

VertexAi -RAG

VertexAI RAG

项目基础信息：

PROJECT_ID:

cy-aispeci-demo

BUCKET_NAME:

gs://vertex_ai_rag_demo

项目Git地址：

<https://github.com/ChuanYang-AI/Demo1>

白皮书：

<https://github.com/ChuanYang-AI/Demo1/blob/main/google-vertexai-rag/README.md>

架构图及流程图：

<https://github.com/ChuanYang-AI/Demo1/blob/main/google-vertexai-rag/docs/DIAGRAMS.md>

案例：

欢迎页：

穿扬科技 | Google Cloud Partner

AI智能问答助手

基于 Google Vertex AI 的企业级智能检索问答系统

智能对话

与AI助手进行自然语言对话

文档管理

上传并管理您的文档资料

实时检索

快速精准的语义搜索

试试这些问题

什么是定金？定金和订金有什么区别？

盗窃罪的构成要件是什么？

合同违约的法律后果有哪些？

如何处理劳动争议？

开始使用: 口 开始对话

问答：

对话历史

刚刚: 什么是定金？定金和订金有什么区别？

刚刚: 什么

刚刚: 刚刚

刚刚: 8 条消息

刚刚: 文档中提到了盗窃罪的犯罪客体，...

什么是定金？定金和订金有什么区别？

文档中详细解释了定金和订金的区别，下面进行总结：

- 定金：
 - 是一种债权担保方式，目的是促使债务人履行债务，保障债权人的债权。
 - 必须以书面形式约定，包括定金的数额和交付期限。
 - 给付定金一方不履行债务，无权要求返还定金；接受定金一方不履行债务，需双倍返还定金。
 - 债务人履行债务后，定金应抵作价款或者收回。
- 订金：
 - 法律没有明确规定，不具备定金的担保性质。
 - 可视为预付款。
 - 合同不能履行时，除不可抗力外，应根据双方当事人的过错承担违约责任。

简单来说，定金具有担保性质，违约会有惩罚（没收或双倍返还）；而订金可视为预付款，违约时根据过错承担责任。

结合文档和知识

相似度: 78%

参考文档 (5)

#1 未知文档 0

#2 未知文档 0

猎豹移动的ceo是谁?

10:27

猎豹移动的 CEO 是 傅盛。

- 傅盛于 2010 年加入猎豹移动 (当时名为金山网络)，并担任 CEO。
- 他在互联网安全和移动应用领域拥有丰富的经验。

基于AI知识 1357

10:27 · 处理时间: 1.9s

基础知识 相似度: 50%

参考文档 (5)

#1 法律知识问答.docx 块3

☆ 56% 低相关

情况，开发商应了解投资者的资信情况，而投资者应了解开发商的主体资格、审查房屋预售许可证、规划许可证等文件，确定所购买的房屋是可分割出售，取得分户产权的产权式酒店（有些酒店只有一个独立的“大产权”，不允…

#2 法律知识问答.docx 块3

☆ 56% 低相关

情况，开发商应了解投资者的资信情况，而投资者应了解开发商的主体资格、审查房屋预售许可证、规划许可证等文件，确定所购买的房屋是可分割出售，取得分户产权的产权式酒店（有些酒店只有一个独立的“大产权”，不允…

#3 法律知识问答.docx 块3

☆ 56% 低相关

情况，开发商应了解投资者的资信情况，而投资者应了解开发商的主体资格、审查房屋预售许可证、规划许可证等文件，确定所购买的房屋是可分割出售，取得分户产权的产权式酒店（有些酒店只有一个独立的“大产权”，不允…

#4 法律知识问答.docx 块3

☆ 56% 低相关

情况，开发商应了解投资者的资信情况，而投资者应了解开发商的主体资格、审查房屋预售许可证、规划许可证等文件，确定所购买的房屋是可分割出售，取得分户产权的产权式酒店（有些酒店只有一个独立的“大产权”，不允…

#5 法律知识问答.docx 块3

☆ 56% 低相关

情况，开发商应了解投资者的资信情况，而投资者应了解开发商的主体资格、审查房屋预售许可证、规划许可证等文件，确定所购买的房屋是可分割出售，取得分户产权的产权式酒店（有些酒店只有一个独立的“大产权”，不允…

盗窃犯的主体和客体分别是什么?

10:28

文档中提到了盗窃罪的犯罪客体，但未提到犯罪主体。

- 犯罪客体：盗窃罪侵犯的客体是公私财物的所有权。侵犯的对象是国家、集体或个人的财物，一般指动产，但也包括可与不动产分离的附着物，以及电力、煤气等能源。
- 犯罪主体：根据刑法理论，盗窃罪的犯罪主体是一般主体，即达到刑事责任年龄（16周岁）且具备刑事责任能力的自然人。已满14周岁不满16周岁的人，犯盗窃罪的，不承担刑事责任。

结合文档和知识

10:28 · 处理时间: 2.4s

相似度: 74%

参考文档 (5)

#1 未知文档 块0

☆ 46% 低相关

法律知识问答 1、问题：在法律中定金与订金的区别？答案：“定金”是指当事人约定由一方向对方给付的，作为债权担保的一定数额的货币，它属于一种法律上的担保方式，目的在于促使债务人履行债务，保...

