How to Build a Smart Chatbot in 10 mins with LangChain

ALEX XU
6 JUIN 2023 · PAID

♡ 149 💬 5 Share

A large number of people have shown a keen interest in learning how to build a smart chatbot. To help us gain a better understanding of the process, I'm excited to bring you a special guest post by Damien Benveniste. He is the author of The AiEdge newsletter and was a Machine Learning Tech Lead at Meta. He holds a PhD from The Johns Hopkins University.

Below, he shares how to build a smart chatbot in 10 minutes with LangChain.

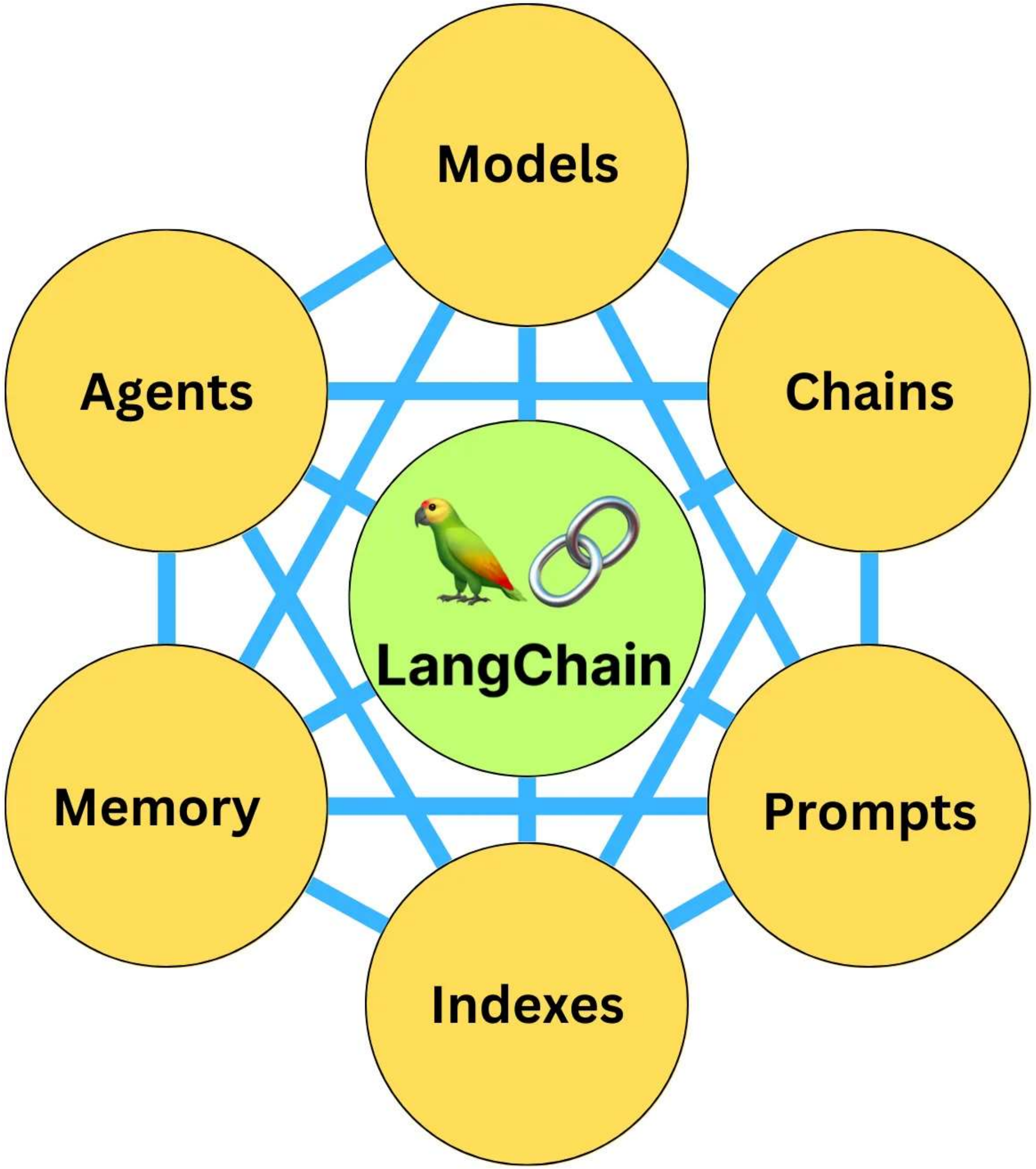
Subscribe to Damien's The AiEdge newsletter for more. You can also follow him on LinkedIn and Twitter.

LangChain is an incredible tool for interacting with Large Language Models (LLM.) In this deep dive, I'll show you how to use databases, tools and memory to build a smart chatbot. At the end, I show how to ask ChatGPT for investment advice. This article covers:

- What is LangChain?

What is LangChain?

LangChain is a package to build applications using LLMs. It is composed of 6 modules:


Models
Agents
Chains
LangChain
Memory
Prompts
Indexes

- **Prompts:** This module allows you to build dynamic prompts using templates. It can adapt to different LLM types depending on the context window size and input variables used as context, such as conversation history, search results, previous answers, and more.
- **Models:** This module provides an abstraction layer to connect to most available third- party LLM APIs. It has API connections to ~40 public LLMs, chat and embedding models.
- **Memory:** This gives the LLMs access to the conversation history.
- **Indexes:** Indexes refer to ways to structure documents so that LLMs can best interact with them. This module contains utility functions for working with documents and integration to different vector databases.
- **Agents:** Some applications require not just a predetermined chain of calls to LLMs or other tools, but potentially to an unknown chain that depends on the user's input. In these types of chains, there is an agent with access to a suite of tools. Depending on the user's input, the agent can decide which – if any – tool to call.
- **Chains:** Using an LLM in isolation is fine for some simple applications, but many more complex ones require the chaining of LLMs, either with each other, or other experts. LangChain provides a standard interface for Chains, as well as some common implementations of chains for ease of use.

Currently, the API is not well documented and is disorganized, but if you are willing to dig into the source code, it is well worth the effort. I advise you to watch the following introductory video to get more familiar with it:

Impossible d'afficher cette page

Selon les stratégies d'accès de votre entreprise, ce site Web (https://www.youtube-nocookie.com/embed/LbT1yp6quS8?rel=0&autoplay=0&showinfo=0&enablejsapi=0) a été bloqué car la catégorie Web "Streaming Video" n'est pas autorisée.

Si vous avez des questions, contactez Le centre de service (selfservice@gs2e.ci) et indiquez les codes affichés ci-dessous.

Date : Mon, 21 Aug 2023 09:44:52 GMT
Nom d'utilisateur : UNIVERS\souleysanogo@ActiveDirectory
IP source : 10.109.247.173
URL : GET https://www.youtube-nocookie.com/embed/LbT1yp6quS8?rel=0&autoplay=0&showinfo=0&enablejsapi=0
Catégorie : Streaming Video
Motif : BLOCK-WEBCAT
Notification : WEBCAT

