ideation

October 9, 2023

[ ]:
```
# Copyright 2023 Google LLC
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#     https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
```

1 Ideation with Generative Models on Vertex AI

Run in Colab

View on GitHub

Open in Vertex AI Workbench

1.1 Overview

Ideation is the creative process of generating, developing, and communicating new ideas. It is a key part of the design thinking process, and can be used to solve problems, come up with new products or services, or other creative tasks.

Generative models are a powerful tool that can be used to boost creativity and innovation. By learning how to use them effectively, you can improve your ability to come up with new ideas and solutions to problems. A key part in this is learning how to structure prompts to use generative models for ideation tasks.

Learn more about prompt design in the official documentation.

1.1.1 Objective

In this tutorial, you will learn how to use generative models from the Vertex AI SDK to accelerate the ideation process by working through the following examples: - Marketing campaign generation

- Creating reading comprehension questions - Meme generation - Interview question generation - Name generation - General tips and advice - Generating answers through “impersonation”

1.1.2 Costs

This tutorial uses billable components of Google Cloud:

- Vertex AI Generative AI Studio

Learn about Vertex AI pricing, and use the Pricing Calculator to generate a cost estimate based on your projected usage.

1.2 Getting Started

1.2.1 Install Vertex AI SDK

[ ]:
```
!pip install google-cloud-aiplatform --upgrade --user
```

Colab only: Uncomment the following cell to restart the kernel or use the button to restart the kernel. For Vertex AI Workbench you can restart the terminal using the button on top.

[ ]:
```
# # Automatically restart kernel after installs so that your environment can␣
  ↪access the new packages
# import IPython

# app = IPython.Application.instance()
# app.kernel.do_shutdown(True)
```

1.2.2 Authenticating your notebook environment

- If you are using **Colab** to run this notebook, uncomment the cell below and continue.
- If you are using **Vertex AI Workbench**, check out the setup instructions here.

[ ]:
```
# from google.colab import auth
# auth.authenticate_user()
```

1.2.3 Import libraries

Colab only: Uncomment the following cell to initialize the Vertex AI SDK. For Vertex AI Workbench, you don’t need to run this.

[ ]:
```
# import vertexai

# PROJECT_ID = "[your-project-id]"  # @param {type:"string"}
# vertexai.init(project=PROJECT_ID, location="us-central1")
```

[ ]:
```
from vertexai.language_models import TextGenerationModel
```

1.2.4 Import models

```
[ ]: generation_model = TextGenerationModel.from_pretrained("text-bison@001")
```

1.3 Ideation Examples

1.3.1 Marketing campaign generation

In this example, our generation example will involve the process of creating new cookie recipes. Let's see how this can be done using the PaLM API.

```
[ ]: prompt = "Generate a marketing campaign for sustainability and fashion"

print(
    generation_model.predict(
        prompt, temperature=0.2, max_output_tokens=1024, top_k=40, top_p=0.8
    ).text
)
```

