Install Required Packages

```
! pip install -q --upgrade google-generativeai langchain-google-genai python-dotenv
```

```
  ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 50.4/50.4 kB 1.5 MB/s eta 0:00:00
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 399.9/399.9 kB 10.2 MB/s eta 0:00:00
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 290.2/290.2 kB 15.4 MB/s eta 0:00:00
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 76.4/76.4 kB 4.0 MB/s eta 0:00:00
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 77.9/77.9 kB 4.6 MB/s eta 0:00:00
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 141.9/141.9 kB 8.4 MB/s eta 0:00:00
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 58.3/58.3 kB 2.9 MB/s eta 0:00:00
```

```
#///// prompt: create a .env file in the workspace
# https://makersuite.google.com/

!echo -e 'AIzaSyAc3ma3dyWjQzyVygkoMasnLBtcEuz-jlk' > .env
```

```
!ls -a
```

```
.  ..  .config  .env  sample_data
```

```
from dotenv import load_dotenv
load_dotenv()
```

```
True
```

```
from IPython.display import display
from IPython.display import Markdown
import textwrap


def to_markdown(text):
  text = text.replace('•', '  *')
  return Markdown(textwrap.indent(text, '> ', predicate=lambda _: True))
```

```
import google.generativeai as genai
import os
```

```
import os
genai.configure(api_key= "AIzaSyAc3ma3dyWjQzyVygkoMasnLBtcEuz-jlk")
```

Text Generation

```
model = genai.GenerativeModel(model_name = "gemini-pro")
model
```

```
genai.GenerativeModel(
    model_name='models/gemini-pro',
    generation_config={},
    safety_settings={},
    tools=None,
    system_instruction=None,
    cached_content=None
)
```

```
prompt = [
 "What is Mixture of Experts?",
]

response = model.generate_content(prompt)
to_markdown(response.text)
```

Mixture of Experts (MoE)

Definition:

Mixture of Experts is a machine learning algorithm that combines multiple specialized experts (sub-models) to improve the accuracy and performance of a model.

How it Works:

MoE divides the input data into multiple subsets or "gates." Each subset is then processed by a different expert, which is a specialized model trained to handle a specific aspect of the problem.

The outputs of the experts are then combined using a gating network. The gating network learns to assign weights to each expert, determining the contribution of each expert to the final prediction.

Benefits:

- **Improved accuracy:** By combining multiple experts, MoE can capture different patterns and relationships in the data, resulting in more accurate predictions.
- **Efficient training:** MoE trains the individual experts separately, reducing the computational cost of training the entire model.
- **Flexibility:** MoE allows for the use of different types of experts, including linear models, decision trees, and neural networks.

Applications:

MoE has been successfully applied to various machine learning tasks, including:

- Image classification
- Natural language processing
- Speech recognition
- Time series forecasting
- Recommendation systems

Why it's Called "Mixture of Experts":

- The gating network resembles a mixture model, which combines multiple probability distributions to represent a complex distribution.
- Each expert is an expert in a specific aspect of the problem, representing different areas of expertise.

Variations:

- **Temporal Mixture of Experts (TMoE):** Uses a recurrent neural network as the gating network to process sequential data.
- **Bayesian Mixture of Experts (BMoE):** Uses Bayesian inference to combine the outputs of the experts.
- **Deep Mixture of Experts (DMoE):** Uses deep neural networks as both the experts and the gating network.

```
to_markdown(response.text)
```

Mixture of Experts (MoE)

Definition:

Mixture of Experts is a machine learning algorithm that combines multiple specialized experts (sub-models) to improve the accuracy and performance of a model.

How it Works:

MoE divides the input data into multiple subsets or "gates." Each subset is then processed by a different expert, which is a specialized model trained to handle a specific aspect of the problem.

The outputs of the experts are then combined using a gating network. The gating network learns to assign weights to each expert, determining the contribution of each expert to the final prediction.

Benefits:

- **Improved accuracy:** By combining multiple experts, MoE can capture different patterns and relationships in the data, resulting in more accurate predictions.
- **Efficient training:** MoE trains the individual experts separately, reducing the computational cost of training the entire model.
- **Flexibility:** MoE allows for the use of different types of experts, including linear models, decision trees, and neural networks.

Applications:

MoE has been successfully applied to various machine learning tasks, including:

- Image classification
- Natural language processing
- Speech recognition
- Time series forecasting
- Recommendation systems

Why it's Called "Mixture of Experts":

- The gating network resembles a mixture model, which combines multiple probability distributions to represent a complex distribution.
- Each expert is an expert in a specific aspect of the problem, representing different areas of expertise.