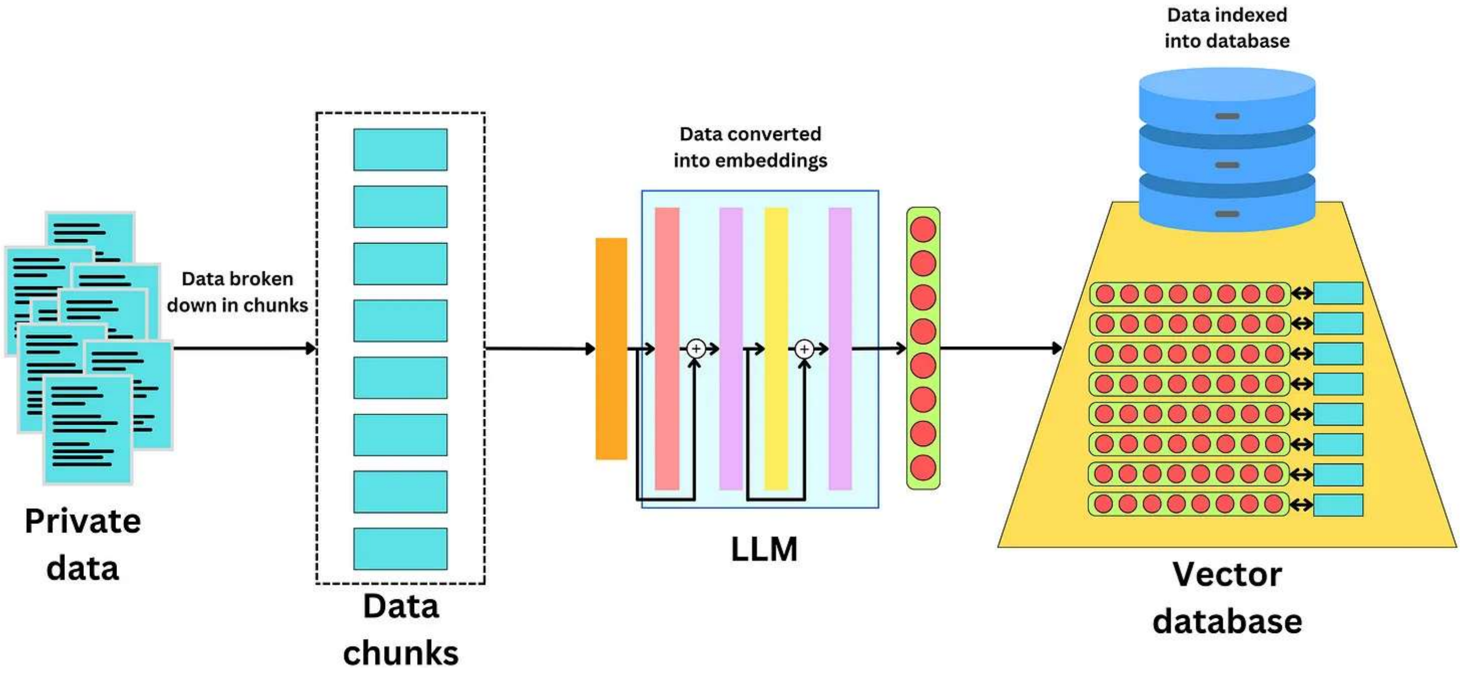
I now demonstrate how to use LangChain. You can install all the necessary libraries by running the following:

```
pip install pinecone-client langchain openai wikipedia google-api-python-client
unstructured tabulate pdf2image
```

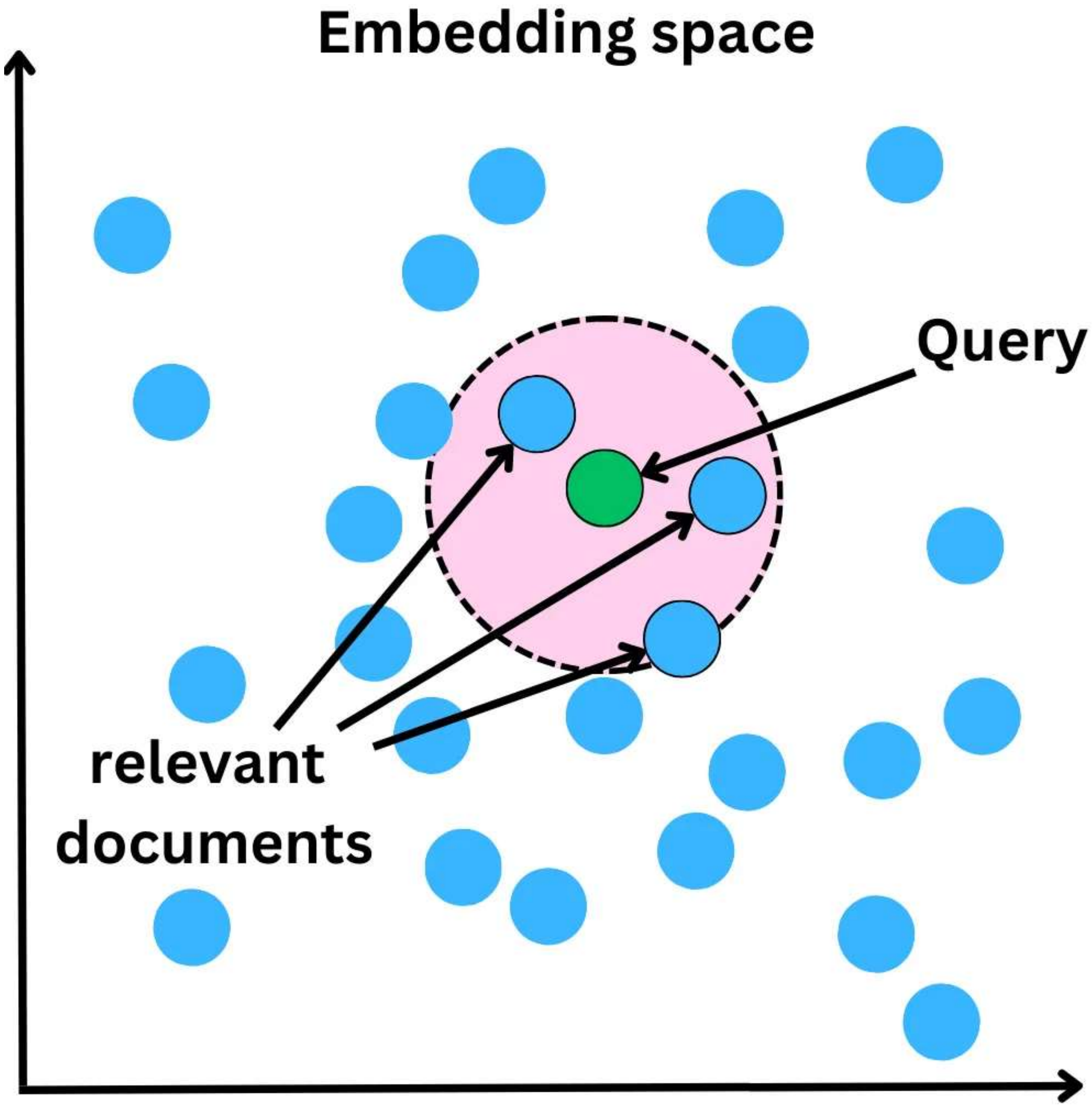
Indexing and searching new data

One difficulty with LLMs is that they only know what they learned during training. So how do we get them to use private data? One way is to make new text data discoverable by the LLM. The typical way to do this is to convert all private data into embeddings stored in a vector database. The process is as follows:

- Chunk the data into small pieces
- Pass that data through an LLM. The resulting final layer of the network can be used as a semantic vector representation of the data
- The data can then be stored in a database of the vector representation used to recover that piece of data