1.3.2 Creating reading comprehension questions

Reading comprehension tests are often used in schools and universities to assess a student's reading skills. You can use the PaLM API to generate some example questions to test a person's understanding of a provided passage of text.

```
[ ]: prompt = """
Generate 5 questions that test a reader's comprehension of the following text.

Text:
The Amazon rainforest, also called Amazon jungle or Amazonia, is a moist␣
 ↪broadleaf tropical rainforest in the Amazon biome that covers most of the␣
 ↪Amazon basin of South America. This basin encompasses 7,000,000 km2␣
 ↪(2,700,000 sq mi), of which 5,500,000 km2 (2,100,000 sq mi) are covered by␣
 ↪the rainforest. This region includes territory belonging to nine nations and␣
 ↪3,344 formally acknowledged indigenous territories.

The majority of the forest, 60%, is in Brazil, followed by Peru with 13%,␣
 ↪Colombia with 10%, and with minor amounts in Bolivia, Ecuador, French␣
 ↪Guiana, Guyana, Suriname, and Venezuela. Four nations have "Amazonas" as the␣
 ↪name of one of their first-level administrative regions, and France uses the␣
 ↪name "Guiana Amazonian Park" for French Guiana's protected rainforest area.␣
 ↪The Amazon represents over half of the planet's remaining rainforests, and␣
 ↪comprises the largest and most biodiverse tract of tropical rainforest in␣
 ↪the world, with an estimated 390 billion individual trees in about 16,000␣
 ↪species.
```

```
More than 30 million people of 350 different ethnic groups live in the Amazon,␣
 ↪which are subdivided into 9 different national political systems and 3,344␣
 ↪formally acknowledged indigenous territories. Indigenous peoples make up 9%␣
 ↪of the total population, and 60 of the groups remain largely isolated.

The rainforest likely formed during the Eocene era (from 56 million years to 33.
 ↪9 million years ago). It appeared following a global reduction of tropical␣
 ↪temperatures when the Atlantic Ocean had widened sufficiently to provide a␣
 ↪warm, moist climate to the Amazon basin. The rainforest has been in␣
 ↪existence for at least 55 million years, and most of the region remained␣
 ↪free of savanna-type biomes at least until the current ice age when the␣
 ↪climate was drier and savanna more widespread.

Following the Cretaceous-Paleogene extinction event, the extinction of the␣
 ↪dinosaurs and the wetter climate may have allowed the tropical rainforest to␣
 ↪spread out across the continent. From 66 to 34 Mya, the rainforest extended␣
 ↪as far south as 45°. Climate fluctuations during the last 34 million years␣
 ↪have allowed savanna regions to expand into the tropics. During the␣
 ↪Oligocene, for example, the rainforest spanned a relatively narrow band. It␣
 ↪expanded again during the Middle Miocene, then retracted to a mostly inland␣
 ↪formation at the last glacial maximum. However, the rainforest still managed␣
 ↪to thrive during these glacial periods, allowing for the survival and␣
 ↪evolution of a broad diversity of species.

Aerial view of the Amazon rainforest
During the mid-Eocene, it is believed that the drainage basin of the Amazon was␣
 ↪split along the middle of the continent by the Púrus Arch. Water on the␣
 ↪eastern side flowed toward the Atlantic, while to the west water flowed␣
 ↪toward the Pacific across the Amazonas Basin. As the Andes Mountains rose,␣
 ↪however, a large basin was created that enclosed a lake; now known as the␣
 ↪Solimões Basin. Within the last 5-10 million years, this accumulating water␣
 ↪broke through the Púrus Arch, joining the easterly flow toward the Atlantic.
```

```
There is evidence that there have been significant changes in the Amazon␣
 ↪rainforest vegetation over the last 21,000 years through the last glacial␣
 ↪maximum (LGM) and subsequent deglaciation. Analyses of sediment deposits␣
 ↪from Amazon basin paleolakes and the Amazon Fan indicate that rainfall in␣
 ↪the basin during the LGM was lower than for the present, and this was almost␣
 ↪certainly associated with reduced moist tropical vegetation cover in the␣
 ↪basin. In present day, the Amazon receives approximately 9 feet of rainfall␣
 ↪annually. There is a debate, however, over how extensive this reduction was.␣
 ↪Some scientists argue that the rainforest was reduced to small, isolated␣
 ↪refugia separated by open forest and grassland; other scientists argue that␣
 ↪the rainforest remained largely intact but extended less far to the north,␣
 ↪south, and east than is seen today. This debate has proved difficult to␣
 ↪resolve because the practical limitations of working in the rainforest mean␣
 ↪that data sampling is biased away from the center of the Amazon basin, and␣
 ↪both explanations are reasonably well supported by the available data.

Sahara Desert dust windblown to the Amazon
More than 56% of the dust fertilizing the Amazon rainforest comes from the␣
 ↪Bodélé depression in Northern Chad in the Sahara desert. The dust contains␣
 ↪phosphorus, important for plant growth. The yearly Sahara dust replaces the␣
 ↪equivalent amount of phosphorus washed away yearly in Amazon soil from rains␣
 ↪and floods.

NASA's CALIPSO satellite has measured the amount of dust transported by wind␣
 ↪from the Sahara to the Amazon: an average of 182 million tons of dust are␣
 ↪windblown out of the Sahara each year, at 15 degrees west longitude, across␣
 ↪2,600 km (1,600 mi) over the Atlantic Ocean (some dust falls into the␣
 ↪Atlantic), then at 35 degrees West longitude at the eastern coast of South␣
 ↪America, 27.7 million tons (15%) of dust fall over the Amazon basin (22␣
 ↪million tons of it consisting of phosphorus), 132 million tons of dust␣
 ↪remain in the air, 43 million tons of dust are windblown and falls on the␣
 ↪Caribbean Sea, past 75 degrees west longitude.

CALIPSO uses a laser range finder to scan the Earth's atmosphere for the␣
 ↪vertical distribution of dust and other aerosols. CALIPSO regularly tracks␣
 ↪the Sahara-Amazon dust plume. CALIPSO has measured variations in the dust␣
 ↪amounts transported - an 86 percent drop between the highest amount of dust␣
 ↪transported in 2007 and the lowest in 2011.
A possibility causing the variation is the Sahel, a strip of semi-arid land on␣
 ↪the southern border of the Sahara. When rain amounts in the Sahel are␣
 ↪higher, the volume of dust is lower. The higher rainfall could make more␣
 ↪vegetation grow in the Sahel, leaving less sand exposed to winds to blow␣
 ↪away.[25]

Amazon phosphorus also comes as smoke due to biomass burning in Africa.

Questions:
```

```
"""

print(
    generation_model.predict(
        prompt, temperature=0.2, max_output_tokens=1024, top_k=40, top_p=0.8
    ).text
)
```

