AI-Assistant Data Science

CSC 405/605/705

UNC GREENSBORO

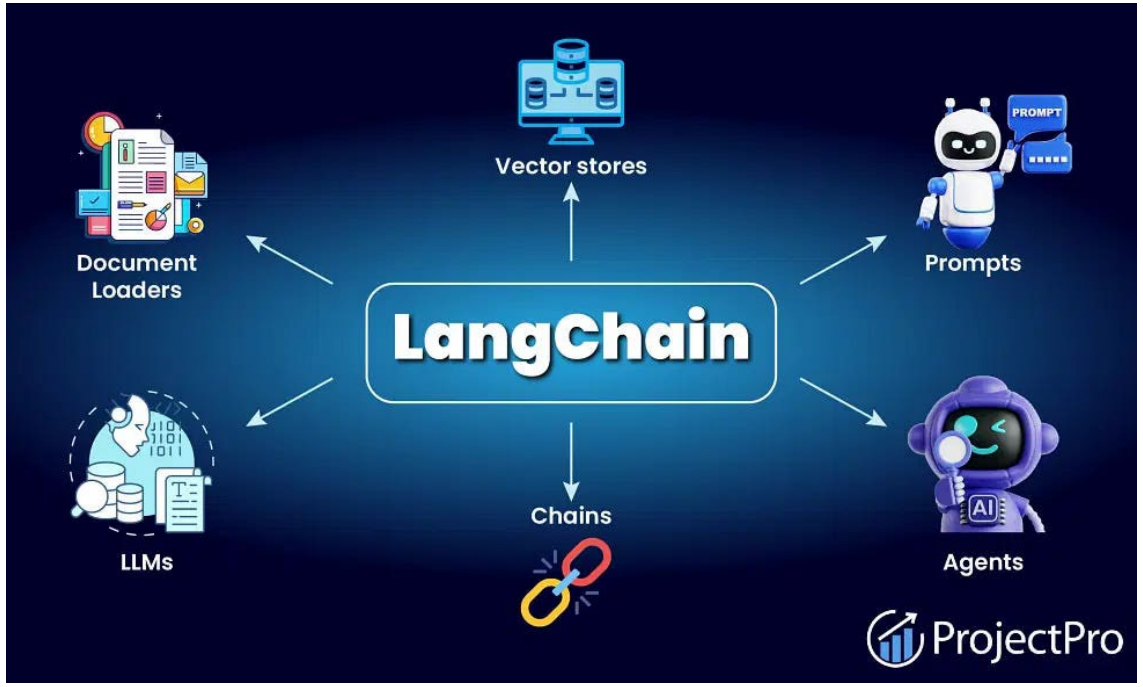
➢ Introduction of LangChain



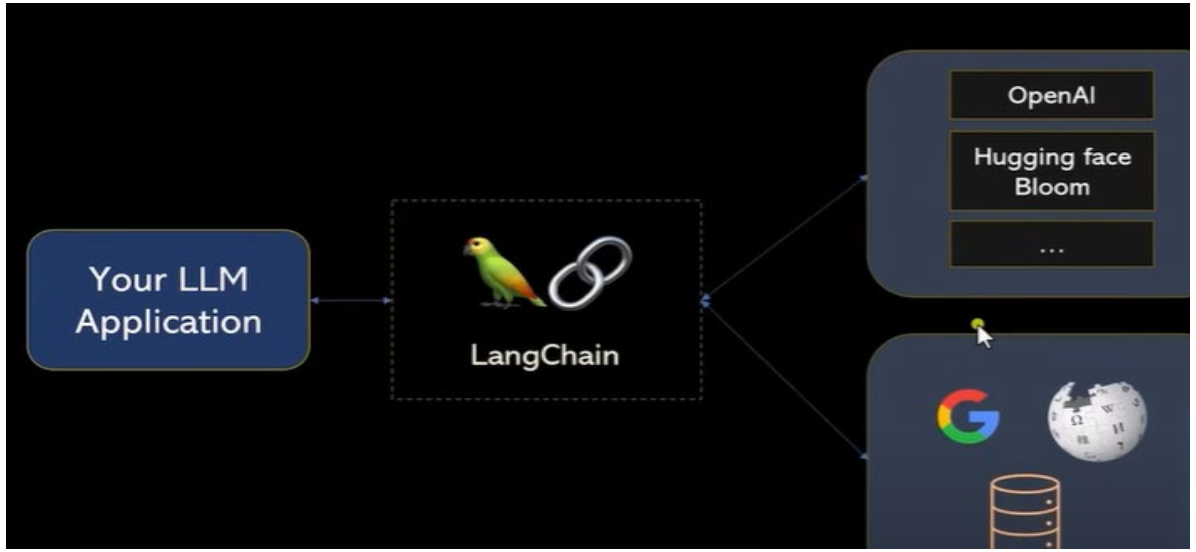
UNC GREENSBORO

https://docs.langchain.com/oss/python/langchain/overview

What is Lang Chain?