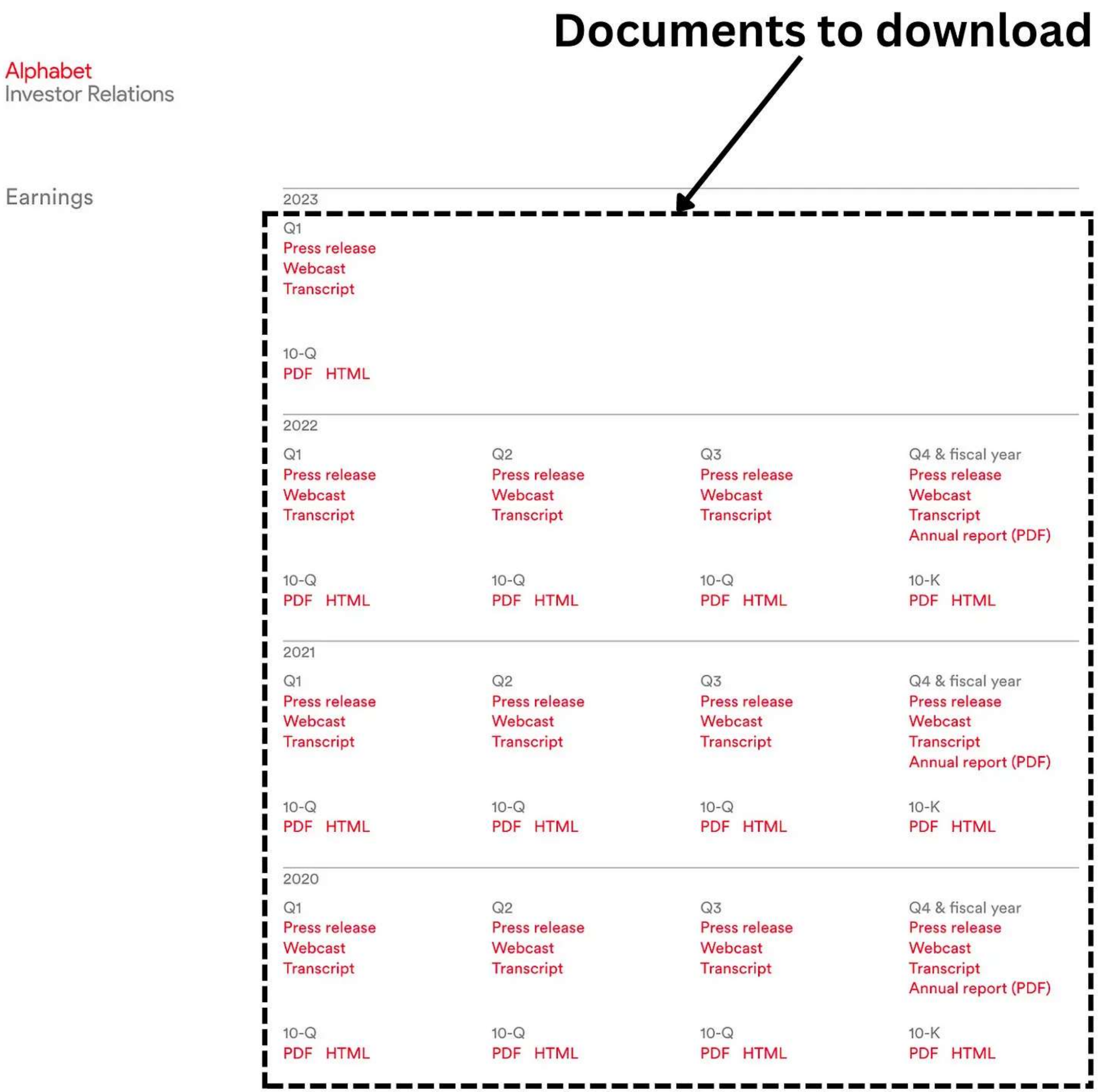

Private data
Data broken down in chunks
Data chunks
Data converted into embeddings
LLM
Data indexed into database
Vector database

A question which we ask can be converted into an embedding, which is the query. We can then search for pieces of data located close to it in the embedding space and feed relevant documents to the LLM for it to extract an answer from:



Let’s get some data

I sourced interesting data for a demonstration and selected the earnings reports of tech giant, Alphabet (Google): https://abc.xyz/investor/previous/



Documents to download

Alphabet
Investor Relations

Earnings

2023

Q1
Press release Webcast Transcript
10-Q PDF HTML

2022

Q1	Q2	Q3	Q4 & fiscal year
Press release Webcast Transcript	Press release Webcast Transcript	Press release Webcast Transcript	Press release Webcast Transcript Annual report (PDF)
10-Q PDF HTML	10-Q PDF HTML	10-Q PDF HTML	10-K PDF HTML

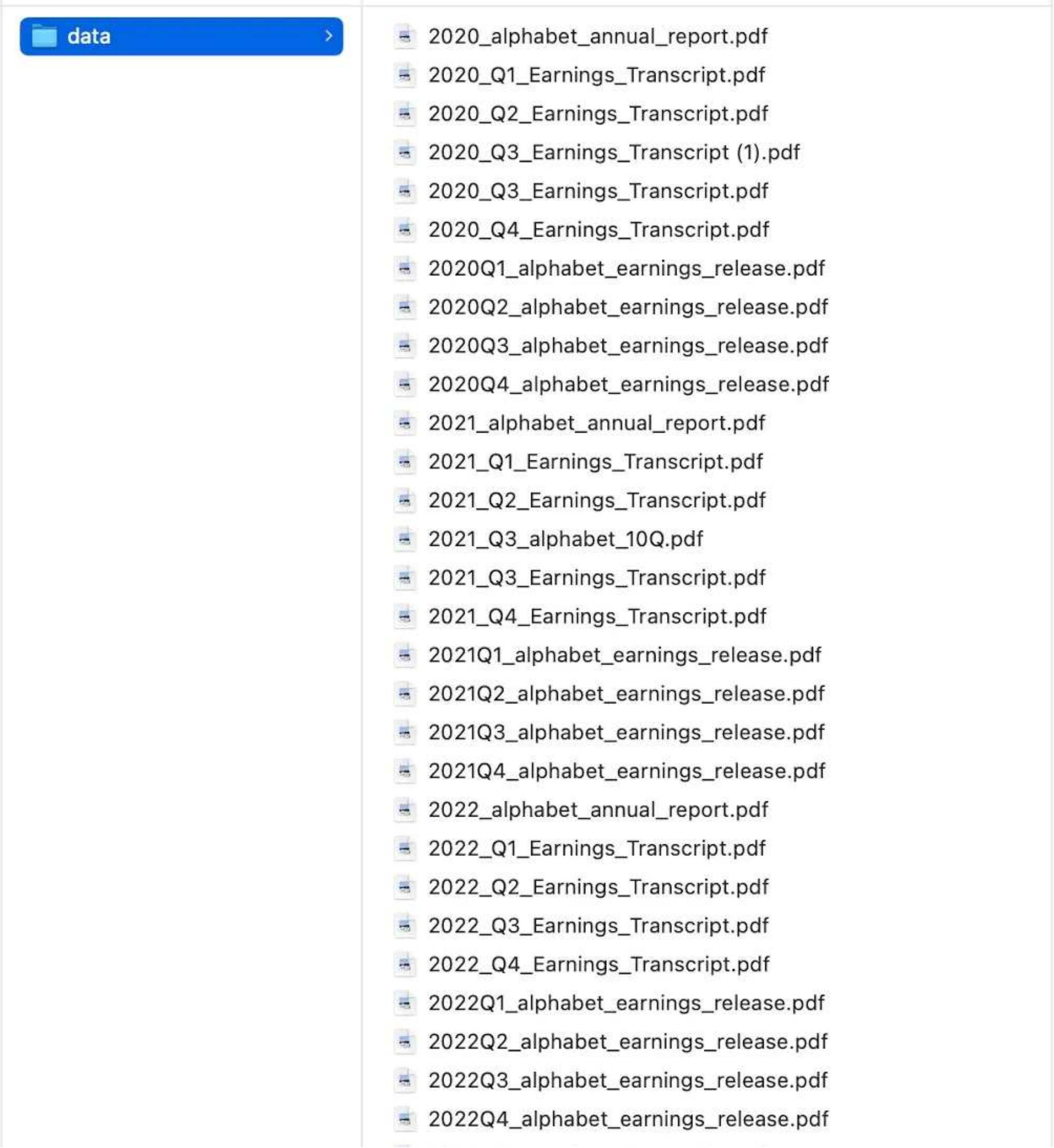
2021

Q1	Q2	Q3	Q4 & fiscal year
Press release Webcast Transcript	Press release Webcast Transcript	Press release Webcast Transcript	Press release Webcast Transcript Annual report (PDF)
10-Q PDF HTML	10-Q PDF HTML	10-Q PDF HTML	10-K PDF HTML

