Model comparison

Constructing your language model application will likely involved choosing between many different options of prompts, models, and even chains to use. When doing so, you will want to compare these different options on different inputs in an easy, flexible, and intuitive way.

LangChain provides the concept of a ModelLaboratory to test out and try different models.

```
from langchain import LLMChain, OpenAI, Cohere, HuggingFaceHub,
PromptTemplate
from langchain.model_laboratory import ModelLaboratory
```

API Reference:

- ModelLaboratory from `langchain.model_laboratory`

```
llms = [
    OpenAI(temperature=0),
    Cohere(model="command-xlarge-20221108", max_tokens=20,
temperature=0),
    HuggingFaceHub(repo_id="google/flan-t5-xl", model_kwargs=
{"temperature": 1}),
]
```

```
model_lab = ModelLaboratory.from_llms(llms)
```

```
model_lab.compare("What color is a flamingo?")
```

```
    Input:
    What color is a flamingo?

    OpenAI
    Params: {'model': 'text-davinci-002', 'temperature': 0.0,
'max_tokens': 256, 'top_p': 1, 'frequency_penalty': 0,
'presence_penalty': 0, 'n': 1, 'best_of': 1}
```

```
    Flamingos are pink.

    Cohere
    Params: {'model': 'command-xlarge-20221108', 'max_tokens': 20,
'temperature': 0.0, 'k': 0, 'p': 1, 'frequency_penalty': 0,
'presence_penalty': 0}


    Pink

    HuggingFaceHub
    Params: {'repo_id': 'google/flan-t5-xl', 'temperature': 1}
    pink
```

```
prompt = PromptTemplate(
    template="What is the capital of {state}?", input_variables=
["state"]
)
model_lab_with_prompt = ModelLaboratory.from_llms(llms, prompt=prompt)
```

```
model_lab_with_prompt.compare("New York")
```

```
    Input:
    New York

    OpenAI
    Params: {'model': 'text-davinci-002', 'temperature': 0.0,
'max_tokens': 256, 'top_p': 1, 'frequency_penalty': 0,
'presence_penalty': 0, 'n': 1, 'best_of': 1}


    The capital of New York is Albany.

    Cohere
    Params: {'model': 'command-xlarge-20221108', 'max_tokens': 20,
'temperature': 0.0, 'k': 0, 'p': 1, 'frequency_penalty': 0,
'presence_penalty': 0}


    The capital of New York is Albany.

    HuggingFaceHub
    Params: {'repo_id': 'google/flan-t5-xl', 'temperature': 1}
    st john s
```

```
from langchain import SelfAskWithSearchChain, SerpAPIWrapper

open_ai_llm = OpenAI(temperature=0)
search = SerpAPIWrapper()
self_ask_with_search_openai = SelfAskWithSearchChain(
    llm=open_ai_llm, search_chain=search, verbose=True
)

cohere_llm = Cohere(temperature=0, model="command-xlarge-20221108")
search = SerpAPIWrapper()
self_ask_with_search_cohere = SelfAskWithSearchChain(
    llm=cohere_llm, search_chain=search, verbose=True
)
```

```
chains = [self_ask_with_search_openai, self_ask_with_search_cohere]
names = [str(open_ai_llm), str(cohere_llm)]
```

```
model_lab = ModelLaboratory(chains, names=names)
```

```
model_lab.compare("What is the hometown of the reigning men's U.S. Open
champion?")
```

```
    Input:
    What is the hometown of the reigning men's U.S. Open champion?

    OpenAI
    Params: {'model': 'text-davinci-002', 'temperature': 0.0,
'max_tokens': 256, 'top_p': 1, 'frequency_penalty': 0,
'presence_penalty': 0, 'n': 1, 'best_of': 1}


    > Entering new chain...
    What is the hometown of the reigning men's U.S. Open champion?
    Are follow up questions needed here: Yes.
    Follow up: Who is the reigning men's U.S. Open champion?
    Intermediate answer: Carlos Alcaraz.
    Follow up: Where is Carlos Alcaraz from?
    Intermediate answer: El Palmar, Spain.
    So the final answer is: El Palmar, Spain
    > Finished chain.

    So the final answer is: El Palmar, Spain

    Cohere
    Params: {'model': 'command-xlarge-20221108', 'max_tokens': 256,
'temperature': 0.0, 'k': 0, 'p': 1, 'frequency_penalty': 0,
'presence_penalty': 0}


    > Entering new chain...
    What is the hometown of the reigning men's U.S. Open champion?
    Are follow up questions needed here: Yes.
    Follow up: Who is the reigning men's U.S. Open champion?
    Intermediate answer: Carlos Alcaraz.
    So the final answer is:

    Carlos Alcaraz
    > Finished chain.

    So the final answer is:

    Carlos Alcaraz
```