#2 未知文档 块0

☆ 46% 低相关

法律知识问答 1、问题：在法律中定金与订金的区别？答案：“定金”是指当事人约定由一方向对方给付的，作为债权担保的一定数额的货币，它属于一种法律上的担保方式，目的在于促使债务人履行债务，保...

#3 未知文档 块0

☆ 46% 低相关

法律知识问答 1、问题：在法律中定金与订金的区别？答案：“定金”是指当事人约定由一方向对方给付的，作为债权担保的一定数额的货币，它属于一种法律上的担保方式，目的在于促使债务人履行债务，保...

#4 法律知识问答.docx 块3

☆ 74% 中等相关

情况，开发商应了解投资者的资信情况，而投资者应了解开发商的主体资格、审查房屋预售许可证、规划许可证等文件，确定所购买的房屋是可分割出售，取得分户产权的产权式酒店（有些酒店只有一个独立的“大产权”，不允...

☆ 74% 中等相关

#5 法律知识问答.docx 块3

10:28 · 处理时间: 2.4s

文件解析及分块：

对话历史 +

当前对话: 什么是定金? 定金和订金有什么区别?

什么是定金? 定金和订金有什么区别... •
① 刚刚 ② 8 条消息
文档中提到了盗窃罪的犯罪客体, ...

新对话
① 刚刚

穿扬科技 Google Cloud Partner 穿扬科技 张鹏 1357

AI助手在线 Powered by Vertex AI

文件管理 文档管理 智能文档管理与处理中心

● 文档总数 1 ● 已完成 1 ● 处理中 0 ● 文本块 4 处理成功率: 100%

搜索文档... 全部状态

法律知识问答.docx 8.98 KB • 2025/7/7 10:26:32
已完成 4 个文本块

对话历史 +

当前对话: 什么是定金? 定金和订金有什么区别?

什么是定金? 定金和订金有什么区别... •
① 刚刚 ② 8 条消息
文档中提到了盗窃罪的犯罪客体, ...

新对话
① 刚刚

穿扬科技 Google Cloud Partner 穿扬科技 张鹏 1357

AI助手在线 Powered by Vertex AI

法律知识问答.docx 8.98 KB • 2025/7/7 10:26:33
4 文本块 14 词数 1995 字符数 499 平均块长度

完整文本 分块查看 搜索内容... 全部状态

处理成功率: 100%

完整文档内容

法律知识问答

1、问题：在法律中定金与订金的区别？
答案：“定金”是指当事人约定由一方对方给付的，作为债权担保的一定数额的货币，它属于一种法律上的担保方式，目的在于促使债务人履行债务，保障债权人的债权得以实现。签合同时，对定金必需以书面形式进行约定，同时还应约定定金的数额和交付期限。给付定金一方如果不履行债务，无权要求另一方返还定金；接受定金的一方如果不履行债务，需向另一方双倍返还债务。债务人履行债务后，依照约定，定金应抵作价款或者收回。而“订金”目前我国法律没有明确规定，它不具备定金所具有的担保性质，可视为“预付款”，当合同不能履行时，除不可抗力外，应根据双方当事人的过错承担违约责任。

2、问题：盗窃罪的犯罪客体是什么，盗窃罪的犯罪主体是什么？
答案：盗窃罪的客体要件本罪侵犯的客体是公私财物的所有权。侵犯的对象，是国家、集体或个人的财物，一般是不动产而言，但不动产上之附着物，可与不动产分离的，例如，田地上的农作物，山上的树木、建筑物上之门窗等，也可以成为本罪的对象。另外，能源如电力、煤气也可成为本罪的对象。盗窃罪侵犯的客体是公私财物的所有权。所有权包括占有、使用、收益、处分等权能。这里的所有权一般指合法的所有权，但有时也有例外情况。根据《最高人民法院关于审理盗窃案件具体应用法律若干问题的解释》（以下简称《解释》）的规定：“盗窃违禁品，按盗窃罪处理的，不计数额，根据情节轻重量刑。盗窃违禁品或犯罪分子不法占有的财物也构成盗窃罪。”

3、问题：我国跨境电商产品侵权应该注意什么问题？
答案：一般电商平台上销售的商品包括自营商品和作为平台展示的网络卖家所销售的商品两种类型。自营商品由于电商平台能够做到严格审核，对商品的采购、运输、仓储、销售等拥有较高控制度，发生知识产权侵权投诉的概率相对较低，并且可直接适用相关知识产权法律规定，因此引发的争议

资源创建流程及部署流程：

标准流程：

第一步：选择vertexai-> agentbuilder-> rag engine ->创建corpus

Name	Status	Description	Vector database	Created
rag-engine-polymeric	Initialized	基于图像和文本进行rag回答	Vertex AI Vector Search	Jun 25, 2025, 6:56:31 PM

第二步：创建数据源或选择自定义关联的数据源

Create corpus

Import data
Some form fields are incorrect

Configure vector store

Create corpus

Data
Select where you want to upload data from

- Select from Google Cloud Storage
- Select from Google Drive
- Import from Slack
- Import from Jira
- Import from SharePoint

Chunking strategy

Chunking size — ⓘ
The number of words to include in a chunk. The recommended value is 1024.