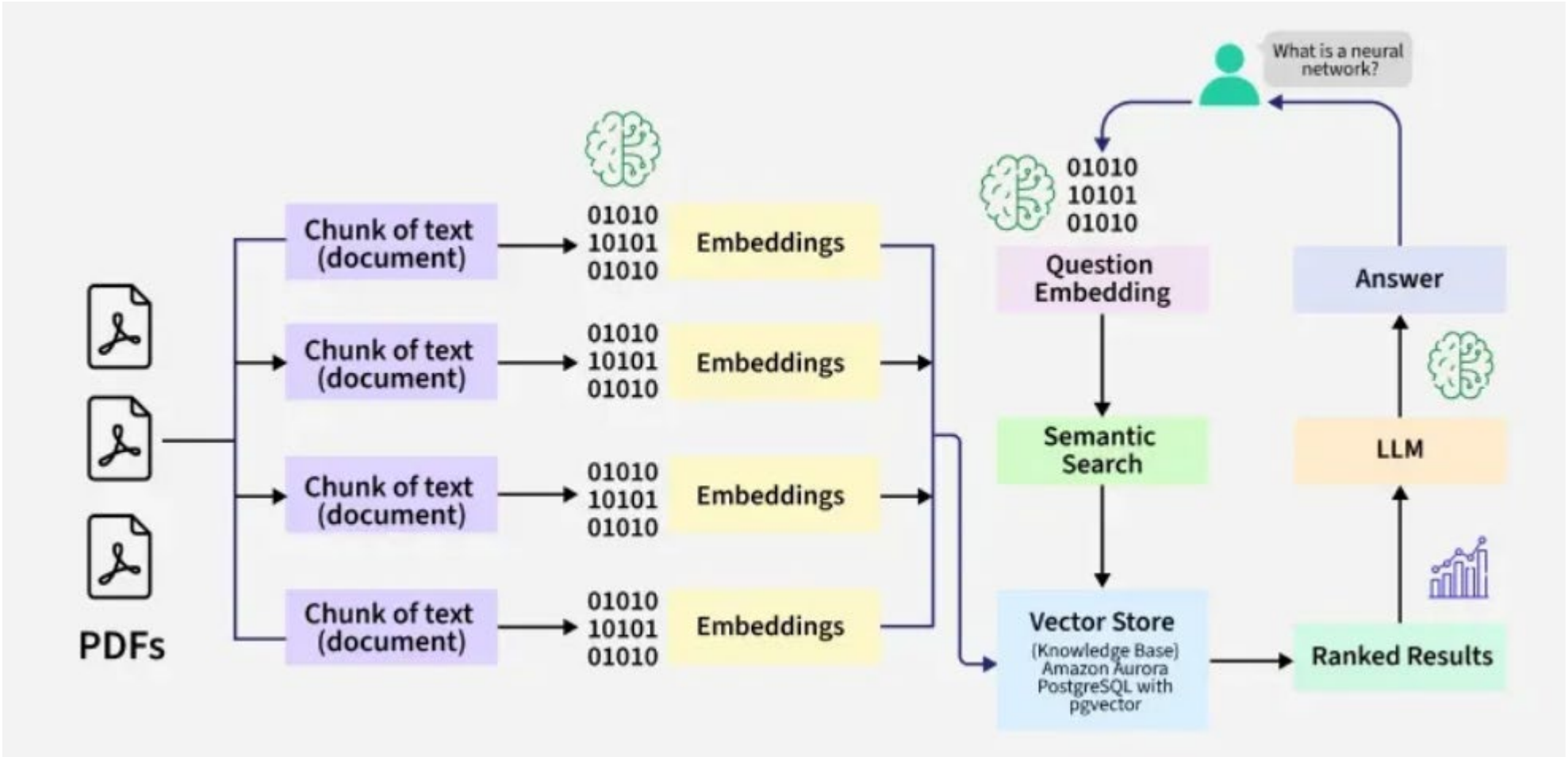


- Lang Chain is an open-source framework that makes it easier to build applications powered by LLMs, like GPT and Gemini.



UNC GREENSBORO

Example of RAG use case: QA over unstructured data



https://www.geeksforgeeks.org/artificial-intelligence/introduction-to-langchain/

Install Lang Chain

```
pip install langchain
```

Installing Model Integrations: The base langchain package does not include specific AI provider libraries. You must install independent packages depending on the Large Language Model (LLM) you plan to use:

- OpenAI: pip install langchain-openai
- Anthropic (Claude): pip install langchain-anthropicGoogle
- Gemini: pip install langchain-google-genai
- Ollama (Local Models): pip install langchain-ollama
- Hugging Face: pip install langchain-huggingface



UNC GREENSBORO

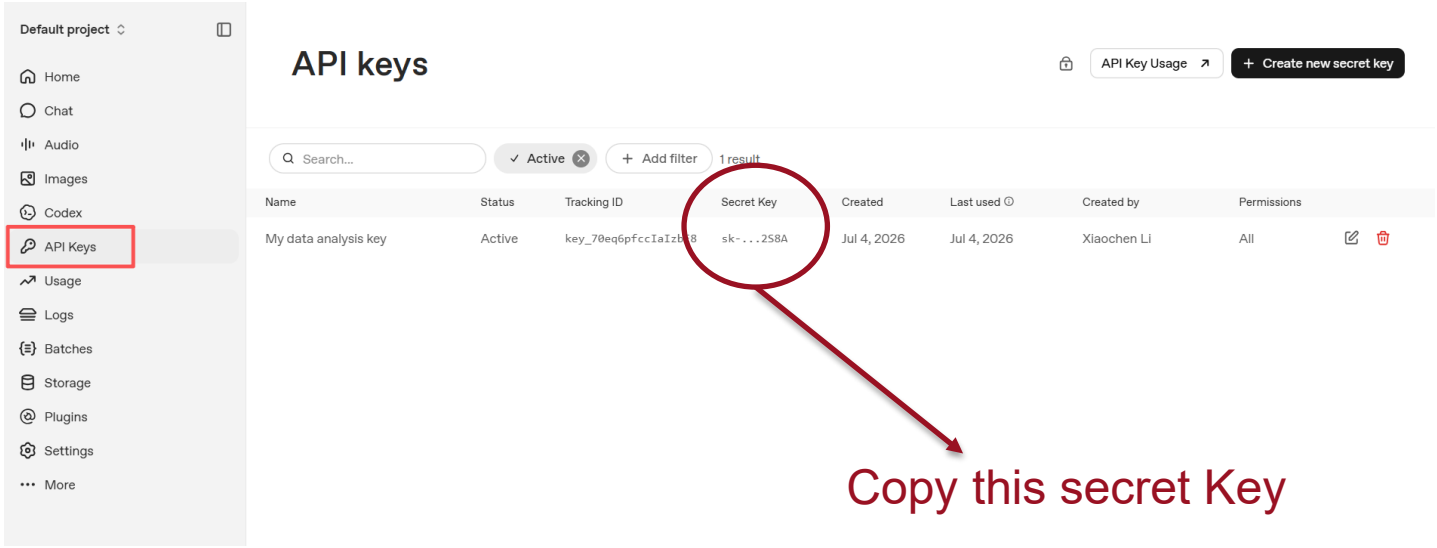
https://docs.langchain.com/oss/python/langchain/install

Setup LLMs

https://platform.openai.com/home

Example: Setup OpenAI

```
pip install langchain-openai
setx OPENAI_API_KEY “xxx"
```




```
from langchain_openai import ChatOpenAI

llm = ChatOpenAI(
model="gpt-4o-mini",
%api_key=os.getenv("OPENAI_API_KEY")
)
```

