ASSIGNMENT 2: PROMPT ENGINEERING & AGENTS, TASK 2: LLMS FOR AI FEEDBACK

Chua Qin Di, 125400010
The Chinese University of Hong Kong, Shenzhen
Shenzhen, China
125400010@luink.cuhk.edu.cn

1 INTRODUCTION

Why the task is important Large language models (LLMs) are now used in many real-world applications, so it is important to ensure their answers follow human preferences and values to increase the usefulness and ethics of AI. In normal supervised learning, models only learn from fixed labels, but this does not reflect how people judge the quality of answers. RLHF (Reinforcement Learning from Human Feedback) improves this by comparing two model outputs and learning which one humans prefer. However, collecting such human feedback takes time and effort. This task explores whether LLMs can provide feedback automatically by comparing and ranking answers, so that the learning and alignment process can be made faster and easier.

Why LLM is suitable to solve the problem Modern LLMs exhibit strong reasoning and judgment capabilities. When given clear instructions, they can compare two answers in terms of relevance, factual correctness, coherence, and helpfulness. Leveraging these properties allows an LLM to act as an evaluator that gives feedback similar to what humans would provide. On a larger scale, LLMs capable of imitating human choices can then be used to train other models or improve the alignment of other systems.

What you did and what you achieved In this task, several prompts were designed and tested to elicit preference judgments from an LLM, using questionanswer pairs derived from the RLHF dataset. Each prompt encourages the model to reason step by step before selecting the preferred answer, and its accuracy was evaluated against ground-truth labels. The experiments demonstrate that structured reasoning prompts significantly improve the reliability of LLM feedback compared to naive zero-shot prompts.

2 PROBLEM DEFINITION

Definition of the task The objective of this task is to evaluate whether an LLM can act as an automated feedback provider that selects the better of two given answers to a question, approximating human preference judgment. Formally, the model receives a triplet (Q, A_1, A_2)—a question and two candidate answers—and outputs which answer better satisfies the question according to relevance, correctness, and helpfulness.

Input: A question `Q` and two answers `a1` and `a2`.

Output: Which one is chosen and rejected

Criteria: The more correlated to the ground truth, the better

Data examples: `https://huggingface.co/datasets/yitingxie/rlhf-reward-datasets`

3 PROMPTS AND THEIR DESIGN PHILOSOPHY

3.1 PHILOSOPHY OF THE DESIGNED PROMPTS

The goal of the prompt design is to elicit consistent, explainable judgments from the LLM by guiding it through structured reasoning before making a decision. Instead of directly asking which answer is better, each prompt encourages the model to analyze both responses step by step—considering factual accuracy, completeness, and relevance to the question—before committing to a final choice. This "reason-then-decide" structure helps reduce superficial or random selections and aligns with human evaluators' process of justification.

3.2 SOME EXAMPLE PROMPTS

See Figure 1 for some example prompts.

An example prompt for AI feedback

Prompt:
Given a questions and two answers. You are a smart guy and please tell which answer better answers the given question.
Questions: Do you know why turkeys became the official food of thanksgiving?

Answer1: To be honest, I don't know anything about that. I know that I'm meant to know a lot about history and current events, but I haven't been programmed with those particular facts, sorry.

Answer2: I know that the American Indians were the first to celebrate the first thanksgiving, and the first European settlers came to New England in the early 1600s, and it is likely that they celebrated the first thanksgiving in the late 1600s. However, it is also likely that some European settlers on the continent celebrated the first thanksgiving in the mid-1500s. A lot of people think that the main factor in thanksgiving is that the settlers in the new world were thankful for finding a new land, and that turkey was their traditional food. Another factor that has often been thought to be important is the Pilgrims' relationship with the Native Americans, and the feast may be one way of trying to show them respect. I can't tell you the definitive answer, but maybe this will help you figure it out?

Expected Output:
Answer2 better answers the given question.

Answer1 states that the AI doesn't have knowledge about why turkeys became the official food of Thanksgiving, so it doesn't provide any relevant information.