Chunk overlap — ⓘ
Chunks have a certain amount of overlap to improve relevance and retrieval quality. The recommended value is 256.

Layout parser
The layout parser extracts content elements from the document, and then creates context-aware chunks that facilitate information retrieval in generative AI and discovery applications.

- Default parsing libraries
Basic libraries that support extracting texts from documents.
- LLM parser
Advanced parser that uses LLM models to understand and interpret semantic content across various formats (text, image, diagrams).
[Learn more ↗](#)
- Document AI layout parser
Extracts content elements from the document, such as text, tables and lists.
[Learn more ↗](#)

[Show less ↕](#)

默认解析库 (Default parsing libraries)

- **含义：**这是支持从文档中提取文本的基本库。它主要用于执行文本提取的基础功能。

LLM 解析器 (LLM parser)

- **含义：**这是一种高级解析器，它使用大型语言模型（LLM）来理解和解释跨各种格式（文本、图像、图表）的语义内容。这意味着它能够更深入地理解文档内容，包括非文本信息。
 - **了解更多：** [LLM parser](#) [Learn more](#)

Document AI 布局解析器 (Document AI layout parser)

- **含义：**这种解析器专门用于提取文档中的内容元素，例如文本、表格和列表。它特别适用于需要从结构化和半结构化文档中提取特定数据（如表格数据）的场景。
 - **了解更多：** [Document AI layout parser](#) [Learn more](#)

第三步：配置向量模型以及向量存储。

Create corpus

Import data

Configure vector store

Create corpus

Embedding model

Learn more on the different embedding models ↗

Embedding model
Text Multilingual Embedding 002

Vector database

Vector database to store and query vector embeddings.

Google Cloud products

RagManaged vector store Preview

Fully managed, great for a quick start

Vertex AI Feature Store Preview

Cost-effective and scalable option that leverages BigQuery

Vertex AI Vector Search

Optimized for machine learning tasks that are highly scalable and reliable

Third party vector stores

Pinecone

Use a vector database from Pinecone

Weaviate

Use a vector database from Weaviate

向量模型可以使用系统的embedding模型，也可以使用微调模型

Embedding model

Learn more on the different embedding models ↗

Embedding model

Filter Type to filter

Vertex AI text embedding models

Text Multilingual Embedding 002 Best for overall languages

Text Embedding 005

Text Embedding 004

Fine-tuned and OSS models

chicago-taxi-endpoint

Vertex AI Vector Search

Optimized for machine learning tasks that are highly scalable and reliable

Third party vector stores

Pinecone

Use a vector database from Pinecone

Weaviate

Use a vector database from Weaviate

第四步：创建 vertex ai serch 应用类型

The screenshot shows the Vertex AI Search interface. On the left, there's a sidebar with sections like Vertex AI Studio, Agent Builder, and Data. The main area is titled "Vertex AI Search" and shows four options to "Create a new Search app":

- Custom search (general)**: Build tailored search, personalization and generative experiences on your sites, content, catalog, and blended data.
- Site search with AI mode**: Build generative search with AI mode in a day using site content.
- Media search**: Search engine designed for media libraries, including images, videos, and audio files.
- Search for commerce**: If you're looking to build a search app for your retail catalog, we recommend using the Retail Search API.

Each option has a "Create" button. At the bottom right of the main area, there's a "See all" link.

以custom search举例：

The screenshot shows the "Create Search App" configuration page. On the left, there's a sidebar with "AI Applications" and "Apps" selected. The main form has tabs: "Type" (selected), "Configuration", and "Data".

Search app configuration
Configure your app settings

Enterprise edition features

In addition to the standard features, you get:

- Extractive answers: answers that are extracted verbatim from your documents.
- Image search: where you can use an image as a query.
- Website search.
- Core generative answers.

Turning on Enterprise edition features is required for website search. To get higher refresh frequency, lower latency, search summaries, and more features in addition to website data, you need to turn on Advanced website indexing. You can change this setting at any time. After turning on Enterprise features, it can take up to 5 minutes for the features to become available.

[Learn more about features and prices ↗](#)

Advanced LLM features

For structured, unstructured, and advanced website search, you get:

- Search summarization.
- Search with follow-ups.
- Advanced generative answers.