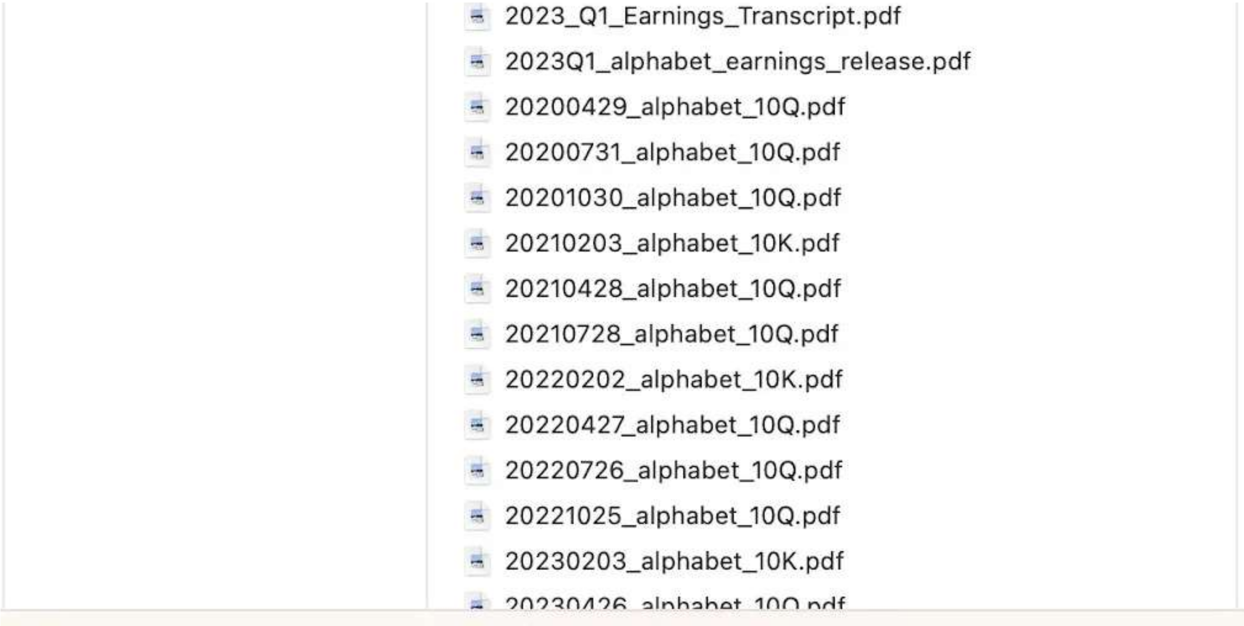
2020

Q1	Q2	Q3	Q4 & fiscal year
Press release Webcast Transcript	Press release Webcast Transcript	Press release Webcast Transcript	Press release Webcast Transcript Annual report (PDF)
10-Q PDF HTML	10-Q PDF HTML	10-Q PDF HTML	10-K PDF HTML

For simplicity, I downloaded and stored the reports on my computer's hard drive:



```
2020_alphabet_annual_report.pdf
2020_Q1_Earnings_Transcript.pdf
2020_Q2_Earnings_Transcript.pdf
2020_Q3_Earnings_Transcript (1).pdf
2020_Q3_Earnings_Transcript.pdf
2020_Q4_Earnings_Transcript.pdf
2020Q1_alphabet_earnings_release.pdf
2020Q2_alphabet_earnings_release.pdf
2020Q3_alphabet_earnings_release.pdf
2020Q4_alphabet_earnings_release.pdf
2021_alphabet_annual_report.pdf
2021_Q1_Earnings_Transcript.pdf
2021_Q2_Earnings_Transcript.pdf
2021_Q3_alphabet_10Q.pdf
2021_Q3_Earnings_Transcript.pdf
2021_Q4_Earnings_Transcript.pdf
2021Q1_alphabet_earnings_release.pdf
2021Q2_alphabet_earnings_release.pdf
2021Q3_alphabet_earnings_release.pdf
2021Q4_alphabet_earnings_release.pdf
2022_alphabet_annual_report.pdf
2022_Q1_Earnings_Transcript.pdf
2022_Q2_Earnings_Transcript.pdf
2022_Q3_Earnings_Transcript.pdf
2022_Q4_Earnings_Transcript.pdf
2022Q1_alphabet_earnings_release.pdf
2022Q2_alphabet_earnings_release.pdf
2022Q3_alphabet_earnings_release.pdf
2022Q4_alphabet_earnings_release.pdf
```

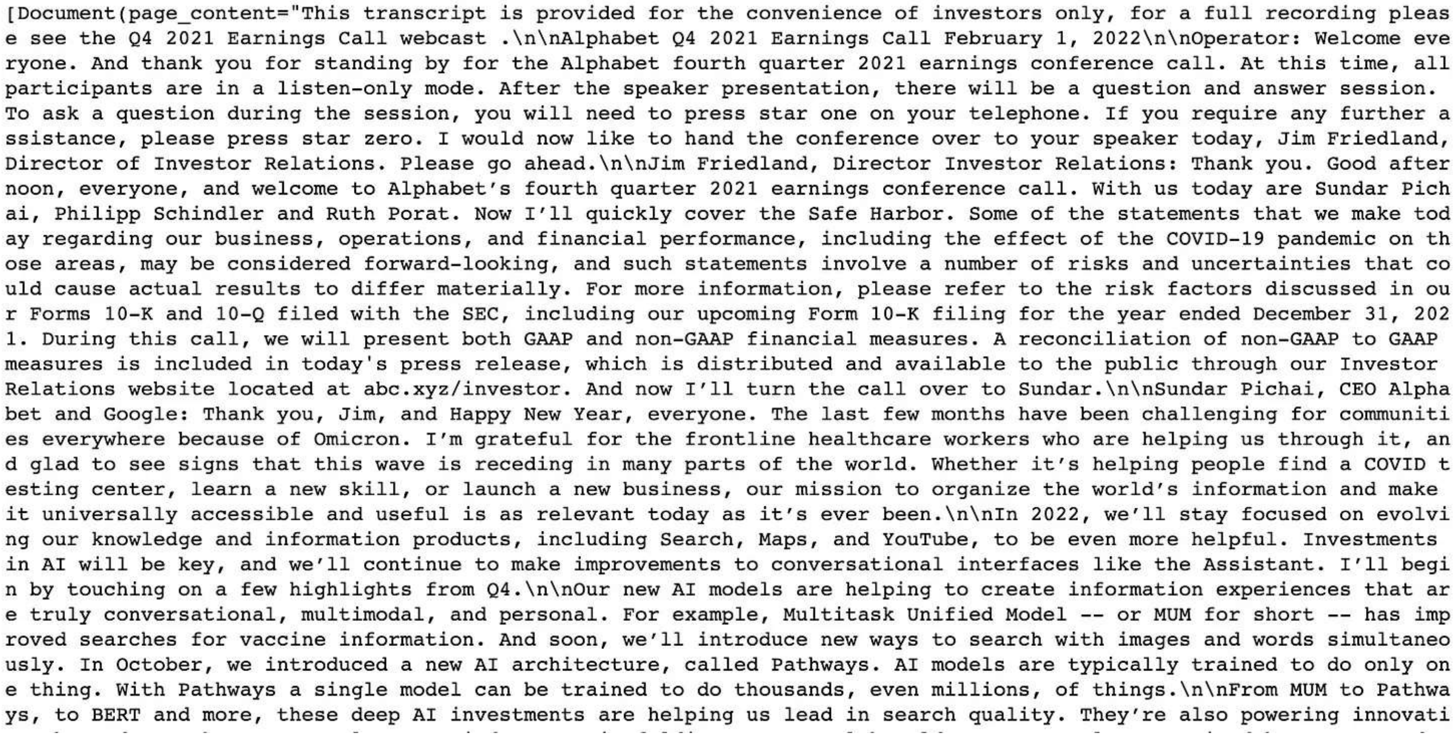


2023_Q1_Earnings_Transcript.pdf
2023Q1_alphabet_earnings_release.pdf
20200429_alphabet_10Q.pdf
20200731_alphabet_10Q.pdf
20201030_alphabet_10Q.pdf
20210203_alphabet_10K.pdf
20210428_alphabet_10Q.pdf
20210728_alphabet_10Q.pdf
20220202_alphabet_10K.pdf
20220427_alphabet_10Q.pdf
20220726_alphabet_10Q.pdf
20221025_alphabet_10Q.pdf
20230203_alphabet_10K.pdf
20230426_alphabet_10Q.pdf

We can now load those documents into memory with LangChain, using 2 lines of code:

```
from langchain.document_loaders import DirectoryLoader

loader = DirectoryLoader(
    './Langchain/data/', # my local directory
    glob='**/*.pdf',     # we only get pdfs
    show_progress=True
)
docs = loader.load()
docs
```

