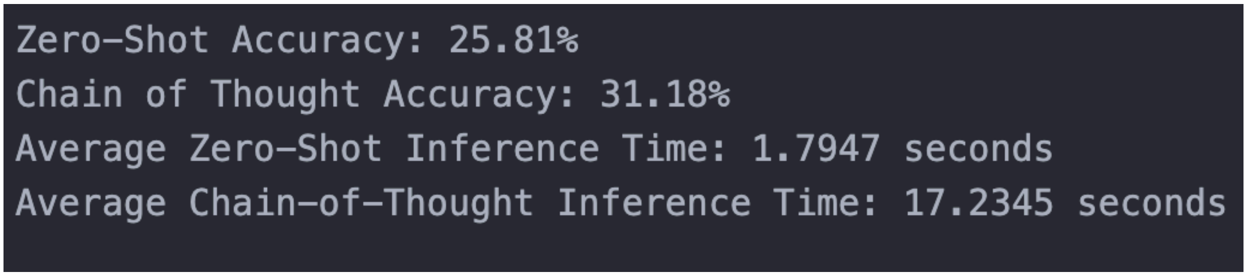
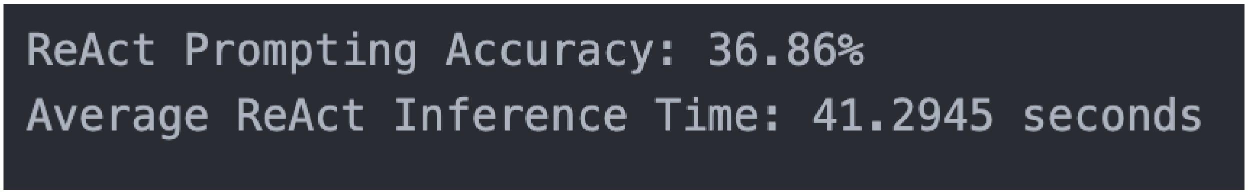
LLM ASSIGNMENT 2

SANMAY SOOD
2021095

GEMMA Results

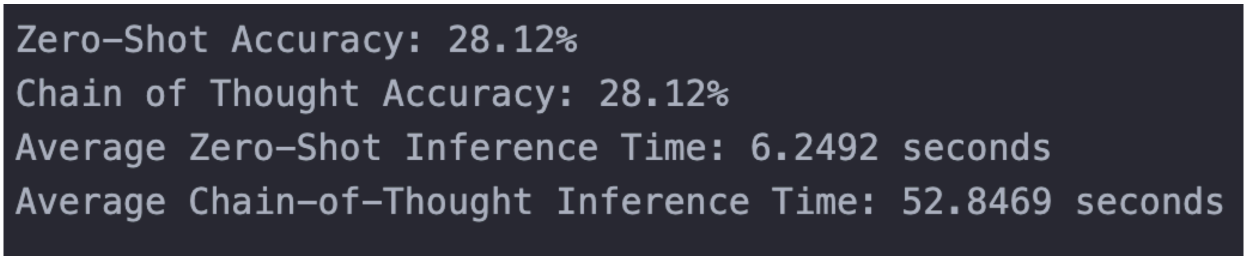


```
Zero-Shot Accuracy: 25.81%
Chain of Thought Accuracy: 31.18%
Average Zero-Shot Inference Time: 1.7947 seconds
Average Chain-of-Thought Inference Time: 17.2345 seconds
```

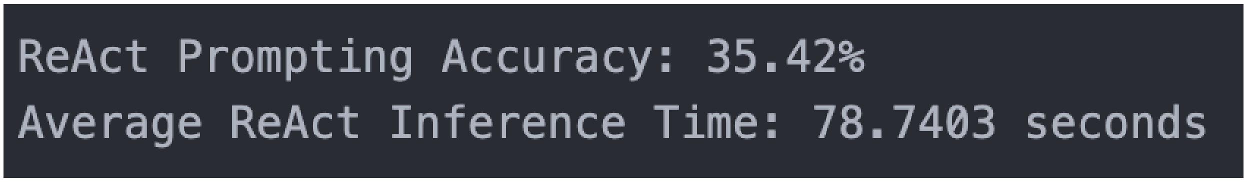


```
ReAct Prompting Accuracy: 36.86%
Average ReAct Inference Time: 41.2945 seconds
```