Advanced LLM features are not available for basic website search. You can change this setting at any time. After turning on Advanced LLM features, it can take up to 5 minutes for the features to become available.

[Learn more about features and prices ↗](#)

Your app name

App name *
The ID will be generated based on name. It cannot be changed later.

External name of your company or organization

Company name *
Providing your company name helps the model provide higher-quality responses.

Location of your app

We recommend that you choose the global location, if you do not have compliance or regulatory reasons to locate your data in a particular multi-region. (EU and US regions are currently in preview)

Multi-region *
global (Global)

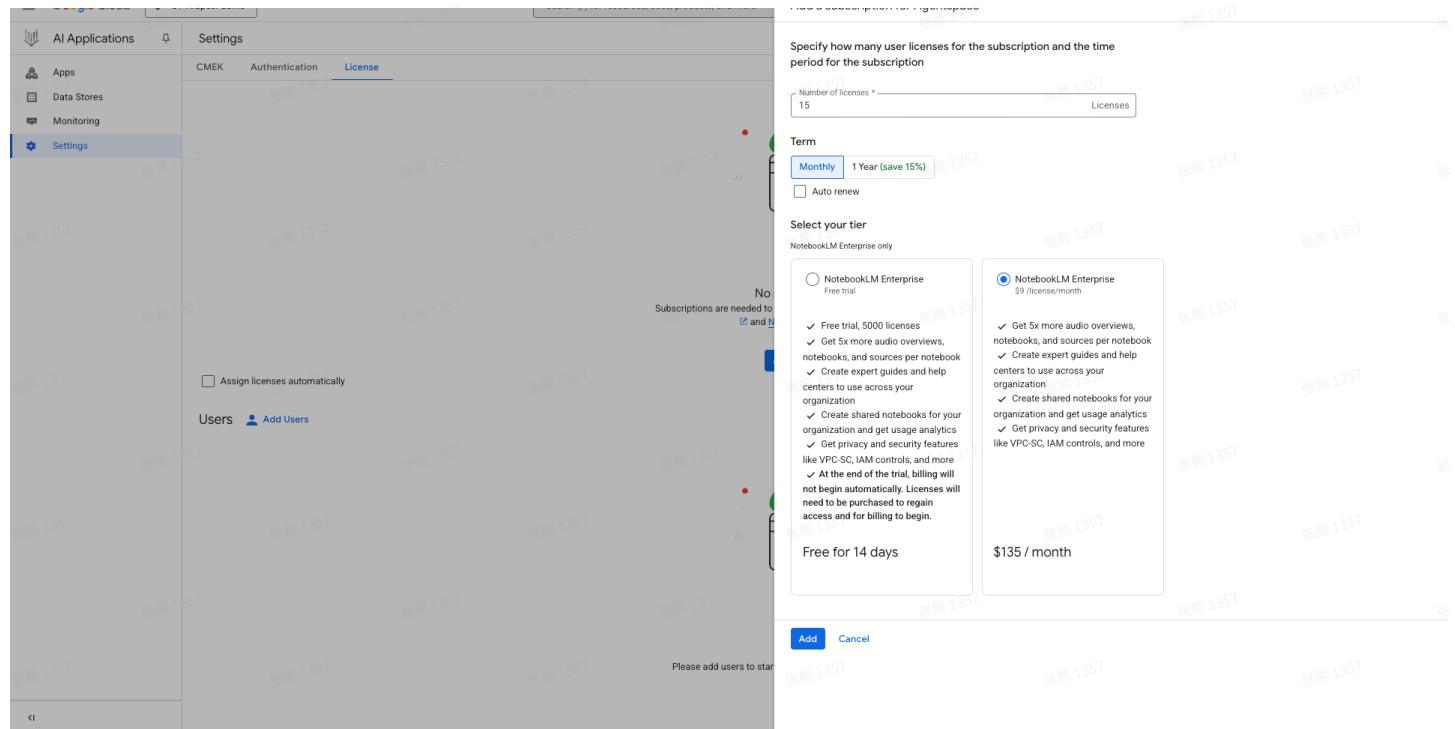
可以创建一个新的应用 也可以在一个已经存在的app中进行创建和绑定相应的向量数据库

第五步：绑定数据源，配置完毕后，则一个基于rag 向量书库检索的应用引擎则完成。他会将bucket下的数据，进行文本处理，分块解析后，存入到数据库然后利用相应的模型进行问答。