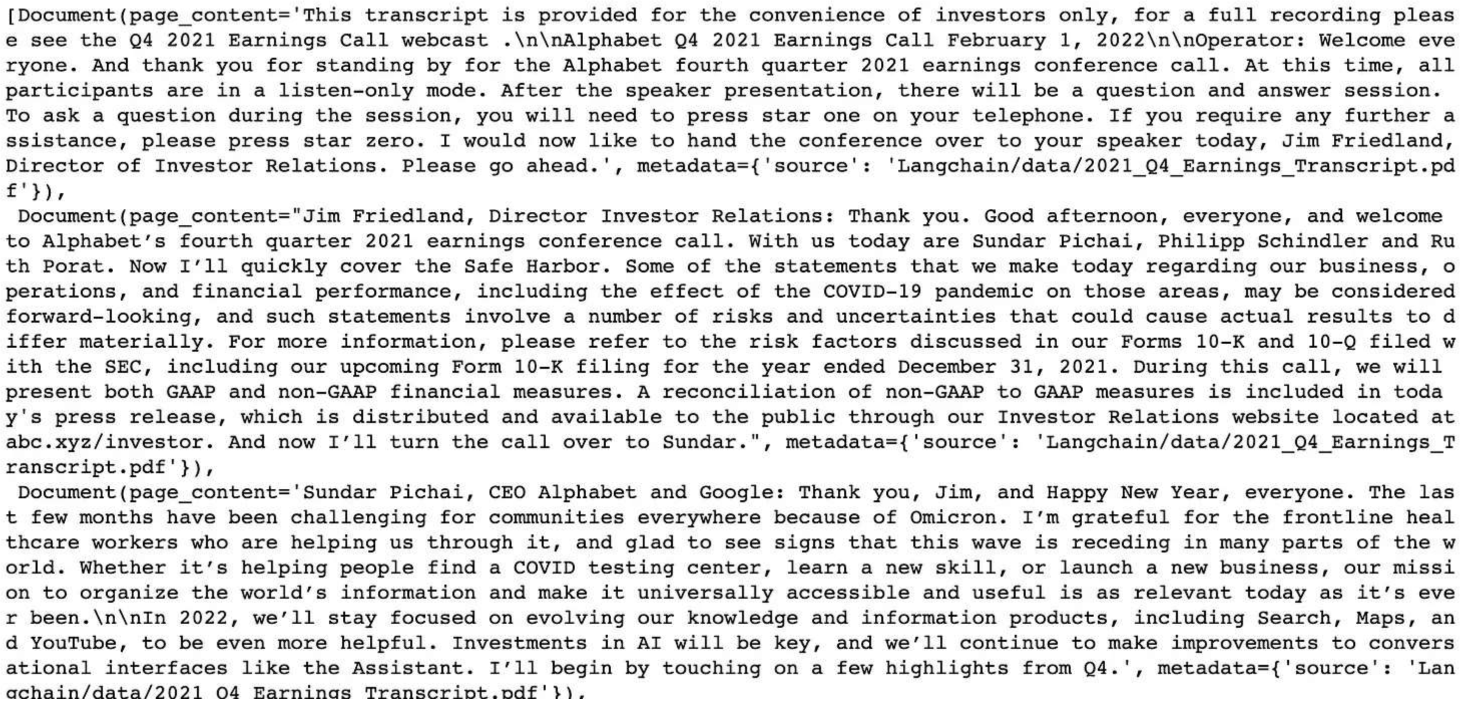


```
[Document(page_content="This transcript is provided for the convenience of investors only, for a full recording pleas
e see the Q4 2021 Earnings Call webcast .\n\nAlphabet Q4 2021 Earnings Call February 1, 2022\n\nOperator: Welcome eve
ryone. And thank you for standing by for the Alphabet fourth quarter 2021 earnings conference call. At this time, all
participants are in a listen-only mode. After the speaker presentation, there will be a question and answer session.
To ask a question during the session, you will need to press star one on your telephone. If you require any further a
ssistance, please press star zero. I would now like to hand the conference over to your speaker today, Jim Friedland,
Director of Investor Relations. Please go ahead.\n\nJim Friedland, Director Investor Relations: Thank you. Good after
noon, everyone, and welcome to Alphabet's fourth quarter 2021 earnings conference call. With us today are Sundar Pich
ai, Philipp Schindler and Ruth Porat. Now I'll quickly cover the Safe Harbor. Some of the statements that we make tod
ay regarding our business, operations, and financial performance, including the effect of the COVID-19 pandemic on th
ose areas, may be considered forward-looking, and such statements involve a number of risks and uncertainties that co
uld cause actual results to differ materially. For more information, please refer to the risk factors discussed in ou
r Forms 10-K and 10-Q filed with the SEC, including our upcoming Form 10-K filing for the year ended December 31, 202
1. During this call, we will present both GAAP and non-GAAP financial measures. A reconciliation of non-GAAP to GAAP
measures is included in today's press release, which is distributed and available to the public through our Investor
Relations website located at abc.xyz/investor. And now I'll turn the call over to Sundar.\n\nSundar Pichai, CEO Alpha
bet and Google: Thank you, Jim, and Happy New Year, everyone. The last few months have been challenging for communiti
es everywhere because of Omicron. I'm grateful for the frontline healthcare workers who are helping us through it, an
d glad to see signs that this wave is receding in many parts of the world. Whether it's helping people find a COVID t
esting center, learn a new skill, or launch a new business, our mission to organize the world's information and make
it universally accessible and useful is as relevant today as it's ever been.\n\nIn 2022, we'll stay focused on evolvi
ng our knowledge and information products, including Search, Maps, and YouTube, to be even more helpful. Investments
in AI will be key, and we'll continue to make improvements to conversational interfaces like the Assistant. I'll begi
n by touching on a few highlights from Q4.\n\nOur new AI models are helping to create information experiences that ar
e truly conversational, multimodal, and personal. For example, Multitask Unified Model -- or MUM for short -- has imp
roved searches for vaccine information. And soon, we'll introduce new ways to search with images and words simultaneo
usly. In October, we introduced a new AI architecture, called Pathways. AI models are typically trained to do only on
e thing. With Pathways a single model can be trained to do thousands, even millions, of things.\n\nFrom MUM to Pathwa
ys, to BERT and more, these deep AI investments are helping us lead in search quality. They're also powering innovati
```

We split them into chunks. Each chunk corresponds to an embedding vector.

```
from langchain.text_splitter import CharacterTextSplitter

text_splitter = CharacterTextSplitter(
    chunk_size=1000,
    chunk_overlap=0
)
docs_split = text_splitter.split_documents(docs)
docs_split
```