UNC GREENSBORO

Setup LLMs

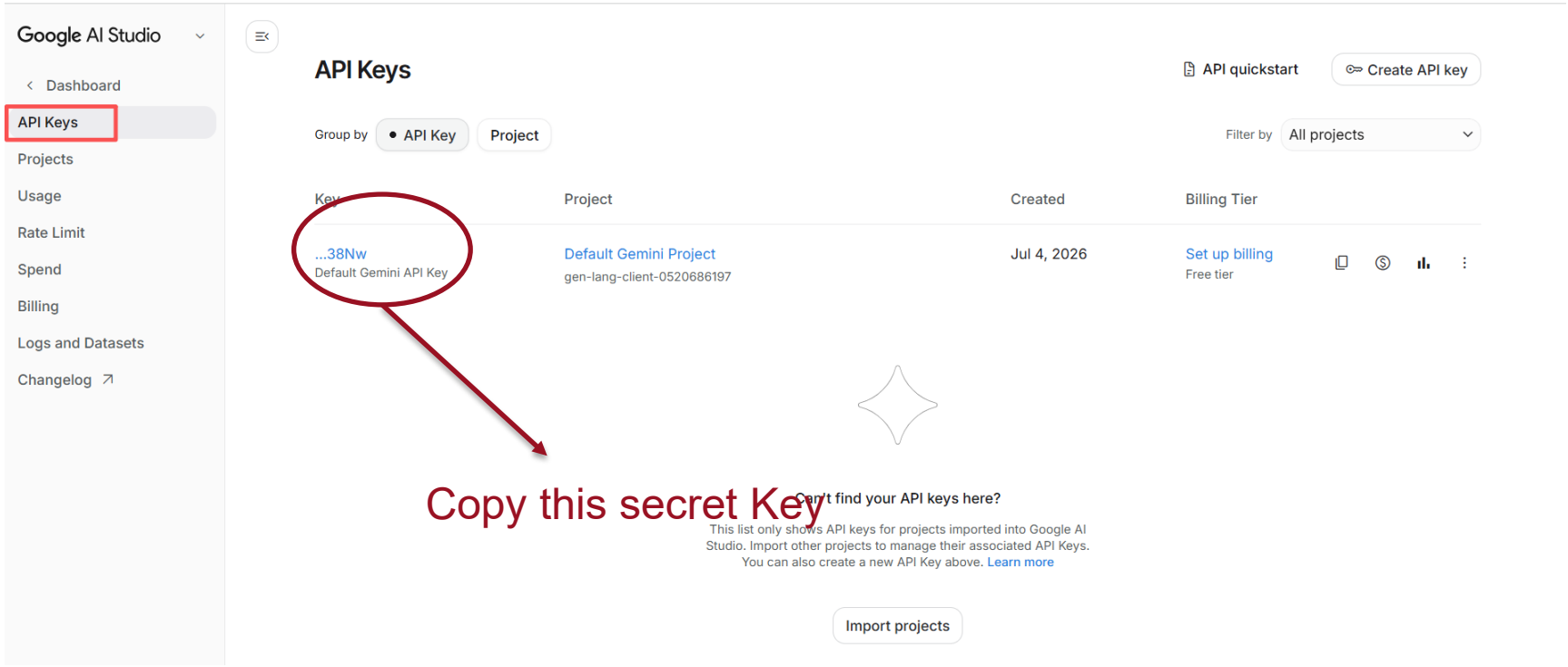
Example: Setup Goole Gemini

```
pip install langchain-google-genai
setx GOOGLE_API_KEY “xxx"
```

```
from langchain_google_genai import
ChatGoogleGenerativeAI

llm = ChatGoogleGenerativeAI(
model="gemini-2.5-flash",
temperature=0
)
```

https://aistudio.google.com/



Copy this secret Key

UNC GREENSBORO

Setup LLMs

Example: Setup Goole Gemini

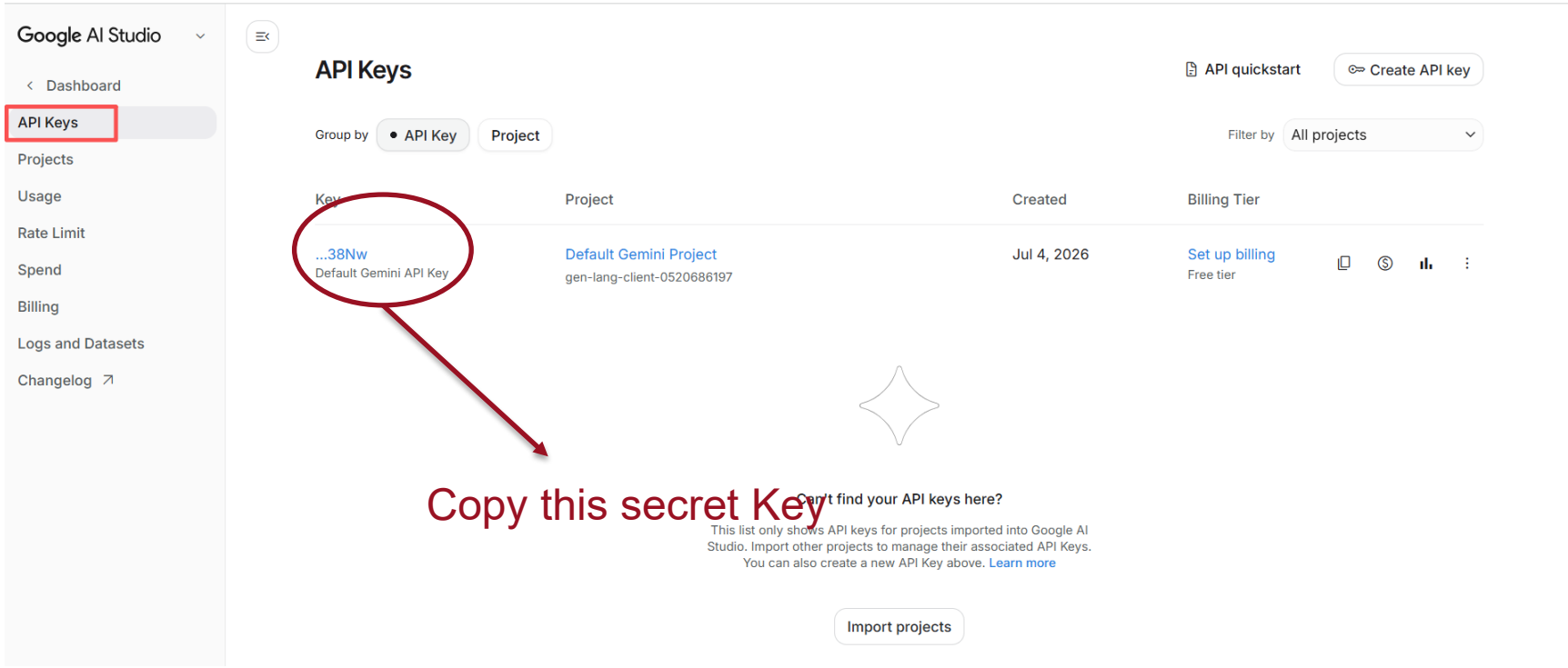
```
pip install langchain-google-genai
setx GOOGLE_API_KEY “xxx"
```

```
from langchain_google_genai import
ChatGoogleGenerativeAI

llm = ChatGoogleGenerativeAI(
model="gemini-2.5-flash",
temperature=0
)
```