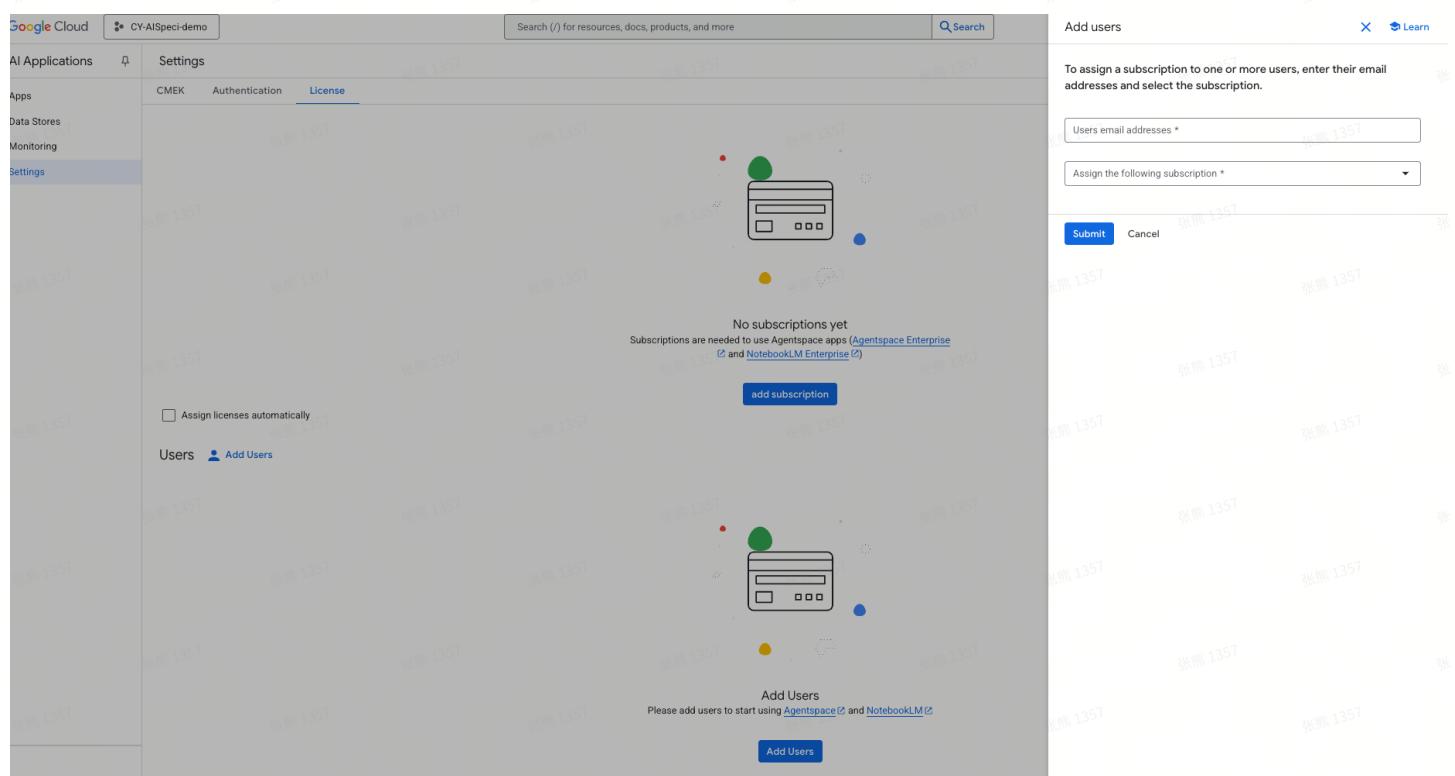
也可以使用一个已经存在的app进行数据源的绑定，进行向量检索和问答应用构建，此处省略。

相应的监控，则在monitoring中可以看到

如果需要相应的人访问对应的rag引擎的app，则需要购买证书。



然后为相应的用户配置证书即可。



自定义流程

第一步选择workbench，创建实例：

Google Cloud | qwiklabs-gcp-04-0380eb7c441f | AI Workbench

Vertex AI / Workbench

Workbench [Create New](#) [Refresh](#)

Instances Executions Schedules

View: [Instances](#) User-managed Notebooks Managed Notebooks

JupyterLab 4 is now available in Vertex AI Workbench.

Workbench Instances have JupyterLab 3 pre-installed and are configured with GPU-enabled machine learning frameworks. [Learn more](#)

Filter

Instance name	Zone	Auto upgrade	Version	Machine type

You don't have any instances in this project.

[Create New](#)

New instance

Name * student-workbench-instance

Region * us-west4 (Las Vegas) Zone * us-west4-a

Some regions are restricted due to a policy set by your organization. [Learn more](#)

Attach 1 NVIDIA T4 GPU

Enable Dataproc Serverless Interactive Sessions

Enable access to Dataproc Spark kernels

Network in this project

Shared network

Network default

Subnetwork * default(10.182.0.0/20)

Instance properties

Machine type e2-standard-4

Data disk 100 GB Balanced persistent disk

Permission Compute Engine default service account

Estimated cost \$162.00 monthly, \$0.22 hourly

[Advanced options](#) [Cancel](#) [Create](#)

Google Cloud | qwiklabs-gcp-04-0380eb7c441f | AI Workbench

Vertex AI / Workbench

Workbench [Create New](#) [Refresh](#)

Instances Executions Schedules

View: [Instances](#) User-managed Notebooks Managed Notebooks

JupyterLab 4 is now available in Vertex AI Workbench.

