Advanced NLP

Ex1

Yael Batat - 213560097

1 Open Questions

1.

- **BoolQ (Boolean Questions)**
  BoolQ consists of yes/no questions based on short passages from Wikipedia. It measures intrinsic language understanding by requiring models to assess whether a passage logically entails a given question, thereby testing **textual entailment and inference skills**.
- **Winograd Schema Challenge (WSC)**
  WSC presents sentences with ambiguous pronouns and asks models to determine the correct referent (e.g., "What is too small?"). This tests the model's ability to perform **coreference resolution**, a fundamental aspect of understanding sentence structure and meaning.
- **KILT – ALEXA / Wikipedia QA**
  In this dataset, QA is used to ground textual mentions to specific **real-world entities** in a knowledge base like Wikipedia. This reflects **entity linking**, an intrinsic skill that combines linguistic disambiguation with knowledge grounding to resolve meaning precisely.

2. A.

1. Chain-of-Thought (CoT) Prompting

- **Description:** The model generates intermediate reasoning steps before producing a final answer.
- **Advantages:** Improves accuracy on complex tasks (e.g., math, logic) by breaking down problems.
- **Bottlenecks:** Longer sequences increase decoding time and memory usage.
- **Parallelizable?** No (decoding is sequential).

2. Self-Consistency

- **Description:** Generates multiple CoT paths via sampling and selects the majority answer.
- **Advantages:** Boosts reliability by aggregating diverse reasoning paths.
- **Bottlenecks:** Multiple forward passes increase compute and latency.
- **Parallelizable?** Yes (independent paths can be generated in parallel).

3. Verifier-Based Methods

- **Description:** Uses rule-based or trained models to filter/correct outputs.
- **Advantages:** Improves output quality by rejecting incorrect answers.
- **Bottlenecks:** Additional compute for verification (e.g., running another LLM).
- **Parallelizable?** Yes (verification can run independently of generation).

4. Planning & Self-Correction (e.g., O1)

- **Description:** Models plan, backtrack, and self-evaluate during inference.
- **Advantages:** Produces more accurate and self-reflective answers.
- **Bottlenecks:** Multiple iterative steps increase compute/time.
- **Parallelizable?** No (steps are interdependent).

B. For a complex scientific task on a single GPU, I would select **Self-Consistency**. Self-Consistency inherently generates multiple Chain-of-Thought-style reasoning paths through sampling (implicit CoT), then selects the most frequent answer—combining the benefits of structured reasoning and error resilience in one streamlined process. It maximizes GPU utilization via parallel sampling of reasoning paths, avoids sequential bottlenecks of methods like Least-to-Most, and requires no additional training like verifiers. While pure CoT alone improves reasoning, Self-Consistency strictly dominates it for scientific tasks by aggregating over diverse solutions, making it the optimal single-method choice when both accuracy and hardware efficiency are priorities.

2 Programming Exercise

2.

- **Evaluation of Hyperparameter Tuning and Test Performance**

I tested three different configurations for fine-tuning the BERT model on the MRPC dataset, each with distinct combinations of learning rate, batch size, and number of training epochs. Below are the results:

Configuration	Epochs	Learning Rate	Batch Size	Validation Accuracy	Test Accuracy
Model 1	2	2e-5	16	0.8309	0.8301
Model 2	3	5e-5	32	0.8627	**0.8417**
Model 3	5	1e-5	8	**0.8701**	0.8394

Although the configuration in Model 3 achieved the highest validation accuracy (87.01%), it did not yield the best test accuracy. Instead, Model 2, with a slightly lower validation accuracy of 86.27%, achieved the highest test accuracy at 84.17%. This indicates that while Model 3 may have fit the validation data more closely, it potentially began to overfit, resulting in slightly reduced generalization performance on unseen data. Model 2, on the other hand,

appears to have maintained a better balance between learning and generalization, which is reflected in its stronger test set performance.

- ## Qualitative Analysis: Best vs. Worst Configuration

To better understand the performance differences between configurations, I wrote additional code to compare the predictions of the best-performing model (epoch 3, learning rate 5e-5, batch size 32) and the worst-performing model (epoch 2, learning rate 2e-5, batch size 16) on the validation set. I modified the script to accept command-line arguments for specifying the paths to both models and to generate a comparison output. The script identified examples where the best model correctly predicted the label, while the worst model failed. The results were saved in a file named comparison_output.txt, containing 13 disagreement examples. I used these examples to examine and characterize the types of mistakes made by the lower-performing configuration, which revealed several patterns that highlight the specific challenges it faced.

1. Lexical Overlap with Contrasting Meaning

Many challenging examples shared significant lexical overlap, such as repeated entities or numbers, but conveyed opposite or unrelated meanings. The lower-performing model often misclassified these cases, likely because it focused on surface-level similarity rather than deeper semantic differences.

Example 90

- Sentence 1: "Lapidus expects foreign brands' sales to be **up** 4 percent..."
- Sentence 2: "Lapidus expects Ford to be **down** 5 percent..."
- True Label: 0 (not paraphrase)
- Model 1 Prediction: 1 (incorrect), Model 2 Prediction: 0 (correct)

Although the sentences mention similar topics and entities, the actual claims are contrasting. The worst model ignored contrasting numerical values (up 4% vs. down 5%), suggesting poor sensitivity to negations and quantifiers.

2. Failure to Capture Fine-Grained Factual Differences

Some examples differed only in specific facts, such as numerical values or statements about timing. These fine-grained differences were often missed by the lower-performing model, which tended to overgeneralize.

Example 375

- *Sentence 1:* "The euro was at 1.5281..., **up 0.2 percent**..."
- *Sentence 2:* "The euro was **steady**..., at 1.5261..."
- *True Label:* 0 (not paraphrase)
- *Model 1 Prediction: 1 (incorrect), Model 2 Prediction: 0 (correct)*

A minor numerical change and shift in meaning ("up" vs. "steady") caused confusion for the lower-performing model.

3. Syntactic Variation with Equivalent Meaning

In several positive examples (true label = 1), the sentences conveyed the same meaning using different syntax or phrasing. The higher-performing model was more robust to these variations, while the lower-performing one sometimes failed to generalize beyond word order or structure.

Example 362

- *Sentence 1:* "...stemming from the disturbance **which** led to Rosenbaum's death."
- *Sentence 2:* "...stemming from the disturbance **that** led to Rosenbaum's death."
- *True Label:* 1 (paraphrase)
- *Model 1 Prediction: 0 (incorrect), Model 2 Prediction: 1 (correct)*

The difference between "which" and "that" is syntactic, not semantic. The best model correctly predicted this as a paraphrase.