PHI Results

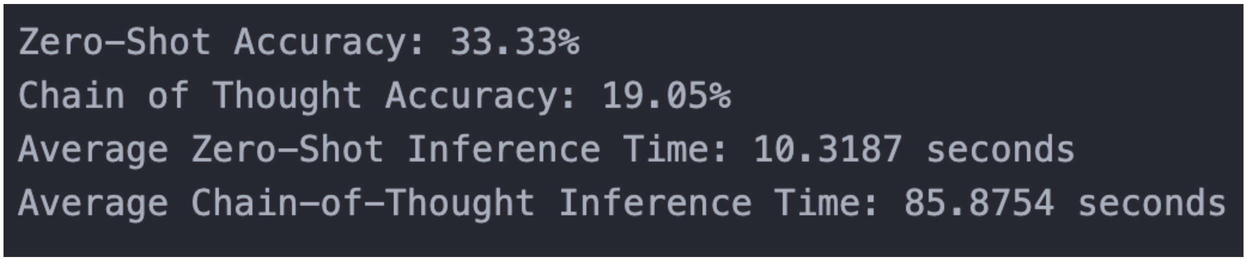


```
Zero-Shot Accuracy: 28.12%
Chain of Thought Accuracy: 28.12%
Average Zero-Shot Inference Time: 6.2492 seconds
Average Chain-of-Thought Inference Time: 52.8469 seconds
```

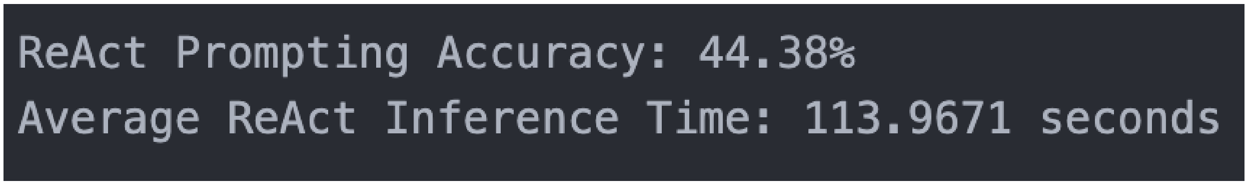


```
ReAct Prompting Accuracy: 35.42%
Average ReAct Inference Time: 78.7403 seconds
```

LLAMA Results



```
Zero-Shot Accuracy: 33.33%
Chain of Thought Accuracy: 19.05%
Average Zero-Shot Inference Time: 10.3187 seconds
Average Chain-of-Thought Inference Time: 85.8754 seconds
```



```
ReAct Prompting Accuracy: 44.38%
Average ReAct Inference Time: 113.9671 seconds
```

Inference Times

1. **Gemma (2 billion parameters)**
   - **Zero-shot:** 1.8s
   - **COT:** 17.2s
   - **React:** 41.3s
2. **Phi (3.5 billion parameters)**
   - **Zero-shot:** 6.2s
   - **COT:** 52.8s
   - **React:** 78.7s
3. **Llama (8 billion parameters)**
   - **Zero-shot:** 10.3s
   - **COT:** 85.8s
   - **React:** 113.9s

Inference Time Analysis

1. **Zero-shot Inference:**
   - **Performance Trend:** As the number of parameters increases, the inference time also increases. This suggests that larger models require

    more computational resources to generate responses, even in zero-shot settings.
  - **Time Comparison:**
    - Gemma is the fastest at 1.8s, followed by Phi at 6.2s and Llama at 10.3s.
2. **COT Inference:**
   - **Performance Trend:** The increase in time is more pronounced in this setting. Gemma takes 17.2s, while Phi takes 52.8s, and Llama takes 85.8s. This indicates that larger models might be leveraging their capacity for more complex reasoning, but at a significant cost to speed.
   - **Time Comparison:** The gap between Gemma and Phi (35.6s) and between Phi and Llama (33.1s) shows that scaling up in parameters significantly affects the time needed for COT tasks.
3. **React Inference:**
   - **Performance Trend:** React strategies show the longest inference times across all models. The trend is consistent with COT, where larger models take even longer. Gemma takes 41.2s, Phi 78.7s, and Llama 113.9s.
   - **Time Comparison:** The time increases considerably between models, with Gemma to Phi increasing by 37.5s and Phi to Llama by 35.2s, highlighting the growing complexity and computational load.

The trend of inference times across the different prompting strategies—zero-shot, Chain of Thought (COT), and React—shows a clear pattern of increasing complexity and computational demand.

