PAL (Program-Aided Language Models)

Gao et al., (2022) presents a method that uses LLMs to read natural language problems and generate programs as the intermediate reasoning steps. Coined, program-aided language models (PAL), it differs from chain-of-thought prompting in that instead of using free-form text to obtain solution it offloads the solution step to a programmatic runtime such as a Python interpreter.

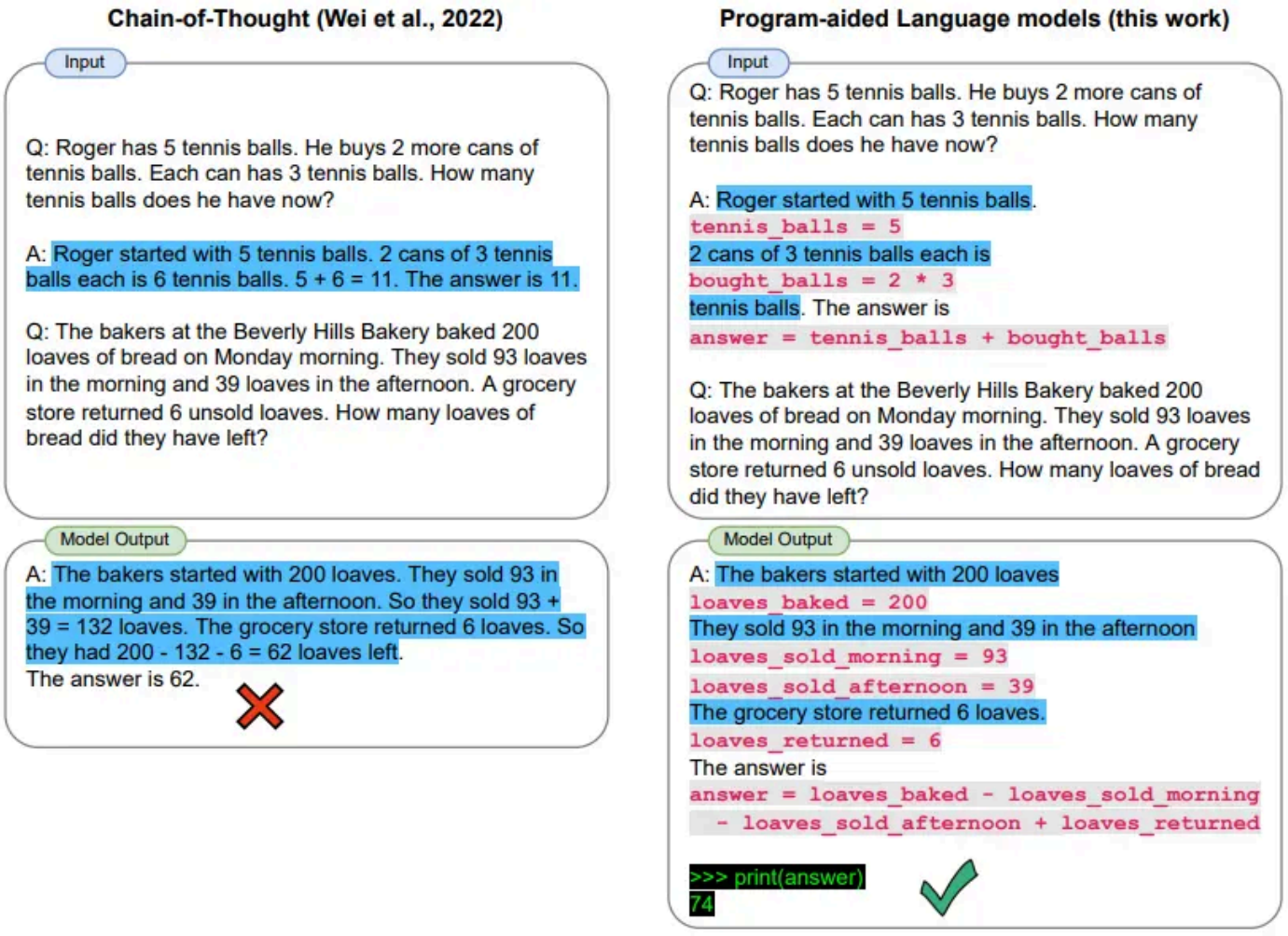

Image Source: Gao et al., (2022)

Let's look at an example using LangChain and OpenAI GPT-3. We are interested to develop a simple application that's able to interpret the question being asked and provide an answer by leveraging the Python interpreter.