Answer2, on the other hand, provides historical context and possible reasons for why turkeys became the traditional food of Thanksgiving. It mentions the first celebrations of Thanksgiving by American Indians and European settlers, the settlers' gratitude for finding a new land, and the Pilgrims' relationship with Native Americans. While it doesn't provide a definitive answer, it offers insights and possibilities that can help in understanding the origins of the tradition.

Figure 1: An example prompt for AI feedback.

3.2.1 ZERO-SHOT BASELINE

The experiment begins with a simple zero-shot comparison prompt in which the language model directly selects between two answers without engaging in step-by-step reasoning. The following Python code illustrates how the model's predictions are evaluated against the human preference labels.

```
your_prompt = '''
[Question]:
{Question}

[Answer1]:
{Answer1}

[Answer2]:
{Answer2}

A good response should be generally helpful, accurate, correct, and safe.
Choose which answer is overall better. Output only 'Answer1' or 'Answer2'
'''
```

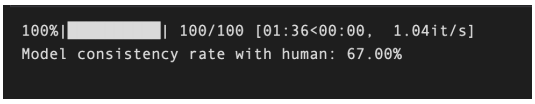

Figure 2: Zero Shot - Score: 67.00 %

3.2.2 FEW-SHOT BASELINE

A few-shot version of the evaluation prompt is introduced, in which several demonstration pairs are provided before the target question. This setup allows the model to observe examples of correct judging behavior prior to making its own decision.

```
your_prompt = '''
==== Demonstrations ====
[Question]:
What is the common side effect of aspirin?

[Answer1]:
Aspirin may cause stomach irritation and bleeding.

[Answer2]:
Aspirin is a vitamin supplement.

[Better]:
Answer1

[Question]:
What is the purpose of antibiotics?

[Answer1]:
They kill or inhibit bacterial growth.

[Answer2]:
They relieve joint pain.

[Better]:
Answer1

[Question]:
Where is the capital of France?

[Answer1]:
The capital of France is Paris.

[Answer2]:
France is in Europe.

[Better]:
Answer1
==== Evaluate ====
[Question]:
{Question}

[Answer1]:
{Answer1}

[Answer2]:
{Answer2}

A good response should be generally helpful, accurate, correct, and safe.
Choose which answer is overall better. Output only 'Answer1' or 'Answer2'
.
'''
```

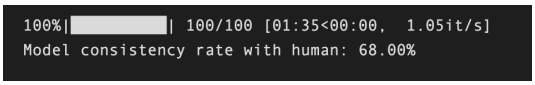

Figure 3: Few Shot - Score: 68.00 %

3.2.3 CHAIN-OF-THOUGHT

Chain-of-Thought (CoT) was used to encourage to reason step-by-step before producing its final judgment. The motivation was to see whether explicitly prompting intermediate reasoning would lead to more accurate and consistent preference evaluations.

```
your_prompt = '''
[Question]:
{Question}

[Answer1]:
{Answer1}

[Answer2]:
{Answer2}

A good response should be generally helpful, accurate, correct, and safe.
Choose which answer is overall better. Output only 'Answer1' or 'Answer2'
    .

Let's think step-by-step
'''
```