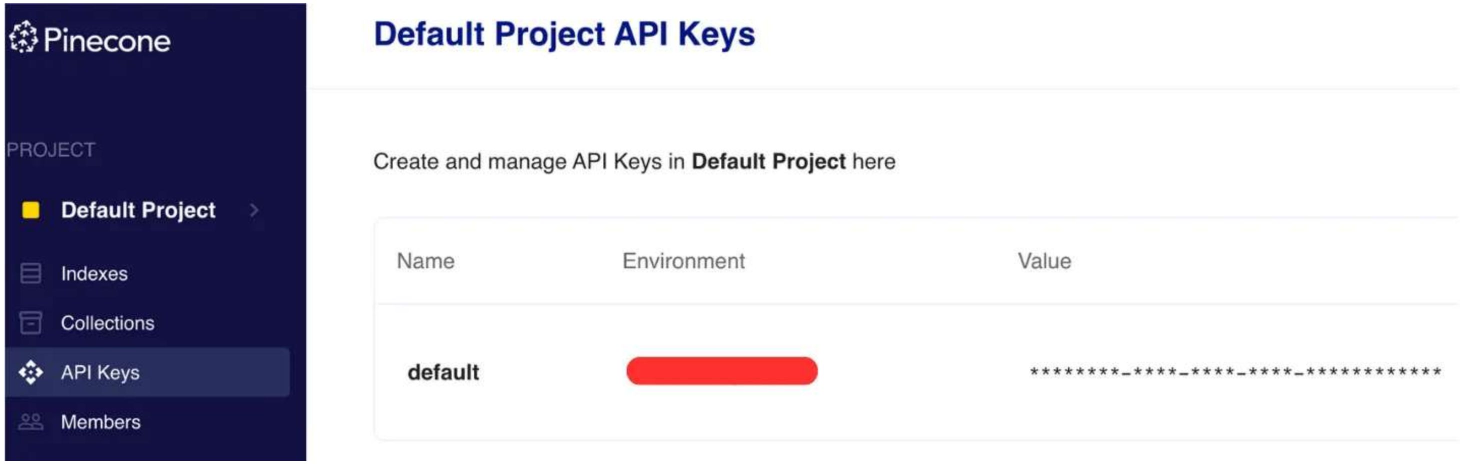


```
[Document(page_content='This transcript is provided for the convenience of investors only, for a full recording pleas
e see the Q4 2021 Earnings Call webcast .\n\nAlphabet Q4 2021 Earnings Call February 1, 2022\n\nOperator: Welcome eve
ryone. And thank you for standing by for the Alphabet fourth quarter 2021 earnings conference call. At this time, all
participants are in a listen-only mode. After the speaker presentation, there will be a question and answer session.
To ask a question during the session, you will need to press star one on your telephone. If you require any further a
ssistance, please press star zero. I would now like to hand the conference over to your speaker today, Jim Friedland,
Director of Investor Relations. Please go ahead.', metadata={'source': 'Langchain/data/2021_Q4_Earnings_Transcript.pd
f'}),
 Document(page_content="Jim Friedland, Director Investor Relations: Thank you. Good afternoon, everyone, and welcome
to Alphabet’s fourth quarter 2021 earnings conference call. With us today are Sundar Pichai, Philipp Schindler and Ru
th Porat. Now I’ll quickly cover the Safe Harbor. Some of the statements that we make today regarding our business, o
perations, and financial performance, including the effect of the COVID-19 pandemic on those areas, may be considered
forward-looking, and such statements involve a number of risks and uncertainties that could cause actual results to d
iffer materially. For more information, please refer to the risk factors discussed in our Forms 10-K and 10-Q filed w
ith the SEC, including our upcoming Form 10-K filing for the year ended December 31, 2021. During this call, we will
present both GAAP and non-GAAP financial measures. A reconciliation of non-GAAP to GAAP measures is included in toda
y's press release, which is distributed and available to the public through our Investor Relations website located at
abc.xyz/investor. And now I’ll turn the call over to Sundar.", metadata={'source': 'Langchain/data/2021_Q4_Earnings_T
ranscript.pdf'}),
 Document(page_content='Sundar Pichai, CEO Alphabet and Google: Thank you, Jim, and Happy New Year, everyone. The las
t few months have been challenging for communities everywhere because of Omicron. I’m grateful for the frontline heal
thcare workers who are helping us through it, and glad to see signs that this wave is receding in many parts of the w
orld. Whether it’s helping people find a COVID testing center, learn a new skill, or launch a new business, our missi
on to organize the world’s information and make it universally accessible and useful is as relevant today as it’s eve
r been.\n\nIn 2022, we’ll stay focused on evolving our knowledge and information products, including Search, Maps, an
d YouTube, to be even more helpful. Investments in AI will be key, and we’ll continue to make improvements to convers
ational interfaces like the Assistant. I’ll begin by touching on a few highlights from Q4.', metadata={'source': 'Lan
gchain/data/2021_Q4_Earnings_Transcript.pdf'}),
```

For this reason, we need to convert the data into embeddings and store them in a database.

Pinecone: A vector database

To store the data, I use Pinecone. You can create a free account and automatically get API keys with which to access the database:

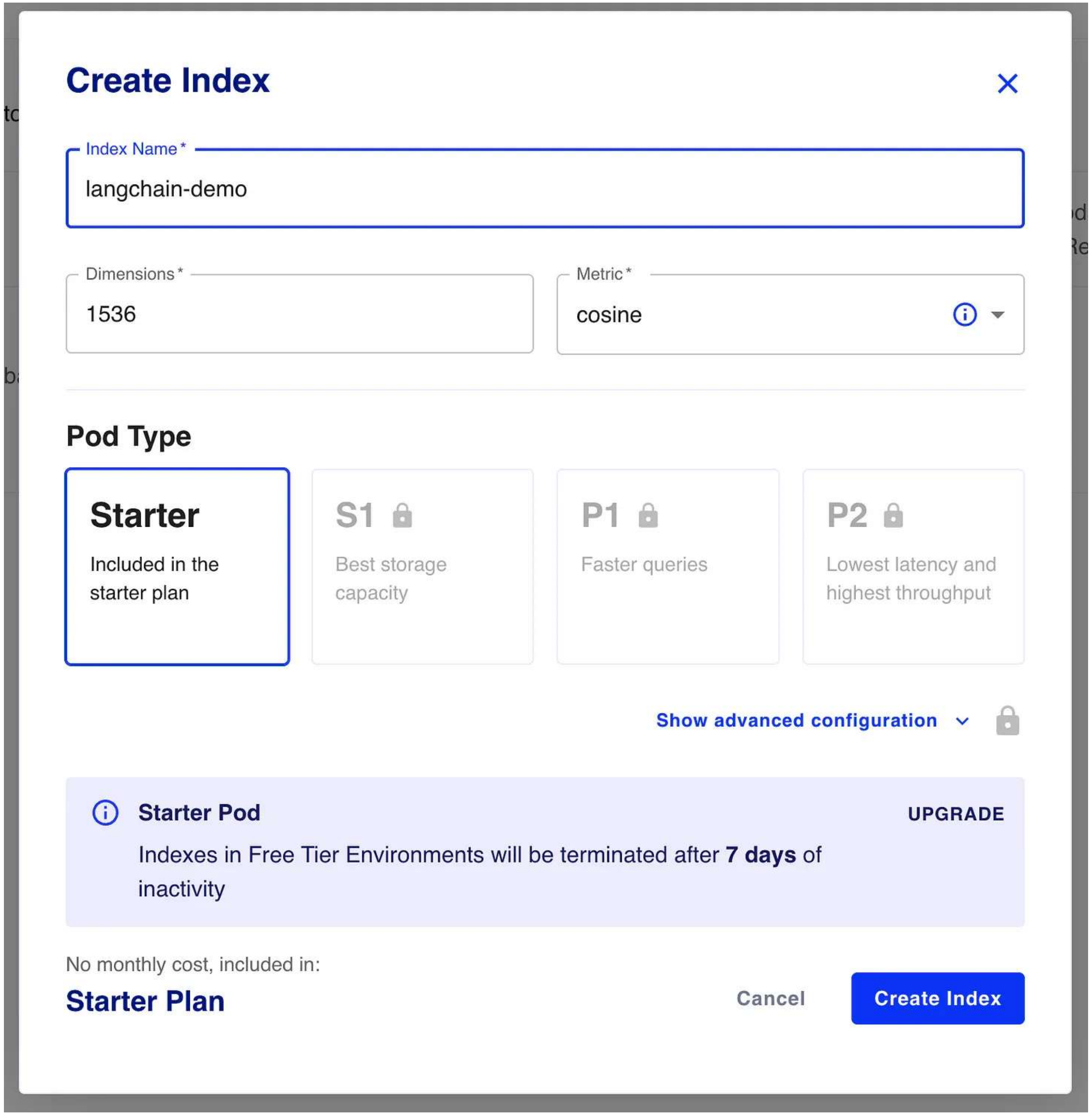


Default Project API Keys

Create and manage API Keys in **Default Project** here

Name	Environment	Value
default	[redacted]	********-****-****-****-************

In the “indexes” tab, click on “create index.” Give it a name and a dimension. I used “1536” for the dimension, as it is the size of the chosen embedding from the OpenAI embedding model. I use the cosine similarity metric to search for similar documents:



Create Index

Index Name * langchain-demo

Dimensions * 1536

Metric * cosine

Pod Type

Starter
Included in the starter plan

S1
Best storage capacity

P1
Faster queries

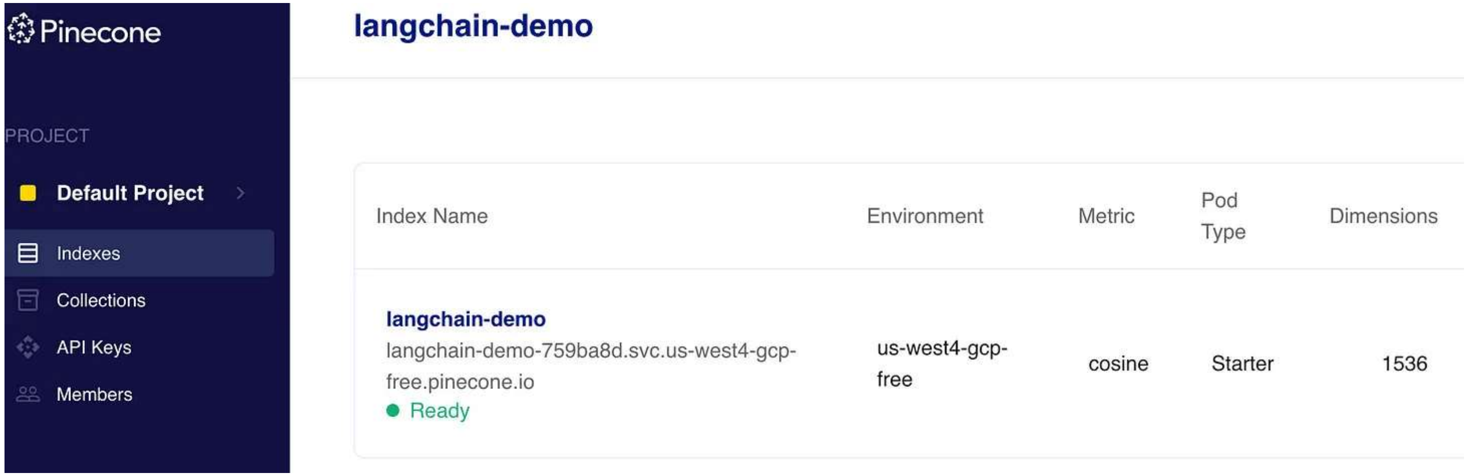
P2
Lowest latency and highest throughput

Show advanced configuration

Starter Pod
Indexes in Free Tier Environments will be terminated after **7 days** of inactivity

No monthly cost, included in:
Starter Plan

This will create a vector table:

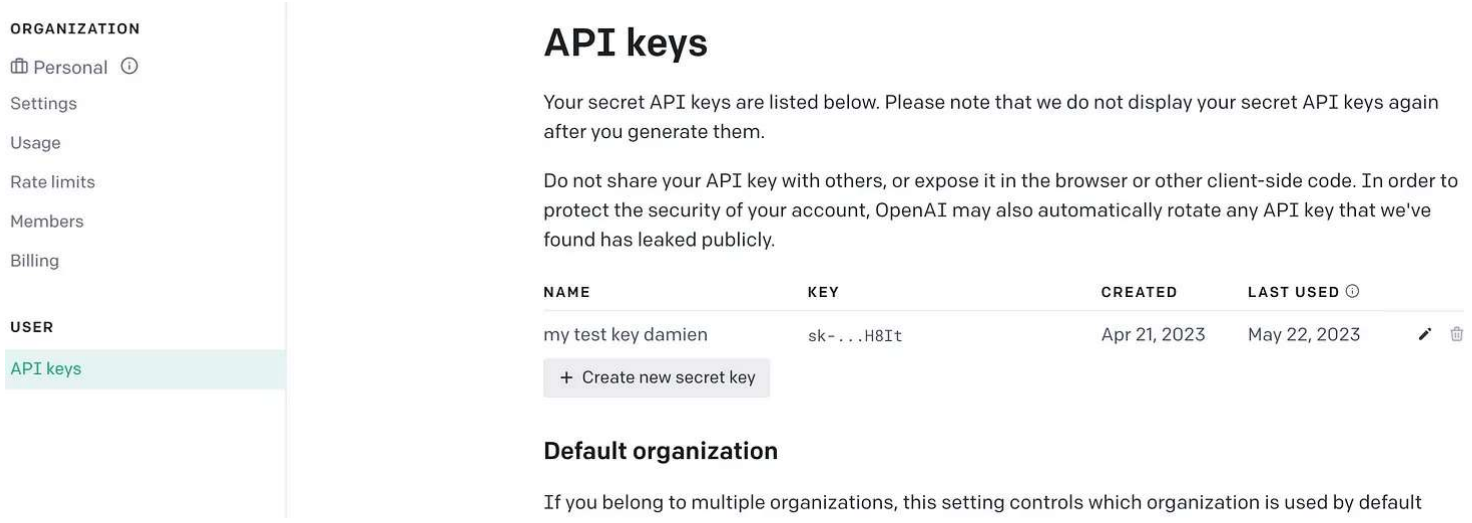


langchain-demo

Index Name	Environment	Metric	Pod Type	Dimensions
langchain-demo langchain-demo-759ba8d.svc.us-west4-gcp-free.pinecone.io ● Ready	us-west4-gcp-free	cosine	Starter	1536

Storing the data

Before continuing, make sure to get a OpenAI API key by signing up to the [OpenAI platform](https://platform.openai.com):



API keys

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, OpenAI may also automatically rotate any API key that we've found has leaked publicly.

NAME	KEY	CREATED	LAST USED
my test key damien	sk-...H8It	Apr 21, 2023	May 22, 2023

+ Create new secret key

Default organization

If you belong to multiple organizations, this setting controls which organization is used by default

Let’s first write down our API keys

```
import os

PINECONE_API_KEY = ... # find at app.pinecone.io
PINECONE_ENV = ...     # next to api key in console
OPENAI_API_KEY = ...   # found at platform.openai.com/account/api-keys

os.environ['OPENAI_API_KEY'] = OPENAI_API_KEY
```

We upload the data to the vector database. The default OpenAI embedding model used in Langchain is 'text-embedding-ada-002' (OpenAI embedding models.) It is used to convert data into embedding vectors

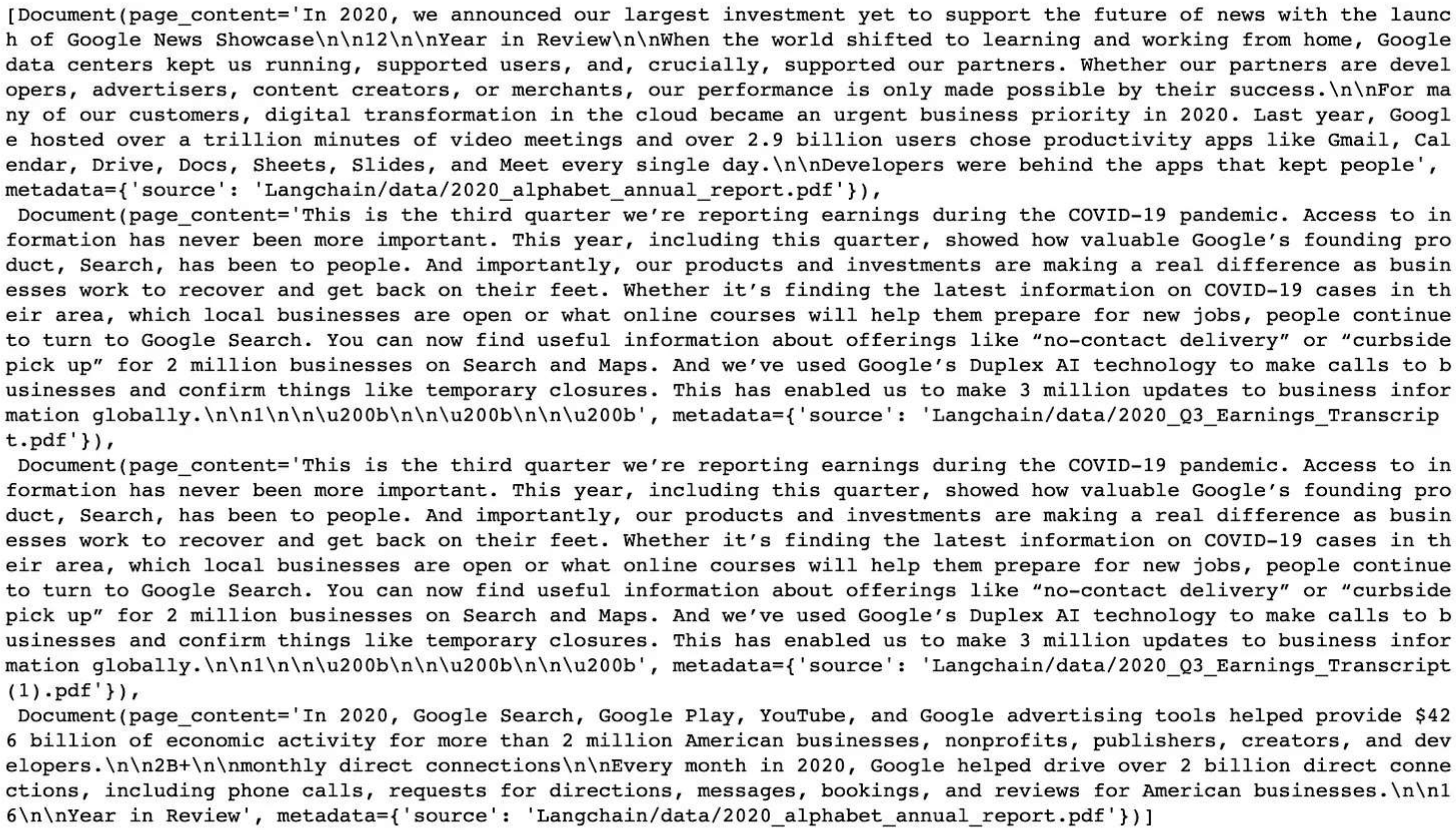
```
import pinecone
from langchain.vectorstores import Pinecone
from langchain.embeddings.openai import OpenAIEmbeddings

# we use the openAI embedding model
embeddings = OpenAIEmbeddings()
pinecone.init(
    api_key=PINECONE_API_KEY,
    environment=PINECONE_ENV
)

doc_db = Pinecone.from_documents(
    docs_split,
    embeddings,
    index_name='langchain-demo'
)
```