Workbench Instances have JupyterLab 3 pre-installed and are configured with GPU-enabled machine learning frameworks. [Learn more](#)

Filter

Instance name	Zone	Auto upgrade	Version	Machine type

You don't have any instances in this project.

[Create New](#)

New instance

Name * student-workbench-instance

Region * us-west4 (Las Vegas) Zone * us-west4-a

Some regions are restricted due to a policy set by your organization. [Learn more](#)

Attach 1 NVIDIA T4 GPU

Enable Dataproc Serverless Interactive Sessions

Enable access to Dataproc Spark kernels

Network in this project

Shared network

Network default

Subnetwork * default(10.182.0.0/20)

Instance properties

Machine type n1-standard-4

Data disk 100 GB Balanced persistent disk

Permission Compute Engine default service account

Estimated cost \$375.48 monthly, \$0.51 hourly

[Advanced options](#) [Cancel](#) [Create](#)

Google Cloud | qwiklabs-gcp-04-0380eb7c441f | Vertex AI / Workbench / Instances / Create instance

Create instance

Details

Name * student-workbench-instance

Must start with a letter followed by up to 47 lowercase letters, numbers, or hyphens (-) and cannot end with a hyphen

Region * us-west4 (Las Vegas) Zone * us-west4-a

Some regions are restricted due to a policy set by your organization. [Learn more](#)

Enable Dataproc Serverless Interactive Sessions

Enable access to Dataproc Spark kernels

Pricing summary

\$375.48 monthly estimate

That's about \$0.514 hourly

Pay for what you use: No upfront costs and per second billing

Networking cost also applies. [Learn more](#)

Details

Labels

+ Add label

Network tags

Network tags

Assign tags to your Workbench resource. [Learn more](#)

Workbench type

Instead of creating a Vertex AI Workbench instance, you can create a user-managed or managed notebook using the same configuration. These are older versions of Workbench and not recommended unless you have a specific need.

Type * Instance

[Continue](#)

Recommended for you

[Introduction to Vertex AI Workbench](#)

Help document

Vertex AI Workbench lets you perform your data science workflow in a JupyterLab notebook-based development environment.

[Create a new Vertex AI Workbench instance](#)

Help document

Create a new Vertex AI Workbench instance with the latest machine learning and data science libraries installed.

[Query data in BigQuery from within JupyterLab](#)

Help document

Access BigQuery data without leaving the JupyterLab interface.

[Use cases for Vertex AI](#)

Help document

Explore use cases, best practices, and industry solutions.

[Terraform samples](#)

Help document

See examples of using Terraform to create Vertex AI resources.

[Architecture guides for AI and machine learning](#)

Help document

Discover best practices and reference architectures for AI and machine learning.

All Vertex AI documentation

[Vertex AI / Workbench / Instances / Create instance](#)

Tools

- Dashboard
- Model Garden
- Pipelines
- Notebooks
- Colab Enterprise
- Workbench**
- Vertex AI Studio
- Overview
- Create prompt
- Media Studio
- Stream realtime
- Prompt gallery
- Prompt management
- Tuning
- Agent Builder
- Agent Garden
- Agent Engine
- RAG Engine
- Vertex AI Search
- Vector Search
- Data
- Provisioned Throughput
- Tutorial

[Create instance](#)

Environment

All environments use JupyterLab 3 by default and have the latest NVIDIA GPU and Intel libraries and drivers installed. You can specify a previous version instead. [Learn more](#)

JupyterLab Version

- JupyterLab 3.x
- JupyterLab 4.x [New](#)

Use custom container

Version

- Use the latest version
- Use a previous version

To learn more about specific versions, see the [Vertex AI Workbench release notes](#).

Post-startup script

Path to post-startup script [Browse](#)

Cloud Storage path to script that automatically runs after the instance boots up.

Metadata

Some metadata keys including `data-disk-uri`, `framework`, `notebooks-api`, `notebooks-api-version`, `nvidia-driver-gcs-path`, `proxy-uri`, `restriction`, `shutdown-script`, `title`, `version` are reserved for system use only. If you use these variable names below, they will be overwritten by system values.

[+ Add metadata](#)

[Back](#) [Continue](#)

Recommended for you

[Introduction to Vertex AI Workbench](#)

Help document

Vertex AI Workbench lets you perform your data science workflow in a JupyterLab notebook-based development environment.

[Create a new Vertex AI Workbench instance](#)

Help document

Create a new Vertex AI Workbench instance with the latest machine learning and data science libraries installed.

[Query data in BigQuery from within JupyterLab](#)

Help document

Access BigQuery data without leaving the JupyterLab interface.

[Use cases for Vertex AI](#)

Help document

Explore use cases, best practices, and industry solutions.