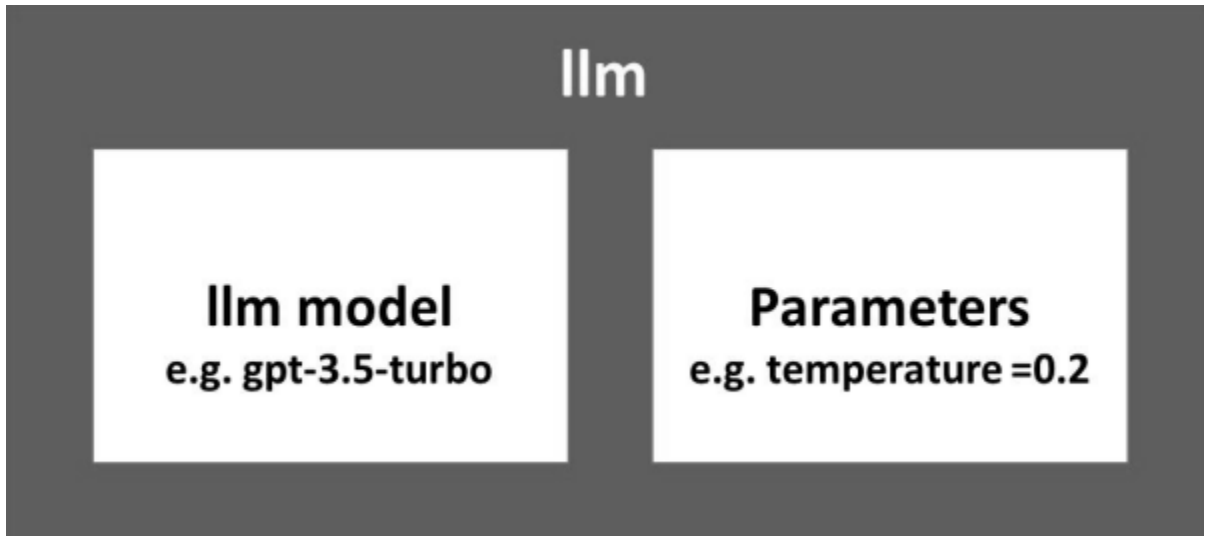
Controls creativity
(0 = more deterministic, 1 = more creative)

https://aistudio.google.com/



Copy this secret Key

Model, Prompts & Parsers

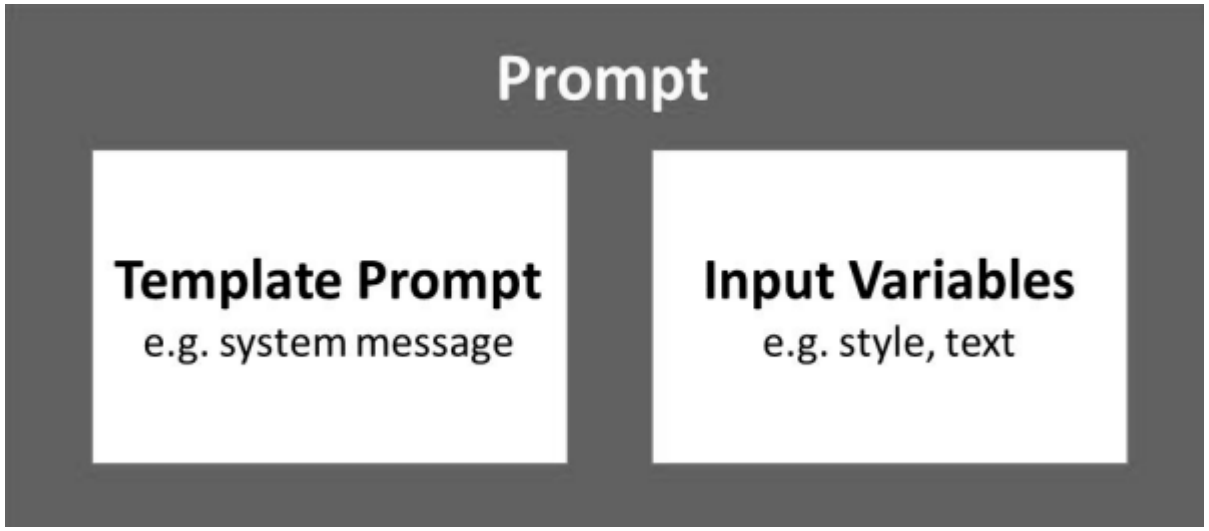


+ API-key as environment

```python
from langchain_google_genai import ChatGoogleGenerativeAI

llm = ChatGoogleGenerativeAI(model="gemini-2.5-flash",temperature=0.7)
```

Model, Prompts & Parsers




```python
# Simple Prompt

prompt = "Suggest me a skill that is required as a Data Scientist"
response = llm.invoke(prompt)
print("Suggested Skill:\n", response.content)
```