Specifically, we are interested to create a functionality that allows the use of the LLM to answer questions that require date understanding. We will provide the LLM a prompt that includes a few exemplars which are adopted from here.

These are the imports we need:

```
import openai
from datetime import datetime
from dateutil.relativedelta import relativedelta
import os
from langchain.llms import OpenAI
from dotenv import load_dotenv
```

Let's first configure a few things:

```
load_dotenv()

# API configuration
openai.api_key = os.getenv("OPENAI_API_KEY")

# for LangChain
os.environ["OPENAI_API_KEY"] = os.getenv("OPENAI_API_KEY")
```

Setup model instance:

```
llm = OpenAI(model_name='text-davinci-003', temperature=0)
```

Setup prompt + question:

```
question = "Today is 27 February 2023. I was born exactly 25 years ago. What is the date I
was born in MM/DD/YYYY?"

DATE_UNDERSTANDING_PROMPT = """
# Q: 2015 is coming in 36 hours. What is the date one week from today in MM/DD/YYYY?
# If 2015 is coming in 36 hours, then today is 36 hours before.
today = datetime(2015, 1, 1) - relativedelta(hours=36)
# One week from today,
one_week_from_today = today + relativedelta(weeks=1)
# The answer formatted with %m/%d/%Y is
one_week_from_today.strftime('%m/%d/%Y')
# Q: The first day of 2019 is a Tuesday, and today is the first Monday of 2019. What is the
date today in MM/DD/YYYY?
# If the first day of 2019 is a Tuesday, and today is the first Monday of 2019, then today is
6 days later.
today = datetime(2019, 1, 1) + relativedelta(days=6)
# The answer formatted with %m/%d/%Y is
today.strftime('%m/%d/%Y')
# Q: The concert was scheduled to be on 06/01/1943, but was delayed by one day to today. What
is the date 10 days ago in MM/DD/YYYY?
# If the concert was scheduled to be on 06/01/1943, but was delayed by one day to today, then
today is one day later.
```

```
today = datetime(1943, 6, 1) + relativedelta(days=1)
# 10 days ago,
ten_days_ago = today - relativedelta(days=10)
# The answer formatted with %m/%d/%Y is
ten_days_ago.strftime('%m/%d/%Y')
# Q: It is 4/19/1969 today. What is the date 24 hours later in MM/DD/YYYY?
# It is 4/19/1969 today.
today = datetime(1969, 4, 19)
# 24 hours later,
later = today + relativedelta(hours=24)
# The answer formatted with %m/%d/%Y is
today.strftime('%m/%d/%Y')
# Q: Jane thought today is 3/11/2002, but today is in fact Mar 12, which is 1 day later. What is the date 24 hours later in MM/DD/YYYY?
# If Jane thought today is 3/11/2002, but today is in fact Mar 12, then today is 3/12/2002.
today = datetime(2002, 3, 12)
# 24 hours later,
later = today + relativedelta(hours=24)
# The answer formatted with %m/%d/%Y is
later.strftime('%m/%d/%Y')
# Q: Jane was born on the last day of Feburary in 2001. Today is her 16-year-old birthday. What is the date yesterday in MM/DD/YYYY?
# If Jane was born on the last day of Feburary in 2001 and today is her 16-year-old birthday, then today is 16 years later.
today = datetime(2001, 2, 28) + relativedelta(years=16)
# Yesterday,
yesterday = today - relativedelta(days=1)
# The answer formatted with %m/%d/%Y is
yesterday.strftime('%m/%d/%Y')
# Q: {question}
""".strip() + '\n'
```

```
llm_out = llm(DATE_UNDERSTANDING_PROMPT.format(question=question))
print(llm_out)
```

This will output the following:

```
# If today is 27 February 2023 and I was born exactly 25 years ago, then I was born 25 years before.
today = datetime(2023, 2, 27)
# I was born 25 years before,
born = today - relativedelta(years=25)
# The answer formatted with %m/%d/%Y is
born.strftime('%m/%d/%Y')
```

The contents of `llm_out` are a Python code snippet. Below, the `exec` command is used to execute this Python code snippet.

```
exec(llm_out)
```

```
print(born)
```

This will output the following: `02/27/1998`

Last updated on September 19, 2024