



Routing by semantic similarity

With LCEL you can easily add **custom routing logic** to your chain to dynamically determine the chain logic based on user input. All you need to do is define a function that given an input returns a `Runnable`.

One especially useful technique is to use embeddings to route a query to the most relevant prompt. Here's a very simple example.

```
%pip install --upgrade --quiet langchain-  
core langchain langchain-openai
```

```
from langchain.utils.math import  
cosine_similarity  
from langchain_core.output_parsers import  
StrOutputParser  
from langchain_core.prompts import  
PromptTemplate  
from langchain_core.runnables import  
RunnableLambda, RunnablePassthrough  
from langchain_openai import ChatOpenAI,
```

OpenAIEmbeddings

```
physics_template = """You are a very smart  
physics professor. \  
You are great at answering questions about  
physics in a concise and easy to understand  
manner. \  
When you don't know the answer to a question  
you admit that you don't know.
```

```
Here is a question:  
{query}"""
```

```
math_template = """You are a very good  
mathematician. You are great at answering  
math questions. \  
You are so good because you are able to break  
down hard problems into their component  
parts, \  
answer the component parts, and then put them  
together to answer the broader question.
```

```
Here is a question:  
{query}"""
```

```
embeddings = OpenAIEmbeddings()  
prompt_templates = [physics_template,  
math_template]  
prompt_embeddings =  
embeddings.embed_documents(prompt_templates)
```

```
def prompt_router(input):
    query_embedding =
embeddings.embed_query(input["query"])
    similarity =
cosine_similarity([query_embedding],
prompt_embeddings)[0]
    most_similar =
prompt_templates[similarity.argmax()]
    print("Using MATH" if most_similar ==
math_template else "Using PHYSICS")
    return
PromptTemplate.from_template(most_similar)

chain = (
    {"query": RunnablePassthrough()}
    | RunnableLambda(prompt_router)
    | ChatOpenAI()
    | StrOutputParser()
)
```

```
print(chain.invoke("What's a black hole"))
```

Using PHYSICS

A black hole is a region in space where gravity is extremely strong, so strong that nothing, not even light, can escape its

gravitational pull. It is formed when a massive star collapses under its own gravity during a supernova explosion. The collapse causes an incredibly dense mass to be concentrated in a small volume, creating a gravitational field that is so intense that it warps space and time. Black holes have a boundary called the event horizon, which marks the point of no return for anything that gets too close. Beyond the event horizon, the gravitational pull is so strong that even light cannot escape, hence the name "black hole." While we have a good understanding of black holes, there is still much to learn, especially about what happens inside them.

```
print(chain.invoke("What's a path integral"))
```

Using MATH

Thank you for your kind words! I will do my best to break down the concept of a path integral for you.

In mathematics and physics, a path integral is a mathematical tool used to calculate the probability amplitude or wave function of a particle or system of particles. It was

introduced by Richard Feynman and is an integral over all possible paths that a particle can take to go from an initial state to a final state.

To understand the concept better, let's consider an example. Suppose we have a particle moving from point A to point B in space. Classically, we would describe this particle's motion using a definite trajectory, but in quantum mechanics, particles can simultaneously take multiple paths from A to B.

The path integral formalism considers all possible paths that the particle could take and assigns a probability amplitude to each path. These probability amplitudes are then added up, taking into account the interference effects between different paths.

To calculate a path integral, we need to define an action, which is a mathematical function that describes the behavior of the system. The action is usually expressed in terms of the particle's position, velocity, and time.

Once we have the action, we can write down the path integral as an integral over all possible paths. Each path is weighted by a

factor determined by the action and the principle of least action, which states that a particle takes a path that minimizes the action.

Mathematically, the path integral is expressed as:

$$\int e^{(iS/\hbar)} D[x(t)]$$

Here, S is the action, \hbar is the reduced Planck's constant, and $D[x(t)]$ represents the integration over all possible paths $x(t)$ of the particle.

By evaluating this integral, we can obtain the probability amplitude for the particle to go from the initial state to the final state. The absolute square of this amplitude gives us the probability of finding the particle in a particular state.

Path integrals have proven to be a powerful tool in various areas of physics, including quantum mechanics, quantum field theory, and statistical mechanics. They allow us to study complex systems and calculate probabilities that are difficult to obtain using other methods.

I hope this explanation helps you understand

the concept of a path integral. If you have any further questions, feel free to ask!