1. **Zero-shot Prompts:**
   - **Lowest Inference Time:** This strategy has the shortest inference time across all models. It requires minimal processing since the model generates responses without additional context or reasoning.
2. **Chain of Thought (COT) Prompts:**
   - **Moderate Inference Time:** COT prompts require the model to engage in more complex reasoning, resulting in significantly longer inference times compared to zero-shot. The model articulates its reasoning steps, increasing computational demand.
3. **React Prompts:**
   - **Highest Inference Time:** React strategies involve interactive and iterative reasoning, leading to the longest inference times. While they still require extensive processing, the increase from COT to React is generally less steep than from zero-shot to COT.

Model Accuracy Overview

1. **Gemma (2 billion parameters)**
   - **Zero-shot:** 25.81
   - **COT:** 31.18
   - **React:** 36.86
2. **Phi (3.5 billion parameters)**
   - **Zero-shot:** 28.12
   - **COT:** 28.12
   - **React:** 35.42
3. **Llama (8 billion parameters)**
   - **Zero-shot:** 33.33
   - **COT:** 19.05
   - **React:** 44.38

Accuracy Analysis

1. **Zero-shot Accuracy:**
   - **Performance Trend:** Llama performs the best in zero-shot tasks with an accuracy of 33.3, followed by Phi at 28.1 and Gemma at 25.8. This indicates that, in this mode, larger models can leverage their training data more effectively.
   - **Insights:** The accuracy improvement from Gemma to Llama suggests that more parameters may enhance the model's generalization capabilities for zero-shot tasks.
2. **COT Accuracy:**
   - **Performance Trend:** The accuracy for COT tasks is somewhat surprising. Gemma achieves the highest accuracy at 31.2, while Phi and Llama both have lower accuracy at 28.1 and 19.0, respectively.
   - **Insights:** The lower COT accuracy of Llama compared to smaller models like Gemma could stem from multiple factors beyond just model size. Issues such as token length constraints, suboptimal prompt design, tuning of model parameters, or ineffective answer extraction procedures could all contribute to this unexpected performance. These variables should be adjusted and optimized to enhance performance on COT tasks, especially for larger models.
3. **React Accuracy:**
   - **Performance Trend:** React strategy shows Llama excelling with an accuracy of 44.38, followed by Gemma at 36.86 and Phi at 35.1. This indicates that Llama is particularly effective in tasks that allow it to utilize its larger capacity.

    - **Insights:** The React strategy seems to leverage the larger model's strengths more effectively, likely due to its ability to process information more thoroughly and generate nuanced responses.

Trade-offs Between Model Size, Speed, Prompt Type and Output Quality

Here's a detailed discussion of these interdependencies:

1. Model Size

- **Larger Models (e.g., Llama):**
    - **Advantages:** Typically offer better output quality, especially in complex tasks. They have a greater capacity to understand nuances in language and context due to their larger parameter count.
    - **Disadvantages:** Inference speed tends to be slower because they require more computational resources and time to process information. This can be a significant drawback for real-time applications.
- **Smaller Models (e.g., Gemma):**
    - **Advantages:** Faster inference times, making them suitable for applications where speed is critical. They are often easier to deploy in environments with limited computational power.
    - **Disadvantages:** May struggle with complex tasks and offer lower output quality compared to larger models.

2. Inference Speed

- **Impacts of Speed:**
    - Faster inference allows for real-time applications, such as chatbots or interactive tools, where user experience relies on quick responses.
    - Slower models may be acceptable in batch processing scenarios or tasks where accuracy and depth of response are prioritized over speed.

3. Prompt Type

- **Zero-shot Prompts:**
    - **Characteristics:** These prompts require the model to generate responses without any specific context or examples.

    - **Performance:** Generally results in faster inference but can lead to lower accuracy, especially for complex tasks. Smaller models may perform adequately in this scenario.
- **Chain of Thought (COT) Prompts:**
    - **Characteristics:** These prompts encourage the model to articulate reasoning steps, often resulting in higher-quality outputs.
    - **Performance:** Slower due to the need for detailed processing but may yield better accuracy for certain tasks. However, larger models do not always benefit as expected from this approach.
- **React Prompts:**
    - **Characteristics:** Designed to encourage interaction and iterative thinking, allowing models to dynamically adapt their responses.
    - **Performance:** Often produces the highest output quality, especially with larger models, but incurs the highest inference costs.

4. Output Quality

- **Quality vs. Complexity:**
    - Larger models tend to provide higher-quality outputs for more nuanced tasks, as they can capture and process a wider range of linguistic patterns.
    - However, output quality can diminish in scenarios where the model is overfitted to certain datasets, or when it misinterprets the context due to poor prompting.