Variations:

- **Temporal Mixture of Experts (TMoE):** Uses a recurrent neural network as the gating network to process sequential data.
- **Bayesian Mixture of Experts (BMoE):** Uses Bayesian inference to combine the outputs of the experts.
- **Deep Mixture of Experts (DMoE):** Uses deep neural networks as both the experts and the gating network.

```
# response.prompt_feedback
# response.candidates
```

Use LangChain to Access Gemini API

```
from langchain_google_genai import ChatGoogleGenerativeAI
```

```
llm = ChatGoogleGenerativeAI(model="gemini-pro", google_api_key="AIzaSyDW_n54TMnahHUp9pMmKu0SjvhujCkSB7I")
```

```
result = llm.invoke("What is the sky blue ?")
```

```
to_markdown(result.content)
```

The sky is blue because of a phenomenon called Rayleigh scattering. This occurs when sunlight, which is made up of all the colors of the visible spectrum, passes through the Earth's atmosphere. The atmosphere is made up of molecules of nitrogen and oxygen, which are much smaller than the wavelength of light. When sunlight strikes these molecules, it causes them to vibrate and scatter the light in all directions. The shorter wavelengths of light, such as blue and violet, are scattered more than the longer wavelengths, such as red and orange. This is because the shorter wavelengths have a higher frequency and therefore interact more with the molecules in the atmosphere. As a result, the sky appears blue to our eyes.

Chat

```
model = genai.GenerativeModel("gemini-1.5-flash")
chat = model.start_chat(history=[])
chat
```

```
ChatSession(
    model=genai.GenerativeModel(
        model_name='models/gemini-1.5-flash',
        generation_config={},
        safety_settings={},
        tools=None,
        system_instruction=None,
        cached_content=None
    ),
    history=[]
)
```

```
response = chat.send_message(
    "In one sentence, explain how a computer works to a young child."
)
to_markdown(response.text)
```

A computer is like a super smart friend that follows instructions you give it using buttons and words, and it can do lots of things like show pictures, play games, and talk to other computers!

```
chat.history
```

```
[parts {
   text: "In one sentence, explain how a computer works to a young child."
 }
 role: "user",
 parts {
   text: "A computer is like a super smart friend that follows instructions you give it using buttons and words, and it can do lots of things like show pictures, play games, and talk to other computers! \n"
 }
 role: "model"]
```

```
response = chat.send_message(
    "Okay, how about a more detailed explanation to a high schooler?", stream=True
)

for chunk in response:
    print(chunk.text)
    print("_" * 80)
```

```
A computer
________________________________________________________________________________
 operates by receiving instructions in the form of code, which it then translates into a series
________________________________________________________________________________
 of electrical signals that manipulate data stored in its memory, processing it to produce outputs
________________________________________________________________________________
 like text, images, or sounds.


________________________________________________________________________________
```

```
for message in chat.history:
    display(to_markdown(f"**{message.role}**: {message.parts[0].text}"))
```

user: In one sentence, explain how a computer works to a young child.

model: A computer is like a super smart friend that follows instructions you give it using buttons and words, and it can do lots of things like show pictures, play games, and talk to other computers!

user: Okay, how about a more detailed explanation to a high schooler?

model: A computer operates by receiving instructions in the form of code, which it then translates into a series of electrical signals that manipulate data stored in its memory, processing it to produce outputs like text, images, or sounds.

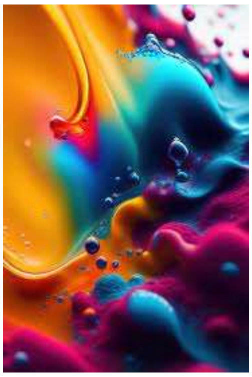
Image & Text Generation

```
!curl -o image.jpg "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQ3SUjbtVb-_8zi-hO-nITKEqRir3ha6wa3D6R3C5Izi3lte1rq2ch1ZB1t86D
```

```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  8012  100  8012    0     0   104k      0 --:--:-- --:--:-- --:--:--  105k
```

```
import PIL.Image

img = PIL.Image.open("image.jpg")
img
```



```
model = genai.GenerativeModel("gemini-1.5-flash")
```

```
response = model.generate_content(img)

to_markdown(response.text)
```