UNC GREENSBORO

Model, Prompts & Parsers

```
PS C:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science>  & 'c:\Users\X_LI12\AppData\Local\Microsoft\WindowsApps\python3.13.exe' 'c:\Users\X_LI12\.vscode\extensions\ms-python.debugpy-2026.6.0-win32-x64\bundled\libs\debugpy\launcher' '53368' '--' 'c:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science\basic_langchain.py'
Both GOOGLE_API_KEY and GEMINI_API_KEY are set. Using GOOGLE_API_KEY.
Suggested Skill:
 While there isn't *one single* skill that defines a Data Scientist (it's a multidisciplinary role), if I had to pick one foundational skill that underpins almost everything else, it would be **Statistical Thinking and Machine Learning Fundamentals**.

Let me elaborate why, and then provide a more comprehensive list of other critical skills:

---

### The Core Skill: **Statistical Thinking & Machine Learning Fundamentals**

**Why it's crucial:**

1.  **Understanding Data:** Statistics provides the framework for understanding data distributions, variability, relationships, and uncertainty. Without it, you can't properly explore, summarize, or interpret your data.
2.  **Informing Decisions:** It enables you to make informed decisions about data quality, feature engineering, model selection, and hypothesis testing.
3.  **Building & Evaluating Models:** Machine Learning algorithms are deeply rooted in statistical principles. Understanding these fundamentals allows you to:
    *   Choose the right algorithm for a given problem.
    *   Tune hyperparameters effectively.
    *   Interpret model outputs (e.g., coefficients, feature importance).
    *   Properly evaluate model performance (e.g., understanding bias-variance trade-off, overfitting, underfitting, and metrics like precision, recall, RMSE, AUC).
    *   Design and interpret experiments (A/B testing).
4.  **Communicating Insights:** You can't effectively communicate the "why" behind your findings or the reliability of your predictions without a solid grasp of statistical concepts.

---

### Other Essential Skills for a Data Scientist:

While the above is foundational, a successful Data Scientist needs a blend of technical, analytical, and soft skills:

**I. Technical Skills:**

1.  **Programming (Python / R):**
    *   **Python:** Dominant for data manipulation (Pandas, NumPy), machine learning (Scikit-learn, TensorFlow, PyTorch), and deployment.
    *   **R:** Strong for statistical analysis, visualization (Tidyverse, ggplot2), and academic research.
2.  **SQL (Structured Query Language):** Essential for extracting, manipulating, and managing data from relational databases. Most companies store their data in SQL databases.
3.  **Data Wrangling & Preprocessing:** The ability to clean, transform, and prepare messy real-world data for analysis and modeling. This often takes up 60-80% of a data scientist's time.
4.  **Data Visualization & Storytelling:**
    *   **Tools:** Matplotlib, Seaborn, Plotly (Python); ggplot2 (R); Tableau, Power BI (BI tools).
```

Model, Prompts & Parsers

```python
from langchain_core.prompts import PromptTemplate

template = "Give me one skill that are required as a {occupation}. Please provide a concise response."
prompt_template = PromptTemplate.from_template(template)

filled_prompt = prompt_template.format(occupation = "Data Scientist")

response = llm.invoke(filled_prompt)
print("Suggested Skill:\n", response.content)
```

```
PS C:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science>  c:; cd 'c:\User
extensions\ms-python.debugpy-2026.6.0-win32-x64\bundled\libs\debugpy\launcher' '61545
Both GOOGLE_API_KEY and GEMINI_API_KEY are set. Using GOOGLE_API_KEY.
Suggested Skill:
 Statistical Modeling.
```

UNC GREENSBORO

Model, Prompts & Parsers

Parser	Output	Use Case
StrOutputParser	str	Normal chatbot responses
JsonOutputParser	dict	JSON output
PydanticOutputParser	Pydantic object	Structured and validated output

Model, Prompts & Parsers

```python
prompt = "Suggest me three skill that is required as a Data Scientist"
llm_response = llm.invoke(prompt)

out_parser = StrOutputParser()

parsed_response = out_parser.parse(llm_response.content)

print(parsed_response)
```

StrOutputParser

```
As a Data Scientist, you bridge the gap between raw data and actionable business insights. This requires a blend of technical prowess, analytical thinking, and effective communication. Here are three core skills:

1.  **Programming (Python/R & SQL):**
    *   **Why it's crucial:** This is your primary toolset for nearly every task. You need to programmatically extract, clean, transform, analyze, and model data.
    *   **Python/R:** These languages are the industry standard for data manipulation (Pandas, dplyr), numerical computing (NumPy), machine learning (Scikit-learn, TensorFlow, PyTorch, Caret), data visualization (Matplotlib, Seaborn, ggplot2), and statistical analysis.
    *   **SQL (Structured Query Language):** Data often resides in databases. SQL is essential for querying, retrieving, and manipulating data from relational databases, which is usually the first step in any data science project.

2.  **Statistics & Machine Learning Fundamentals:**
    *   **Why it's crucial:** This is the theoretical backbone that allows you to understand data, build predictive models, and draw valid conclusions. Without a strong grasp here, you're just running code blindly.
    *   **Statistics:** Understanding concepts like descriptive statistics, probability distributions, hypothesis testing, regression analysis, correlation, and sampling is vital for data exploration, experimental design, and interpreting model results.
    *   **Machine Learning:** You need to understand various ML algorithms (e.g., linear regression, logistic regression, decision trees, random forests, boosting, clustering, neural networks), when to apply them, how to evaluate their performance (e.g., accuracy, precision, recall, F1-score, ROC curves), and techniques like cross-validation and feature engineering.

3.  **Business Acumen & Communication:**
    *   **Why it's crucial:** Technical skills alone aren't enough. A data scientist must understand the business context to frame problems correctly, translate complex technical findings into understandable insights for non-technical stakeholders, and influence decision-making.
    *   **Business Acumen:** This involves understanding the industry, the company's goals, key performance indicators (KPIs), and how your data science project can create value. It helps in defining the right problem to solve and interpreting results in a meaningful way for the business.
    *   **Communication:** You must be able to clearly articulate your methodology, present your findings (often through data visualization and storytelling), highlight the implications for the business, and defend your recommendations to diverse audiences (from executives to fellow engineers). This includes written reports, presentations, and verbal explanations.
PS C:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science>
```