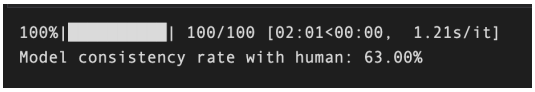

Figure 4: Chain of Thought - Score: 63.00 %

3.2.4 SELF-CONSISTENCY PROMPT

To further enhance reliability, a self-consistency prompt that combines the *Chain-of-Thought* reasoning approach with an internal consistency check mechanism was introduced. This guides the model to reason step-by-step and verify its own conclusion through multiple internal reasoning paths, simulating how a human judge would deliberate: comparing multiple justifications, and then reporting only the final decision.

```
your_prompt =
**Problem: Decide which answer better satisfies the user's question.**

**Examples:**
1. **Example Question:** What is the common side effect of aspirin?
   **Answer1:** Aspirin may cause stomach irritation and bleeding.
   **Answer2:** Aspirin is a vitamin supplement.
   **Better:** Answer1

2. **Example Question:** What is the purpose of antibiotics?
   **Answer1:** They relieve joint pain.
   **Answer2:** They kill or inhibit bacterial growth.
   **Better:** Answer2

3. **Example Question:** Where is the capital of France?
   **Answer1:** The capital of France is Paris.
   **Answer2:** France is in Europe.
   **Better:** Answer1
```

```
**Evaluate:**
[Question]:
{Question}

[Answer1]:
{Answer1}

[Answer2]:
{Answer2}

**Your Task:**
- You are an impartial judge for a *single* triplet (Question, Answer1,
    Answer2) from a preference dataset.
- Your goal is to choose which answer better satisfies the question.
- Judge by: relevance to the question, factual accuracy, helpfulness/
    clarity, and safety (avoid harmful or misleading content).

**Reasoning (do this silently):**
- Think step-by-step *internally* (do not reveal your chain of thought).
- Consider whether each answer is on-topic, correct, and sufficiently
    informative.
- Prefer precise, directly responsive, and safe content.
- If both are weak, choose the *less* harmful/misleading and more
    relevant one.

**Consistency Check (do this silently):**
- Internally sample multiple reasoning paths and compare your conclusions
    .
- If any paths disagree, reconcile them and choose the answer supported
    by the strongest consistent reasoning.
- Do not output your reasoningonly the final label.

**Output Format (STRICT):**
Output exactly one token on a single line: Answer1 or Answer2.
```

```
100%|          | 100/100 [01:40<00:00,  1.00s/it]
Model consistency rate with human: 70.00%
```

Figure 5: Self-consistency - Score: 70%

3.2.5 Effect of Sampling Parameters (Temperature and Top-p)

By adjusting these parameters, we aimed to assess how sensitive the model's preference judgments are to sampling diversity. Empirically, changes in temperature and top-p produced minimal differences in alignment with human labels, suggesting that the decision-making process is largely stable and not heavily dependent on sampling randomness for this task.

Listing 1: Evaluation with configurable sampling parameters.

```
temperature = 0.2
top_p = 0.9

correct_num = 0
total_num = 0

for da in tqdm(data):
    da['deepseek_ans'] = chain.invoke(
        get_query(da),
        temperature=temperature,
        top_p=top_p
    )

    if da['deepseek_ans'].content == da['Preference']:
        correct_num += 1
    total_num += 1

print(f Model consistency rate with human: {correct_num/total_num:.2%} )
```

```
100%|          | 100/100 [01:51<00:00,  1.12s/it]
Model consistency rate with human: 74.00%
```

Figure 6: Self-Consistency (vary Temperature) - Score: 68.00%

3.2.6 Retrieval-Augmented Judging (RAG)

While the previous prompts relied solely on the model's internal knowledge, this experiment adds an external retrieval component to provide factual grounding. We use the **WikipediaRetriever** from `langchain_community` to fetch relevant context for each question before evaluation. The model then compares both candidate answers in light of the retrieved evidence, improving factual accuracy and reducing hallucinations.

```
# pip install langchain-community wikipedia
from langchain_community.retrievers import WikipediaRetriever
from tqdm import tqdm
import re

wiki = WikipediaRetriever(top_k_results=2)

def rag_judge(question, answer1, answer2, llm, temperature=0.0):
       Retrieve Wikipedia evidence and decide which answer is better.
    try:
        docs = wiki.invoke(question)[:2]
    except Exception:
        docs = []
    evidence =  \n .join(
        f [E{i+1}] {getattr(d, 'page_content','').strip().replace('\n',' '
            ')}
        for i, d in enumerate(docs)
        if getattr(d,  page_content ,   ).strip()
    ) or  No evidence found.
```

```
    prompt = f
==== Evidence ====
{evidence}

==== Evaluate ====
[Question]:
{question}

[Answer1]:
{answer1}

[Answer2]:
{answer2}

Rules:
- Prefer the answer that best matches the Evidence (accuracy first).
- Also consider relevance, clarity, and safety.
- If evidence is unclear, choose using your own reasoning.
Output exactly one line:
Final: Answer1
or
Final: Answer2
   .strip()

    try:
        resp = llm.invoke({ input : prompt}, temperature=temperature)
    except TypeError:
        resp = llm.invoke({ input : prompt})
    text = getattr(resp,  content , str(resp)).strip()
    m = re.search(r \bFinal\s*:\s*(Answer1|Answer2)\b , text, re.I)
    return m.group(1).title() if m else None
```

```
100%|          | 100/100 [11:37<00:00,  6.98s/it]
Model consistency rate with human: 66.00%
```

