does not involve categorical accuracy metrics, the learned VA representations proved sufficiently informative to be used as input signals for downstream subtasks.

In particular, the predicted Valence–Arousal scores were later integrated into Subtask 3 as part of the structured input representation, contributing to improved category disambiguation in sentiment classification.

Overall, the DistilBERT-based regression approach provides a simple yet effective solution for affective modeling in Subtask 1, serving as a foundational component for the subsequent stages of the ABSA pipeline.

4.2 Subtask 2: Aspect-Opinion Extraction

The objective of this subtask is to extract *Aspect–Opinion* term pairs from a given input text. Both Aspects and Opinions may take the value *NULL*, representing general or implicit sentiments that cannot be directly aligned with explicit lexical units in the text.

Model	laptop dataset	laptop + restaurant dataset	Competition results (subtask 2/subtask 3)
t5-small	55%	63%	0.4502/0.2405
t5-base	58%	68%	0.4690/0.2405

Table 3: F1 scores of trained models on the laptop dataset

4.2.1 Model Architecture

To address this task, we adopt a *sequence-to-sequence* (seq2seq) formulation based on the **T5** architecture. The primary model used is **T5-base** (approximately 200M parameters), implemented using the Hugging Face *Transformers* library and trained with the *Seq2SeqTrainer* framework. All models were trained on Kaggle's GPU environment using GPU T4 x2. Training time did not exceed six hours per model.

Input texts were minimally processed: aside from tokenization, no additional normalization or preprocessing steps were applied prior to training.

To enable structured prediction of Aspect–Opinion pairs, model outputs were constrained to the following linearized format:

```
Aspect1#Opinion1|Aspect2#Opinion2|
```

This representation allows the model to generate multiple Aspect–Opinion pairs within a single output sequence. Since generated outputs did not always exactly match the word forms present in the input text, a post-processing step was applied. This step corrected incomplete *NULL* tokens, normalized contracted negations (e.g., *not* $\rightarrow$ *'t*), and enforced exact case-sensitive matching with the input text whenever possible.

The main hyperparameters are summarized below:

- **Optimizer**: AdamW
- **Learning rate**: 5×10^{-5}
- **Weight decay**: 0.01
- **Batch size**: 4
- **Training epochs**: 25 (with early stopping)

4.2.2 Evaluation Metric

Model performance was evaluated using the F1 score. No ordering constraint was imposed on the predicted pairs, and partial matches were not counted as correct. This metric emphasizes the correctness of predicted Aspect-Opinion pairs rather than sequence-level similarity.

4.2.3 Results

Two seq2seq models were trained: a *T5-small* model (60M parameters) and a *T5-base* model (200M parameters). First they were trained on 80% of the laptop training dataset, and the following experiment trained on both the entire restaurant dataset and 70% of the laptop dataset. Results and competition submission results are detailed in Table3.

4.3 Subtask 3: Category Classification

The goal of Subtask 3 is to predict the sentiment *Category* corresponding to a given *Aspect–Opinion* pair, together with its associated Valence–Arousal (VA) score. The task is formulated as a multi-class classification problem over the predefined `Entity#Attribute` label space.

4.3.1 Label Space and Input Construction

For the laptop domain, the category space consists of **22 entities** and **9 attributes**, resulting in a total of **198 possible categories**. Each training instance corresponds to exactly one category label.

To make full use of the available supervision, each example is converted into a structured textual input sequence that explicitly encodes all relevant fields:

```
Text: <text> Aspect: <aspect>. Opinion: <opinion>. VA: <va>
```

This representation allows the model to jointly condition on the sentence-level context, the extracted aspect, the associated opinion expression (or `NULL`), and the affective signal provided by the VA score.

4.3.2 Model Architecture

Category prediction is approached using a transformer-based sequence classification model. We fine-tune **BERT-base-uncased** as implemented in the Hugging Face *Transformers* library.

The architecture follows the standard BERT classification setup:

- A **BERT encoder** with 12 layers, hidden size 768, and 12 self-attention heads.
- The final hidden representation of the `[CLS]` token is fed into a **linear classification head**.
- A softmax layer produces a probability distribution over the labels.

Figure 21 illustrates the end-to-end processing pipeline for Subtask 3.

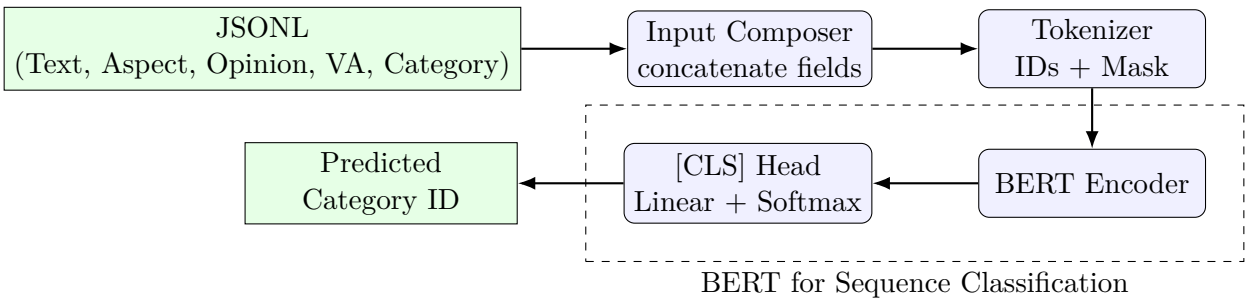

Figure 21: Architecture of the proposed model for Subtask 3 category classification.