Model, Prompts & Parsers

```python
parser = JsonOutputParser()

prompt = """
Suggest me three skill that is required as a Data Scientist.

Return your answer in JSON:

{
  "summary": "...",
  "skill_list": [...],
  "description": [...]
}
"""

response = llm.invoke(prompt)

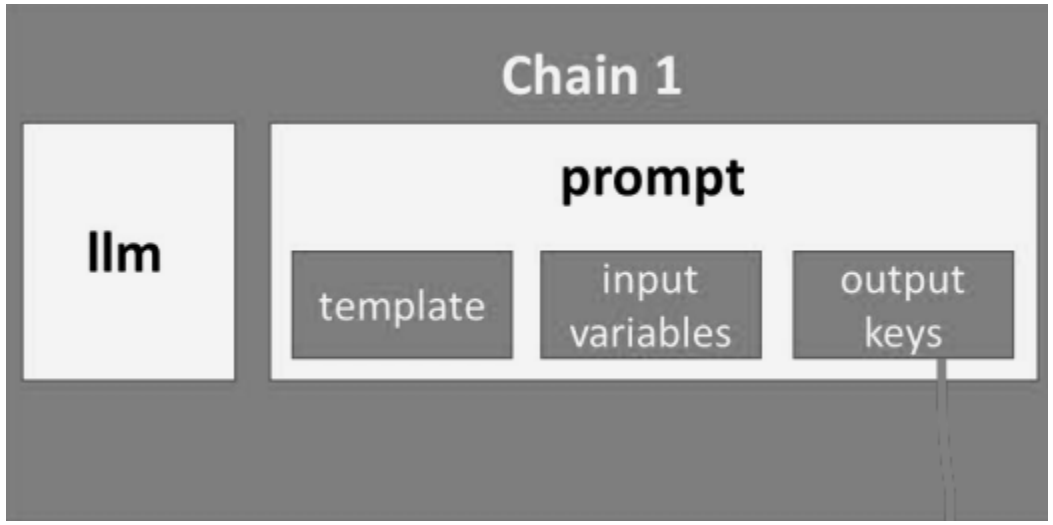
result = parser.parse(response.content)

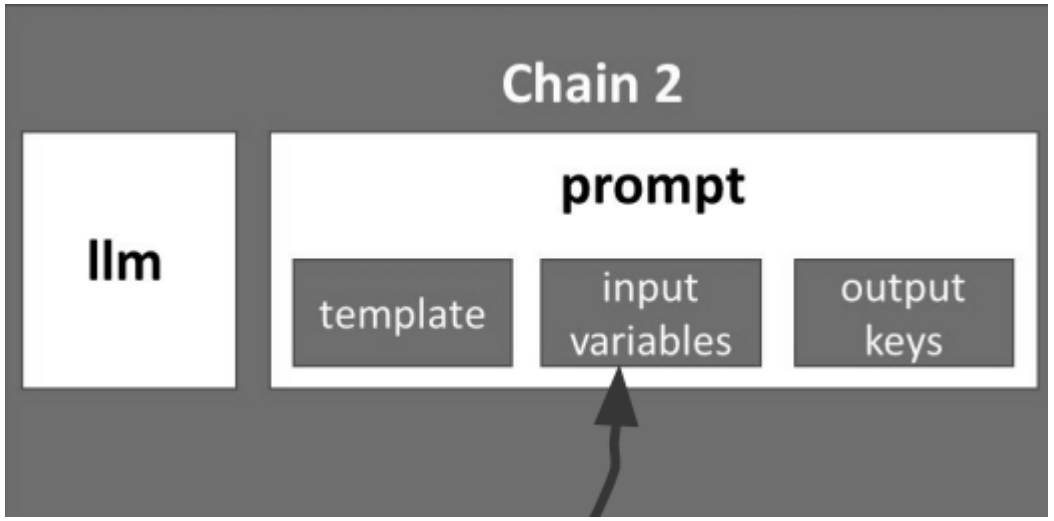
print(result)
```

JsonOutputParser

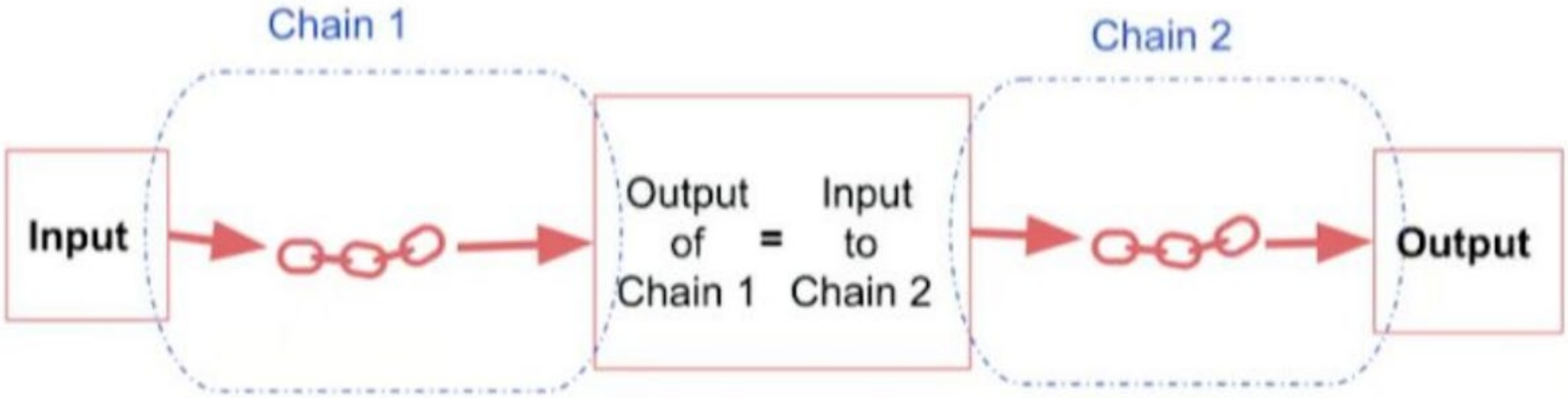
```
{'summary': 'These three skills form the technical bedrock for any aspiring or practicing Data Scientist, enabling them to extract insights, build predictive models, and drive data-informed decisions.', 'skill_list': ['Programming Proficiency (Python/R)', 'Statistical Foundations', 'Machine Learning Expertise'], 'description': ['Data scientists heavily rely on programming languages like Python or R for data cleaning, transformation, analysis, statistical modeling, machine learning algorithm implementation, and automation. Proficiency includes understanding data structures, algorithms, and key libraries (e.g., Pandas, NumPy, Scikit-learn in Python; dplyr, ggplot2 in R).', 'A strong grasp of statistics and probability is crucial for understanding data distributions, hypothesis testing, experimental design, regression analysis, and interpreting the significance and reliability of model results. It helps in making robust data-driven decisions and avoiding misinterpretations.', "This involves understanding various machine learning algorithms (supervised, unsupervised, reinforcement learning), knowing when and how to apply them, performing feature engineering, evaluating model performance, and understanding deployment strategies. It's the core of building predictive and prescriptive solutions."]}
PS C:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science>
```