[Terraform samples](#)

Help document

See examples of using Terraform to create Vertex AI resources.

[Architecture guides for AI and machine learning](#)

Help document

Discover best practices and reference architectures for AI and machine learning.

[Google Cloud](#) [qwiklabs-gcp-04-0380eb7c441f](#)

[Vertex AI / Workbench / Instances / Create instance](#)

Tools

- Dashboard
- Model Garden
- Pipelines
- Notebooks
- Colab Enterprise
- Workbench**
- Vertex AI Studio
- Overview
- Create prompt
- Media Studio
- Stream realtime
- Prompt gallery
- Prompt management
- Tuning
- Agent Builder
- Agent Garden
- Agent Engine
- RAG Engine
- Vertex AI Search
- Vector Search
- Data
- Provisioned Throughput
- Tutorial

[Create instance](#)

Machine type

General purpose GPUs

Machine types for common workloads, optimized for cost and flexibility

Series	Description	vCPUs	Memory
E2	Low cost, day-to-day computing	2 - 32	4 - 128 GB
N2	Balanced price & performance	2 - 128	4 - 864 GB
N2D	Balanced price & performance	2 - 224	4 - 896 GB
<input checked="" type="radio"/> N1	Balanced price & performance	2 - 96	3.6 - 624 GB

Machine type: **n1-standard-4** (4 vCPU, 2 core, 15 GB memory)

vCPU: 4

Memory: 15 GB

GPUs

The number of attached GPUs affects the VM's maximum number of memory and CPUs. [Learn More](#)

GPU type: NVIDIA T4 Number of GPUs: 1

CPU platform and GPU Reservations

Reservations: Don't use

Use an existing Compute Engine reservation when creating this Notebook.

Shielded VM

Turn on all settings for the most secure configuration. [Learn more](#)

Secure Boot

Created bucket qwiklabs-gcp-04-0380eb7c441f-model-dataset

Create instance

Disks

Boot disk type: **Balanced Persistent Disk** | Boot disk size in GB: **150**

Data disk type: **Balanced Persistent Disk** | Data disk size in GB: **100**

Delete to trash: Deleting in JupyterLab will use the operating system's trash behavior.

Encryption

Google-managed encryption key: Keys owned by Google.

Cloud KMS key: Keys owned by customers.

Pricing summary

\$375.48 monthly estimate
That's about \$0.514 hourly
Pay for what you use: No upfront costs and per second billing
Networking cost also applies. [Learn more](#)

Details

Back **Continue**

Recommended for you

- [Introduction to Vertex AI Workbench](#) Help document
Vertex AI Workbench lets you perform your data science workflow in a JupyterLab notebook-based development environment.
- [Create a new Vertex AI Workbench instance](#) Help document
Create a new Vertex AI Workbench instance with the latest machine learning and data science libraries installed.
- [Query data in BigQuery from within JupyterLab](#) Help document
Access BigQuery data without leaving the JupyterLab interface.
- [Use cases for Vertex AI](#) Help document
Explore use cases, best practices, and industry solutions.
- [Terraform samples](#) Help document
See examples of using Terraform to create Vertex AI resources.
- [Architecture guides for AI and machine learning](#) Help document
Discover best practices and reference architectures for AI and machine learning.

Create instance

Networking

The instance requires internet access to be used. Make sure one of the following is selected. [Learn more](#)

- + Assign an external IP address
- + Select a network that has internet access
- + Turn on [Private Google Access](#)

Network in this project

Shared network

Network: **default**

Subnetwork: **default(10.182.0.0/20)**

Assign external IP address: Lets the instance communicate with the internet or other VPC networks.

Allow proxy access: If proxy access is disabled, you must SSH to access your instance. [Learn more](#)

Pricing summary

\$375.48 monthly estimate
That's about \$0.514 hourly
Pay for what you use: No upfront costs and per second billing
Networking cost also applies. [Learn more](#)

Details

Back **Continue**

Recommended for you

- [Introduction to Vertex AI Workbench](#) Help document
Vertex AI Workbench lets you perform your data science workflow in a JupyterLab notebook-based development environment.
- [Create a new Vertex AI Workbench instance](#) Help document
Create a new Vertex AI Workbench instance with the latest machine learning and data science libraries installed.
- [Query data in BigQuery from within JupyterLab](#) Help document
Access BigQuery data without leaving the JupyterLab interface.
- [Use cases for Vertex AI](#) Help document
Explore use cases, best practices, and industry solutions.
- [Terraform samples](#) Help document
See examples of using Terraform to create Vertex AI resources.
- [Architecture guides for AI and machine learning](#) Help document
Discover best practices and reference architectures for AI and machine learning.

Create instance

IAM and security

Determines who can use the instance's JupyterLab interface. This cannot be changed after the instance is created. [Learn more](#)

Service account: Anyone with the iam.serviceAccounts.actAs can access the instance account.

Single user: Restricts access to one user.