This is a beautiful image of colorful liquid swirling and mixing together. The vibrant colors and the abstract patterns created by the mixing liquids create a visually appealing and dynamic image. The small bubbles scattered throughout the image add a sense of depth and movement. The image could be interpreted in many different ways, depending on the viewer's perspective. For example, some might see it as a representation of the beauty and chaos of nature, while others might see it as a metaphor for the complexities of human emotions. Regardless of its interpretation, this is a truly captivating image.

```
response = model.generate_content(
    [
        "Write a short, engaging blog post based on this picture. It should include a description of the meal in the photo and talk about
        img,
    ],
    stream=True,
)
response.resolve()
```

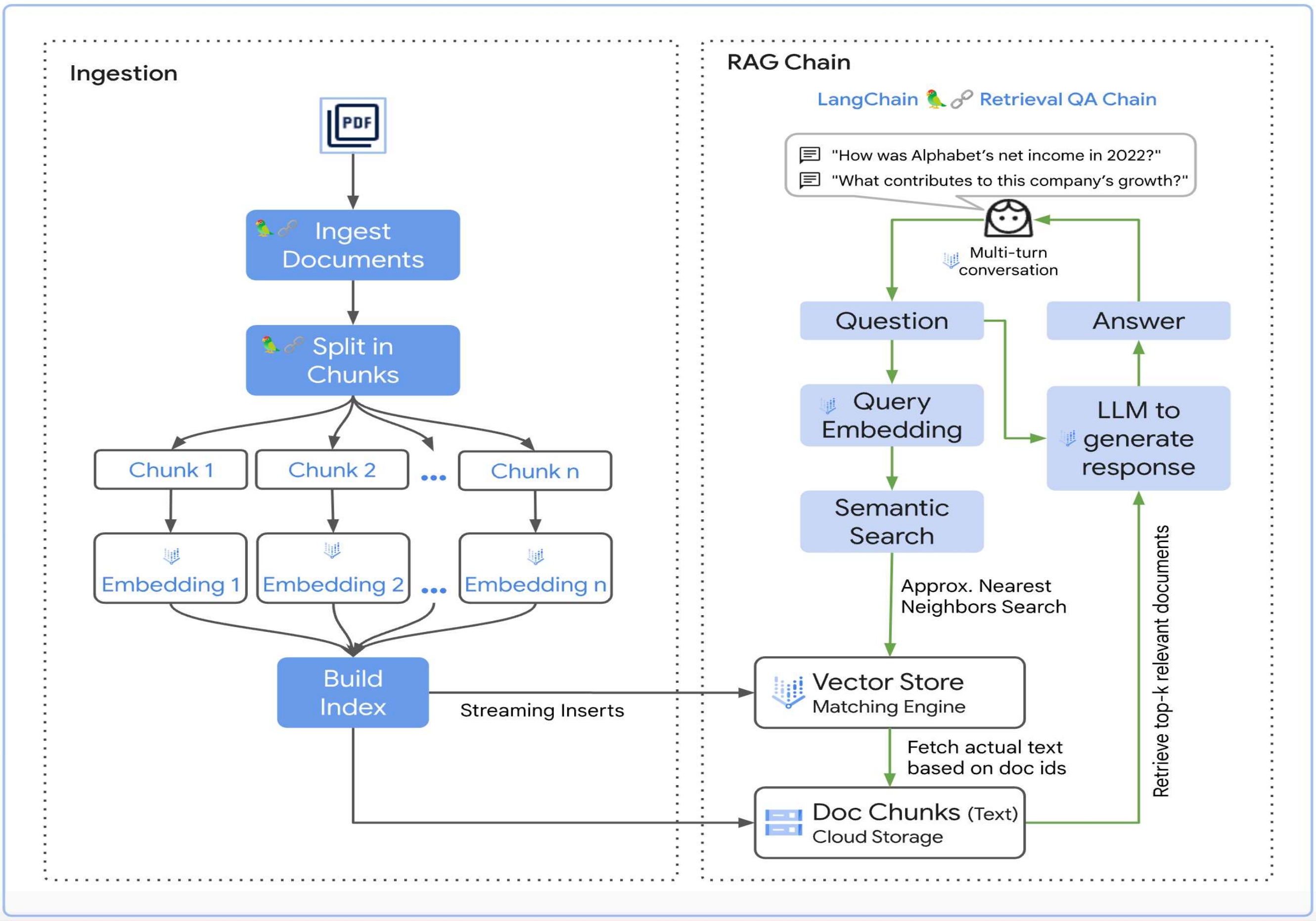
```
to_markdown(response.text)
```

It seems you provided an image that is not related to food or meal prepping. Therefore, I cannot write a blog post based on this picture.

Please provide a different image or description of the meal you would like me to write about. I am happy to help you create an engaging blog post about your meal prepping journey!

Chat with Documents

to_markdown(f"""



""")

Start coding or generate with AI.