Chains

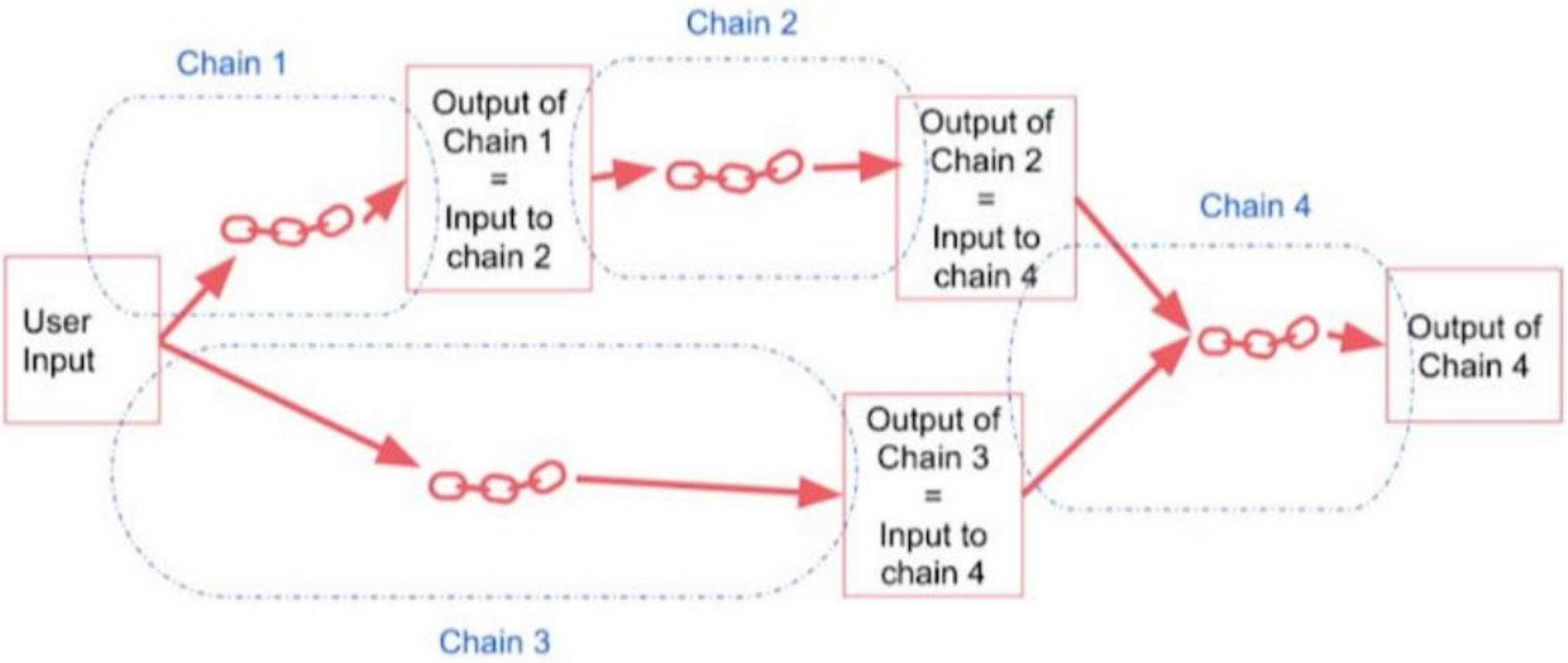

Chain 1
llm
prompt
template
input variables
output keys


Chain 2
llm
prompt
template
input variables
output keys

Chains


Chain 1
Chain 2
Input
Output of Chain 1 = Input to Chain 2
Output

Chains


Chain 1
Chain 2
Chain 3
Chain 4
User Input
Output of Chain 1 = Input to chain 2
Output of Chain 2 = Input to chain 4
Output of Chain 3 = Input to chain 4
Output of Chain 4

Chains

Without LangChain
Chain and Parse

```
# Step 1: Translate to English
translate_prompt = PromptTemplate.from_template(
    "Translate the following review into English:\n\n{review}"
)

translate_response = llm.invoke(
    translate_prompt.format(review=review)
)

english_review = translate_response.content

print("English Review:")
print(english_review)

# Step 2: Summarize
summary_prompt = PromptTemplate.from_template(
    "Summarize the following review in one sentence:\n\n{review}"
)

summary_response = llm.invoke(
    summary_prompt.format(review=english_review)
)

summary = summary_response.content

print("\nSummary:")
print(summary)
```

Chains

```
Beneath its sci-fi exterior, the show's true core lies in the most unvarnished, heartfelt emotional experiences of ordinary people: a child's unwavering attachment to their mother, a teenager's regret over life choices, a lover's greedy desi
re to freeze a moment of affection, and an elder's serene acceptance of aging and farewell. On this wondrous land, the rules of the world are gently bent: time no longer flows unidirectionally; past and future can overlap and collide; two te
enagers accidentally swap souls in a machine, completely rewriting their lives; a single bracelet can freeze time, allowing lovers to savor an eternal moment; and two versions of oneself meet across parallel dimensions, offering each other s
alvation. Within this cyclical loop, virtually no miracle is impossible.

But is this truly the case? "Nothing is impossible" ultimately proves to be nothing more than a gentle self-consolation. "The Loop" may bend space-time and conjure wondrous encounters, but it lacks the power to undo regrets or alter the past
. We can travel through time to reunite with deceased loved ones; we can swap identities to experience another life; we can permanently seal moments of profound emotion – yet we cannot erase the void left by separation, mend fractured intima
te relationships, or wipe away all past mistakes and pain. After circling through fate's loop, we still must accept inevitable loss, understand that what is broken cannot be fully restored, and that time cannot be reversed.

All the sci-fi elements are merely a shell; what "Tales from the Loop" ultimately presents are the most ordinary human emotions: separation, growth, longing, and reconciliation. Within this aesthetic of beautiful sadness, in this stark and h
azy town, we repeatedly loop through experiences, finally realizing that life's regrets are irreversible. To complete life's loop, carrying our imperfections, is the most gentle answer "The Loop" offers its audience.

Summary:
"Tales from the Loop" is a melancholic sci-fi drama that, despite its poetic and fantastical bending of reality, ultimately explores profound human emotions and the irreversible nature of loss and regret within life's cyclical journey.
PS C:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science>
```