We can now search for relevant documents in that database using the cosine similarity metric

```
query = "What were the most important events for Google in 2021?"
search_docs = doc_db.similarity_search(query)
search_docs
```



```
[Document(page_content='In 2020, we announced our largest investment yet to support the future of news with the launc
h of Google News Showcase\n\n12\n\nYear in Review\n\nWhen the world shifted to learning and working from home, Google
data centers kept us running, supported users, and, crucially, supported our partners. Whether our partners are devel
opers, advertisers, content creators, or merchants, our performance is only made possible by their success.\n\nFor ma
ny of our customers, digital transformation in the cloud became an urgent business priority in 2020. Last year, Googl
e hosted over a trillion minutes of video meetings and over 2.9 billion users chose productivity apps like Gmail, Cal
endar, Drive, Docs, Sheets, Slides, and Meet every single day.\n\nDevelopers were behind the apps that kept people',
metadata={'source': 'Langchain/data/2020_alphabet_annual_report.pdf'}),
 Document(page_content='This is the third quarter we’re reporting earnings during the COVID-19 pandemic. Access to in
formation has never been more important. This year, including this quarter, showed how valuable Google’s founding pro
duct, Search, has been to people. And importantly, our products and investments are making a real difference as busin
esses work to recover and get back on their feet. Whether it’s finding the latest information on COVID-19 cases in th
eir area, which local businesses are open or what online courses will help them prepare for new jobs, people continue
to turn to Google Search. You can now find useful information about offerings like “no-contact delivery” or “curbside
pick up” for 2 million businesses on Search and Maps. And we’ve used Google’s Duplex AI technology to make calls to b
usinesses and confirm things like temporary closures. This has enabled us to make 3 million updates to business infor
mation globally.\n\n1\n\n\u200b\n\n\u200b\n\n\u200b', metadata={'source': 'Langchain/data/2020_Q3_Earnings_Transcrip
t.pdf'}),
 Document(page_content='This is the third quarter we’re reporting earnings during the COVID-19 pandemic. Access to in
formation has never been more important. This year, including this quarter, showed how valuable Google’s founding pro
duct, Search, has been to people. And importantly, our products and investments are making a real difference as busin
esses work to recover and get back on their feet. Whether it’s finding the latest information on COVID-19 cases in th
eir area, which local businesses are open or what online courses will help them prepare for new jobs, people continue
to turn to Google Search. You can now find useful information about offerings like “no-contact delivery” or “curbside
pick up” for 2 million businesses on Search and Maps. And we’ve used Google’s Duplex AI technology to make calls to b
usinesses and confirm things like temporary closures. This has enabled us to make 3 million updates to business infor
mation globally.\n\n1\n\n\u200b\n\n\u200b\n\n\u200b', metadata={'source': 'Langchain/data/2020_Q3_Earnings_Transcript
(1).pdf'}),
 Document(page_content='In 2020, Google Search, Google Play, YouTube, and Google advertising tools helped provide $42
6 billion of economic activity for more than 2 million American businesses, nonprofits, publishers, creators, and dev
elopers.\n\n2B+\n\nmonthly direct connections\n\nEvery month in 2020, Google helped drive over 2 billion direct conne
ctions, including phone calls, requests for directions, messages, bookings, and reviews for American businesses.\n\n1
6\n\nYear in Review', metadata={'source': 'Langchain/data/2020_alphabet_annual_report.pdf'})]
```

Retrieving data with ChatGPT

We can now use a LLM to utilize the database data. Let’s get an LLM such as GPT-3 using:

```
from langchain import OpenAI
llm = OpenAI()
```

or we could get ChatGPT using

```
from langchain.chat_models import ChatOpenAI
llm = ChatOpenAI()
```

Let’s use the RetrievalQA module to query that data:

```
from langchain.chains import RetrievalQA

qa = RetrievalQA.from_chain_type(
    llm=llm,
    chain_type='stuff',
    retriever=doc_db.as_retriever(),
)

query = "What were the earnings in 2022?"
result = qa.run(query)

result

> 'The total revenues for the full year 2022 were $282,836 million, with
operating income and operating margin information not provided in the given
context.'
```

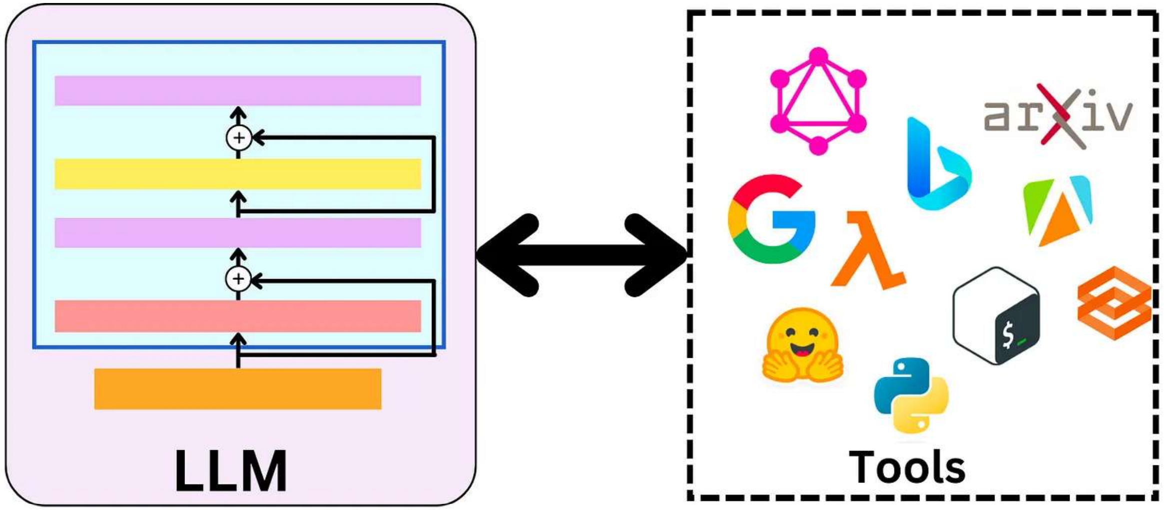
RetrievalQA is actually a wrapper around a specific prompt. The chain type “stuff“ will use a prompt, assuming the whole query text fits into the context window. It uses the following prompt template:

```
Use the following pieces of context to answer the users question.
If you don't know the answer, just say that you don't know, don't try to make
up an answer.
----------------
{context}

{question}
```

Here the context will be populated with the user’s question and the results of the retrieved documents found in the database. You can use other chain types: “map_reduce”, “refine”, and “map-rerank” if the text is longer than the context window.

Giving ChatGPT access to tools


LLM
arXiv
Tools