4.3.3 Training Setup

Training and validation were performed using an **80% / 20% split** of the official laptop dataset, resulting in **4,618 training samples** and **1,155 validation samples**. Due to the presence of extremely rare classes in the tail of the distribution, a standard random split was employed as stratified sampling was not feasible.

To mitigate the impact of class imbalance during optimization, we computed **inverse-frequency class weights**. These weights were integrated into the Cross-Entropy loss function, penalizing misclassifications of rare categories more heavily than dominant ones.

All experiments were run in **Google Colab** using a single **NVIDIA T4 GPU**. The model was trained using PyTorch and the Hugging Face *Trainer* API with the following hyperparameters:

- **Optimizer**: AdamW
- **Learning rate**: 3×10^{-5}
- **Weight decay**: 0.05
- **Scheduler**: Linear with 10% warmup steps
- **Batch size**: 8
- **Epochs**: 35 (with early stopping)

Early stopping was applied based on the validation **Macro-F1** score, with a patience of 3 epochs and a minimum improvement threshold of 10^{-4}.

4.3.4 Evaluation Metrics

Model performance was assessed using:

- **Accuracy**, measuring exact category prediction correctness.
- **Macro-F1**, computed as the unweighted average F1-score across all categories.
- **Weighted Cross-entropy loss**.

Given the imbalanced nature of the dataset, Macro-F1 was prioritized as the primary metric for model selection.

4.3.5 Results

During training, the model showed significant convergence in the initial epochs. The best validation performance was achieved at **epoch 19**, yielding a validation Macro-F1 of **0.3932** and an accuracy of **63.12%**. Following the patience criteria, training was automatically halted at epoch 22 to prevent overfitting.

Final evaluation on the held-out validation set confirms these metrics, as reported in Table 4.

Table 4: Final evaluation results on the laptop validation split (20%).

Split	Loss	Accuracy	Macro-F1
Laptop (validation)	1.7477	0.6312	0.3932

These results indicate that while the model learns to classify the dominant categories effectively (reflected in the higher accuracy), the lower Macro-F1 score highlights the challenge of correctly predicting rare, long-tail categories within the fine-grained sentiment taxonomy.

5 Competition Results

Or final submission had the following results:

TASK 1: DIMASR (VA PREDICTION)

LANGUAGE	DOMAIN	RMSE_VA	PCC_V	PCC_A
ENG	laptop	1.9153	0.6165	0.3048
	restaurant	1.8942	0.6702	0.2545
	AVERAGE	1.9047	0.6434	0.2797

Figure 22: Competition results on the first subtask

TASK 2: DIMASTE (TRIPLET EXTRACTION)

LANGUAGE	DOMAIN	CF1	CPRECISION	CRECALL	CTP	TP	FP	FN
ENG	laptop	0.4690	0.4766	0.4616	146.3147	189	118	128
	restaurant	0.5532	0.5652	0.5417	220.9968	286	105	122
	AVERAGE	0.5111	0.5209	0.5016	183.6557	237.5	111.5	125.0

Figure 23: Competition results on the second subtask

TASK 3: DIMASQP (QUADRUPLET EXTRACTION)

LANGUAGE	DOMAIN	CF1	CPRECISION	CRECALL	CTP	TP	FP	FN
ENG	laptop	0.2405	0.2437	0.2375	75.2880	98	211	219
	AVERAGE	0.2405	0.2437	0.2375	75.2880	98.0	211.0	219.0

Figure 24: Competition results on the third subtask

References

[1] Zhang, Wenxuan and Li, Xin and Deng, Yang and Bing, Lidong and Lam, Wai: A Survey on Aspect-Based Sentiment Analysis: Tasks, Methods, and Challenges IEEE Transactions on Knowledge and Data Engineering (2022).

[2] Yusuf, Kabir, Ogbuju, Emeka, Abiodun, Taiwo, Oladipo, Francisca: A Technical Review of the State-of-the-Art Methods in Aspect-Based Sentiment Analysis Journal of Computing Theories and Applications (2024).

[3] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL-HLT (2019).

[4] Wolf, T., Debut, L., Sanh, V., et al.: Transformers: State-of-the-Art Natural Language Processing. EMNLP (2020).

[5] Zhen Wu, Chengcan Ying, Fei Zhao, Zhifang Fan, Xinyu Dai, Rui Xia: Grid Tagging Scheme for Aspect-oriented Fine-grained Opinion Extraction EMNLP (2020).

[6] Yue Mao, Yi Shen, Jingchao Yang, Xiaoying Zhu, Longjun Cai: Seq2Path: Generating Sentiment Tuples as Paths of a Tree ACL (2022).

[7] Abigail See and Peter J. Liu and Christopher D. Manning: Get To The Point: Summarization with Pointer-Generator Networks arXiv preprint (2017).