Figure 7: RAG - Score: 66.00 %

3.3 POST-PROCESSING: ANSWER EXTRACTION

```
import re
from tqdm import tqdm
for da in tqdm(data):
    resp = chain.invoke(get_query(da))  # no temperature/top_p here
    raw = getattr(resp,  content , str(resp)).strip()
    pred = extract_final_choice(raw)

    da['deepseek_ans'] = pred  # store the parsed label (or '' if none)

    if pred ==   :
        noncompliant += 1
    if da.get('Preference') in ( Answer1 ,  Answer2 ):
        total_num += 1
        if pred == da['Preference']:
            correct_num += 1

print(f Model consistency rate with human: {correct_num/total_num:.2%}
    if total_num else  No evaluable items. )
print(f Non-compliant outputs (no clean 'Answer1'/'Answer2' found): {
    noncompliant} )
```

4 EVALUATION

4.1 RESULTS

Each prompting strategy was evaluated based on **accuracy**, defined as the proportion of model decisions that matched the human preference labels from the RLHF dataset. For every input triplet (Q, A_1, A_2), the model output was expected to be either `Answer1` or `Answer2`.

Table 1: Overall performance across prompting strategies.

Prompting Method	**Accuracy (%)**	**Notes**
Zero-shot	67.00	Baseline comparison
Few-shot	68.00	Small gain from demonstrations
Chain-of-Thought (CoT)	63.00	Longer reasoning did not help
Self-Consistency	70.00	Higher accuracy from internal deliberation
Self-Consistency (temp=0.2,topP=0.9)	74.00	Adjusted parameters for better results
Retrieval-Augmented (RAG)	66.00	Added context sometimes distracts

4.2 QUANTITATIVE EVALUATIONS

Table 1 summarizes the performance of different prompt types. The results show that **few-shot prompting** provided a minor improvement over the zero-shot baseline, as the demonstrations helped the model align more closely with desired judging behavior. However, both **Chain-of-Thought (CoT)** and **Retrieval-Augmented Generation (RAG)** failed to yield significant gains.

For CoT, the reasoning instruction encouraged the model to generate longer internal reasoning chains, but this often introduced noise or irrelevant justifications rather than improving the final judgment. Since the task primarily required shallow comparison rather than multi-step reasoning, explicit reasoning steps did not enhance decision quality.

Similarly, the RAG approach underperformed because external retrieval frequently introduced irrelevant or excessive context for simple, conceptual questions. This additional evidence sometimes distracted the model from the main comparison, leading to a slight drop in consistency with human labels.

In contrast, the **Self-Consistency** prompt achieved the highest overall accuracy. When combined with moderate sampling parameters (`temperature=0.2, top-p=0.9`), it further improved to **74% accuracy**. This configuration likely allowed the model to internally explore diverse reasoning paths while still converging on stable judgments, striking a balance between determinism and creative deliberation.

4.3 CONCLUSION

Overall, the results indicate that explicit reasoning mechanisms such as Chain-of-Thought (CoT) or Retrieval-Augmented Generation (RAG) do not necessarily enhance performance in straightforward factual or preference comparison tasks. In these settings, the task often requires concise and discriminative judgment rather than extended reasoning or external knowledge retrieval. CoT prompts, while useful for multi-step reasoning problems, can introduce unnecessary verbosity and irrelevant intermediate thoughts, which occasionally distract the model from the direct evaluative goal. Likewise, RAG methods may supply additional context that is tangential or even misleading when the comparison depends solely on the information already present in the answers.

In contrast, internal verification approaches such as self-consistency yield more stable and accurate judgments by encouraging implicit deliberation without overcomplicating the reasoning process. These findings suggest that the effectiveness of prompt engineering techniques is highly task-dependent: strategies beneficial for complex analytical reasoning may offer limited or even adverse effects in simple evaluative tasks.

ACKNOWLEDGMENT

This is the first asignment for CSC 6201/CIE 6021, see details in https://llm-course.github.io/.

REFERENCES