1.3.3 Meme generation

A more lighthearted text generation example is to generate memes based on a certain topic.

[ ]:
```
prompt = "Give me 5 dog meme ideas:"

print(
    generation_model.predict(
        prompt, temperature=0.2, max_output_tokens=1024, top_k=1, top_p=0.8
    ).text
)
```

1.3.4 Interview question generation

Whether you are the interviewer or interviewee, having some sample interview questions you can work with can be very helpful in job interviews. Below we use the PaLM API to help us generate some potential interview questions for a particular role.

[ ]:
```
prompt = "Give me ten interview questions for the role of prompt engineer."

print(
    generation_model.predict(
        prompt, temperature=0.2, max_output_tokens=256, top_k=1, top_p=0.8
    ).text
)
```

1.3.5 Name generation

Name generation is useful in a variety of scenarios, such as creating new characters for a story or naming a new product or company. You can generate ideas for names of a specified entity using the PaLM API.

[ ]:
```
prompt = "What's a good name for a flower shop that specializes in selling␣
 ↪bouquets of dried flowers?"

print(
    generation_model.predict(
        prompt, temperature=0.2, max_output_tokens=256, top_k=1, top_p=0.8
    ).text
)
```

1.3.6 General tips and advice

Below is an example of using the PaLM API to get tips and advice on general topics.

[ ]:
```
prompt = "What are some strategies for overcoming writer's block?"

print(
    generation_model.predict(
        prompt, temperature=0.2, max_output_tokens=1024, top_k=1, top_p=0.8
    ).text
)
```

1.3.7 Generating answers through "impersonation"

Below is an example for using PaLM API to impersonating a pirate.

[ ]:
```
prompt = """You are a pirate. Take the following sentence and rephrase it as a␣
 ↪pirate.
'Learn as if you will live forever, live like you will die tomorrow.'
"""

print(
    generation_model.predict(
        prompt, temperature=0.8, max_output_tokens=1024, top_k=40, top_p=0.8
    ).text
)
```