UNC GREENSBORO

Chains

With LangChain Chain and Parse

```
# Prompt 1
translate_prompt = PromptTemplate.from_template(
    "Translate the following review into English:\n\n{review}"
)

# Prompt 2
summary_prompt = PromptTemplate.from_template(
    "Summarize the following review in one sentence:\n\n{review}"
)

parser = StrOutputParser()

# Build the first chain
translation_chain = translate_prompt | llm | parser

# Build the second chain
summary_chain = summary_prompt | llm | parser

# Execute sequentially
english_review = translation_chain.invoke(
    {"review": review}
)

summary = summary_chain.invoke(
    {"review": english_review}
)
```

Chain 1

Chain 2

UNC GREENSBORO

Chains

```
PS C:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science>  c:; cd 'c:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science'; & 'c:\Users\X_LI12\AppData\Local\Microsoft\WindowsApps\python3.13.exe' 'c:\Users\X_LI12\.vscode\
extensions\ms-python.debugpy-2026.6.0-win32-x64\bundled\libs\debugpy\launcher' '49972' '--' 'c:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science\basic_langchain.py'
 moment of affection, an elder's serene acceptance of aging and departure... In this extraordinary land, the established rules of the world are gently bent: time ceases its unidirectional flow, allowing past and future to overlap and collide
; two teenagers stumble into a machine, swapping souls and irrevocably altering their lives; a simple bracelet can freeze time, granting lovers an exclusive eternity; and parallel selves can meet, offering each other a chance at redemption.
Within the cyclical nature of the Loop, it seems as though no miracle is truly impossible.

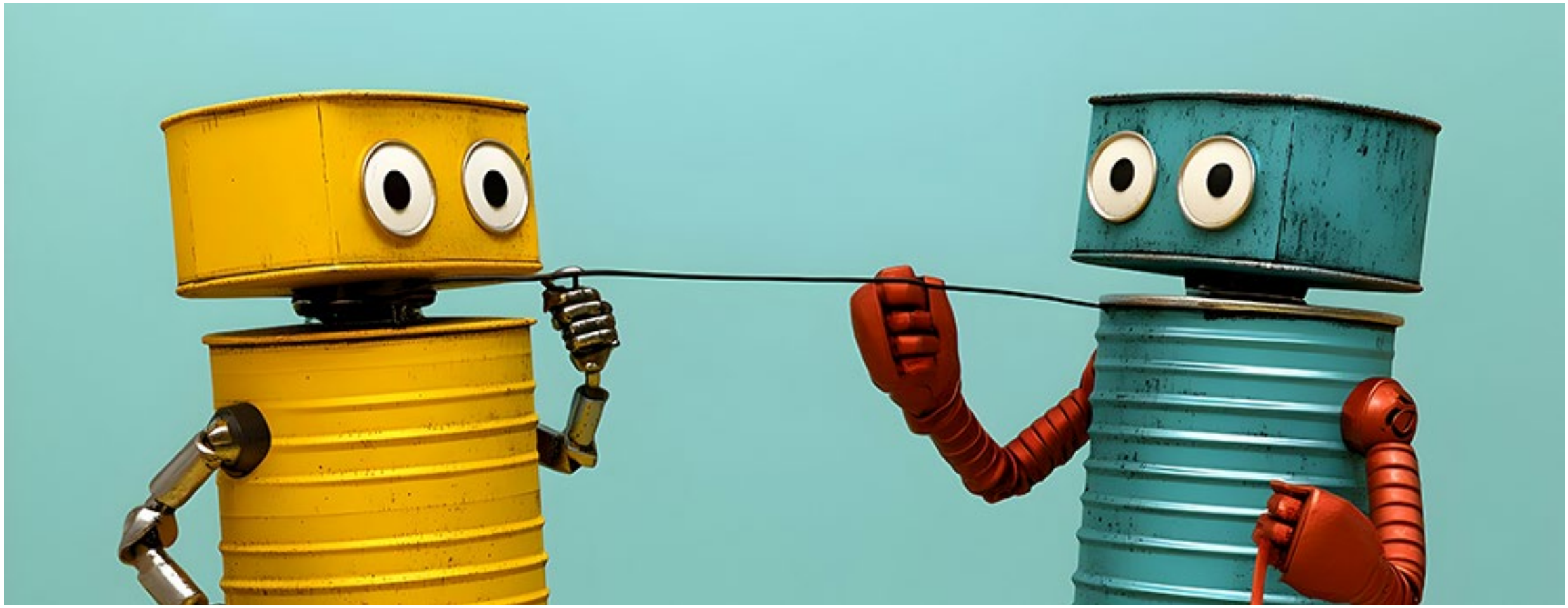
But is this truly the case? The notion of "nothing impossible" ultimately proves to be merely a tender form of self-consolation. "The Loop" may indeed warp space-time and conjure extraordinary encounters, yet it lacks the power to reverse re
gret or alter the past. We might traverse time to reunite with departed loved ones; we might swap identities to live another life; we might even permanently preserve cherished moments of affection. Yet, we remain powerless to mend the void l
eft by farewells, to repair fractured intimate relationships, or to erase all past mistakes and hurts. Even after traversing the full circle of destiny, we are still compelled to accept inevitable losses, to understand that brokenness cannot
 be fully restored, and that time cannot flow backward.

All the sci-fi elements merely serve as a shell. What "Tales from the Loop" ultimately reveals are the most common human emotions: separation, growth, longing, and reconciliation. Within its melancholic beauty, amidst the stark and misty bac
kdrop of the small town, we repeatedly experience these loops, eventually coming to terms with the irreversible nature of life's regrets. To complete life's loop, carrying our imperfections, is the gentlest truth "Tales from the Loop" offers
 its audience.

Summary:
"Tales from the Loop" is a visually stunning sci-fi drama that, beneath its fantastical veneer, gently explores profound human emotions and the melancholic acceptance of life's irreversible regrets.
PS C:\Users\X_LI12\Documents\data_science\AI-assistant_Data_Science>
```

UNC GREENSBORO

Let’s build a data science report with LLMs!



UNC GREENSBORO