Role-based vs Chain-of-Thought AI Comparison App

1. Objective

The goal of this project is to compare AI outputs generated using role-based prompts versus chain-of-thought (CoT) prompts.

This helps users learn how different prompt strategies affect the clarity, reasoning, and level of explanation in AI-generated responses.

2. Instructions & Implementation

Step 1: Select a Task

- Example task: *Explain how photosynthesis works* or any other concept requiring explanation.
- The task should be suitable for observing differences between **role-based explanations** and **step-by-step reasoning**.

Step 2: Role-based Prompting

Definition: In role-based prompting, the model is asked to assume a specific role (e.g., teacher, expert, student) and generate output according to that role.

Prompt Example:

Implementation (Hugging Face Model):

- Model: google/flan-t5-large (default)
- Task: text2text-generation
- Parameters: max length, temperature, top-p
- Output Example: (Model generates a simplified teacher-friendly explanation.)

Step 3: Chain-of-Thought (CoT) Prompting

Definition: In chain-of-thought prompting, the model is asked to reason step by step before giving the final answer. This improves transparency and detail.

Prompt Example:

vbnet

Explain photosynthesis step by step, reasoning each step clearly.

Implementation (Hugging Face Model):

- Model: google/flan-t5-base (default)
- Task: text2text-generation
- Parameters: max length, temperature, top-p
- Output Example: (Model generates a detailed stepwise explanation.)

Step 4: Compare Outputs

- The app generates both outputs side by side.
- Diff and reflection tools highlight:
 - o Differences in phrasing
 - o Reasoning depth
 - o Similarities or divergences in content

3. Observations

- Role-based output is concise, simple, and oriented to the assumed role.
- CoT output provides stepwise reasoning, enhancing interpretability.
- Outputs may vary in **detail**, **clarity**, and **structure** depending on the prompt style.
- CoT is particularly useful for complex or multi-step tasks.

4. Deliverables

- Streamlit App:
 - a. Interactive interface to enter prompts and run both models.
 - b. Displays outputs, diffs, and reflections in real time.

```
₱ T1_A2.py ×

                                                                  ▷ ~ ⑤ Ⅲ …
                                                                                          T1_A2.py X
                                                                                                                                                                                                 ☑ py + ~ 日 🛍 … | 🖸 ×
                                                                                                                                                                                                                                   Σ
                                                                                            T1_A2.py >
                                                                                            34
35
                                                                                                    def reflection(role_out: str, cot_out: str) -> s
                                                                                                                                                                                          Local URL: http://localhost:8
                                                                                                        r = difflib.SequenceMatcher(None, role_out, if r > 0.8: sim = "Outputs are very similar. elif r > 0.4: sim = "Outputs differ moderate
                                                                                                                                                                                       502
        Streamlit App: Individual Inputs for Role-Based Generates meaningful, justified outputs & create
                                                                                                                                                                                       Network URL: http://192.168.2
9.107:8502
                                                                                            36
37
38
39
40
41
42
43
44
45
46
47
48
50
51
52
53
54
55
56
57
58
                                                                                                         else: sim = "Outputs diverge strongly."
return f"Role-based explains as a teacher, Co
        import io, difflib
from datetime import datetime
                                                                                                    def build_pdf(entries: List[Dict[str, Any]], tit
                                                                                                        buf = io.BytesIO()
doc = SimpleDocTemplate(buf, pagesize=letter
styles = getSampleStyleSheet()
        from typing import Dict, Any, List import streamlit as st
        from transformers import pipeline
from reportlab.platypus import SimpleDocTemplate
from reportlab.lib.pagesizes import letter
                                                                                                         14
15
16
17
18
19
20
21
22
23
24
25
26
27
        from reportlab.lib import colors
from reportlab.lib.styles import getSampleStyleS
                                                                                                          ("BACKGROUND",(0,0)
("FONTSIZE",(0,0),(
end(tbl); doc.build(elems); buf.see
                                                                                            59
60
                                                                                                    # ------ Streamlit App ------
st.set_page_config("Role vs CoT (Individual Inpu
st.title(" ** Role-based vs ** Chain-of-Thought ·
 29
30
         62
                                                                                                    st.sidebar.header("Model Settings")
```

Fig1: Input Snapshot

- PDF Report: Automatically generated with:
 - User inputs (role-based and CoT prompts)
 - Model outputs
 - Diff table highlighting differences
 - Reflections and similarity scores

5. Output

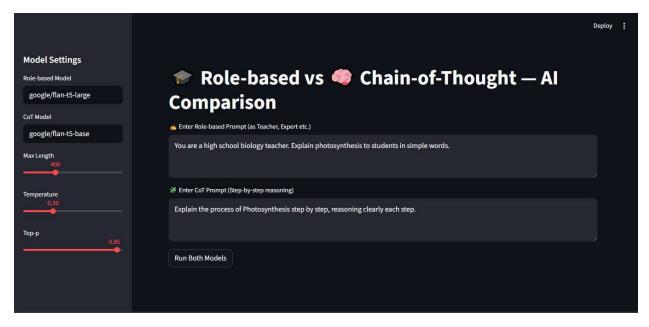


Fig2: Output Snapshot1

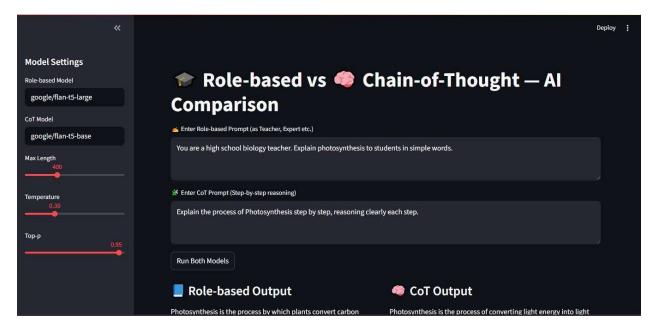


Fig3: Output Snapshot2

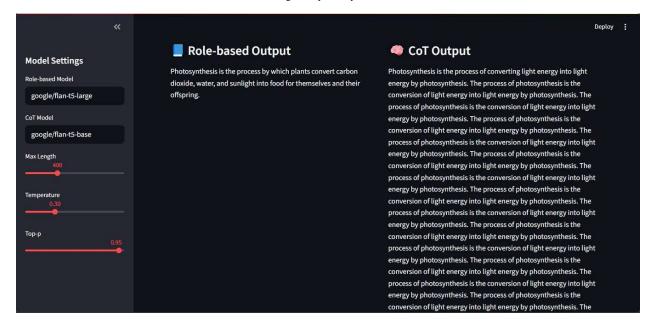


Fig4: Output Snapshot3

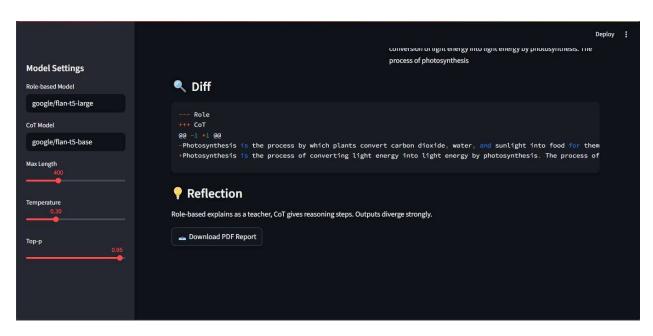


Fig4: Output Snapshot3