Comparative Analysis of Gemma, Phi, and LLaMA Models: Performance, Architecture, and Efficiency

1. Gemma (2B) vs. Phi (3.5B) and LLaMA (8B)

Despite its smaller size, **Gemma (2B)** performs comparably or better than larger models like **Phi (3.5B)** and **LLaMA (8B)**. This can be attributed to several key factors:

- **TPU-Optimized Training**: Gemma leverages **TPU-optimized training** as part of **Google's Pathways system**, which significantly reduces computational overhead and accelerates training efficiency. TPU utilization enables Gemma to achieve faster inference times and high accuracy even with fewer parameters .

This kind of optimization helps Gemma achieve results that typically require larger models like LLaMA.

- **Fine-Tuning with RLHF**: Gemma's fine-tuning process includes **Reinforcement Learning from Human Feedback (RLHF)**, which improves its generalization capabilities. RLHF has been shown in multiple studies to enhance performance on a broad range of tasks, making models like Gemma more effective in tasks where human-like reasoning or decision-making is necessary .

2. Phi (3.5B) vs. LLaMA (8B)

Phi (3.5B), despite being smaller than LLaMA, excels in tasks requiring **reasoning and structured outputs**, such as Chain of Thought (COT) and ReAct prompts. Several factors contribute to Phi's competitive performance:

- **Mixture of Experts (MoE)**: Phi employs an **MoE architecture**, which allows for more efficient use of computational resources by dynamically selecting specific parts of the network during inference . This selective engagement of experts results in **higher computational efficiency** and allows Phi to compete with much larger models like LLaMA. The MoE architecture improves both task-specific performance and speed in inference.
- **High-Quality Training Data**: Phi was trained on a dataset of **3.4 trillion tokens**, which ensures a high level of diversity and robustness in its responses. According to research from Microsoft, using large, diverse datasets can often lead to better generalization, even when compared to larger models like LLaMA .

3. LLaMA (8B)

Although **LLaMA** is the largest model in this comparison, its performance, particularly in Chain of Thought (COT) tasks, is lower than expected. The reasons for this are:

- **General-Purpose Design**: LLaMA's architecture is designed for general-purpose language modeling, which may limit its ability to excel in specialized tasks such as reasoning and multi-step decision-making. Meta's research indicates that without fine-tuning for specific tasks, LLaMA can struggle with performance . This explains its lower accuracy in tasks like COT, where structured reasoning is essential.
- **Parameter Inefficiency**: Larger models like LLaMA can suffer from parameter inefficiency when not carefully fine-tuned for specific tasks. As noted in Meta's technical report on LLaMA, merely increasing the parameter count does not always lead to better performance. Effective utilization of these parameters is essential for tasks requiring complex reasoning .

Trade-Offs: Model Size, Speed, and Accuracy

- **Gemma (2B)** demonstrates the **best balance** of speed and accuracy due to its **TPU-optimized training** and RLHF. Despite its smaller size, it excels in inference speed and offers competitive accuracy, making it suitable for real-time applications. It is optimized for multi-lingual tasks with instruction-tuned datasets for improved handling of prompts like **Chain-of-Thought** and **ReAct**. Its smaller size makes it efficient but limits its speed due to local inference constraints.
- **Phi (3.5B)** performs well in **reasoning-heavy tasks** due to its **Mixture of Experts (MoE) architecture**, which efficiently uses computational resources during inference. It offers a middle ground, with decent speed and accuracy, particularly for tasks like COT and ReAct. It is optimized for multi-lingual tasks with instruction-tuned datasets for improved handling of prompts like **Chain-of-Thought** and **ReAct**. Its smaller size makes it efficient but limits its speed due to local inference constraints.
- **LLaMA (8B)**, while delivering the highest accuracy in complex tasks like **ReAct**, suffers from **longer inference times**. Its size contributes to its strong performance in tasks requiring deeper reasoning, but it sacrifices speed and efficiency .

Citations

1. PHI 3.5 Overview - Microsoft Tech Community
2. Google Gemma Open Models
3. LLaMA: Meta AI's Language Model
4. Google Research on Pathways and TPU Optimization (Research Paper).
5. Google Gemma Technical Overview (Hugging Face).
6. **Pathways: Asynchronous Distributed Learning** by Dean et al., Google AI, 2022 (Paper).
7. Microsoft Phi Model and Mixture of Experts (Research Paper).
8. **Scaling Laws for Model Size, Data, and Performance** by Kaplan et al., 2020 (ArXiv Paper).
9. Meta LLaMA 3.1 Research (Research Paper).
10. Meta LLaMA Model Overview, Meta AI 2023 (Technical Report).