**Assignment 02 — Chain-of-Thought (CoT) Evaluation on Reasoning Benchmarks**

**Objective:**

The purpose of this assignment is to explore and understand evaluation methodologies for language models, particularly in the context of reasoning tasks. The task focuses on replicating benchmark evaluations to compare model performance with and without Chain-of-Thought (CoT) prompting.

**Model Used:**

1. Model: deepseek-ai/deepseek-coder-1.3b-base
2. Mode: CPU-only inference
3. Framework: Hugging Face Transformers

**Benchmark Dataset:**

GSM8K (Grade School Math 8K) – a standard benchmark for evaluating arithmetic and multi-step math reasoning in language models.

**Task Overview:**

Evaluate the DeepSeek model using benchmark datasets (e.g., GSM8K).

Run two separate evaluation regimes:

1. Direct answering (without CoT).
2. CoT-style answering (step-by-step reasoning).

Use custom Python scripts to measure model accuracy on both settings.

Record predictions and calculate answer correctness using exact match.

**Evaluation Method:**

1. For each prompt, compare model prediction with the ground truth answer.
2. Accuracy is measured as:
3. accuracy = (correct\_predictions / total\_questions) × 100
4. Charts are generated to visualize performance gains due to CoT prompting.

**Key Observations:**

1. CoT prompting leads to significantly higher accuracy in multi-step questions.
2. In problems requiring intermediate computation or logic, CoT helps the model maintain context.
3. Without CoT, the model sometimes guesses or skips intermediate reasoning.

**Deliverables:**

1. Accuracy summary table: with-CoT vs. without-CoT.
2. Bar chart visualization of score comparison.
3. Codebase includes benchmarking script with prompt templates.

**Software Requirements:**

Python ≥ 3.10 (tested on Python 3.13)

**Required Python packages**:

1. transformers
2. torch
3. accelerate
4. safetensors
5. matplotlib
6. tqdm
7. datasets

**Installation Command Example:**

pip install transformers torch accelerate safetensors matplotlib tqdm datasets

**Notes:**

1. The evaluation script is compatible with both CPU and GPU environments.
2. Benchmarking follows consistent prompt templates to ensure fair comparisons.
3. This assignment emphasizes the importance of structured prompting in LLM evaluation pipelines.

**Author:**

Bhagyesh Makhamale

Under guidance of: Dr. Yogita Mahatekar

Affiliation: COEP Technological University (Department of AI & DS)

Academic Year: 2025–2026