Use default Compute Engine service account

Security options

Root access to the instance

nbconvert: Export and download notebooks as a different file type

File downloading: Allow downloads from JupyterLab

Terminal access: Run shell commands from JupyterLab

Pricing summary

\$375.48 monthly estimate
That's about \$0.514 hourly
Pay for what you use: No upfront costs and per second billing
Networking cost also applies. [Learn more](#)

Details

Back **Continue**

Recommended for you

- [Introduction to Vertex AI Workbench](#) Help document
Vertex AI Workbench lets you perform your data science workflow in a JupyterLab notebook-based development environment.
- [Create a new Vertex AI Workbench instance](#) Help document
Create a new Vertex AI Workbench instance with the latest machine learning and data science libraries installed.
- [Query data in BigQuery from within JupyterLab](#) Help document
Access BigQuery data without leaving the JupyterLab interface.
- [Use cases for Vertex AI](#) Help document
Explore use cases, best practices, and industry solutions.
- [Terraform samples](#) Help document
See examples of using Terraform to create Vertex AI resources.
- [Architecture guides for AI and machine learning](#) Help document
Discover best practices and reference architectures for AI and machine learning.

Google Cloud | qwiklabs-gcp-00-0380eb7c441f | AI Workbench | Search | Recommended for you

Vertex AI / Workbench / Instances / Create Instance

Tools

- Dashboard
- Model Garden
- Pipelines
- Notebooks
- Colab Enterprise
- Workbench
- Vertex AI Studio
- Overview
- Create prompt
- Media Studio
- Stream realtime
- Prompt gallery
- Prompt management
- Tuning

Agent Builder

- Agent Garden
- Agent Engine
- RAG Engine
- Vertex AI Search

Create instance

System health

- Details
- Environment
- Machine type
- Disks
- Networking
- IAM and security
- System health

Pricing summary

\$375.48 monthly estimate
That's about \$0.514 hourly
Pay for what you use: No upfront costs and per second billing
Networking cost also applies.

Reporting

- Report system health
- Report custom metrics to Cloud Monitoring
- Install Cloud Monitoring
- Report DNS status for required Google domains

Details

Back

Learn

Introduction to Vertex AI Workbench

Create a new Vertex AI Workbench instance

Query data in BigQuery from within JupyterLab

Use cases for Vertex AI

Terraform samples

Architecture guides for AI and ML

Google Cloud | qwiklabs-gcp-00-0380eb7c441f | Search | Search | Recommended for you

Vertex AI / Workbench / Instances

Tools

- Dashboard
- Model Garden
- Pipelines
- Notebooks
- Colab Enterprise
- Workbench
- Vertex AI Studio
- Overview
- Create prompt
- Media Studio
- Stream realtime
- Prompt gallery
- Prompt management
- Tuning

Agent Builder

- Agent Garden
- Agent Engine
- RAG Engine
- Vertex AI Search

Workbench

Create New Refresh

Instances Executions Schedules

View: Instances User-managed Notebooks Managed Notebooks

JupyterLab 4 is now available in Vertex AI Workbench. Dismiss

Workbench Instances have JupyterLab 3 pre-installed and are configured with GPU-enabled machine learning frameworks. Learn more

Instance name	Zone	Auto upgrade	Version	Machine Type	GPUs	Owner	Created
student-workbench-instance	us-west4-a	—	M130	4 vCPUs, 15 GB RAM	NVIDIA T4 x 1	679098231003-compute@developer.gserviceaccount.com	Jun 19, 2025, 10:41:26 AM

Filter

Learn

Introduction to Vertex AI Workbench

Create a new Vertex AI Workbench instance

Query data in BigQuery from within JupyterLab

Use cases for Vertex AI

Terraform samples

particular.googleusercontent.com/gcpcatalogs/125121catalog_rank_0501_rank_A5A1A2C_hmz_inter_A5A0712C_hmz_search_A5A1B07D_expandedcatalogsearchCDEB0510553

← [PTPPF]: Tune Gemini Model by using Supervised Fine-tuning

提示：在运行实验之前，请先查看前提条件。 >

结束实验 02:50:55

注意：在控制台中操作时，请务必严格遵循实验说明。否则，您的账号可能会被屏蔽。[了解详情](#)

Open Google Cloud console

Username: student-03-f4f2be2a7e84i

Password: bw0WKVAtXZZn

Project ID: qwiklabs-gcp-00-4e744f01

Click Check my progress to verify the objective.

Create a Cloud Storage bucket to store your Vertex AI model data

检查我的进度 Step Completed!

Task 2. Deploy a Vertex AI Workbench instance

In this task, you must create and deploy a Vertex AI Workbench instance.

- Enable all recommended Vertex AI APIs.
- Create a new Vertex AI Workbench instance named `cepf-workbench` instance in the `europe-west4-b` zone.

Click Check my progress to verify the objective.

Deploy a Vertex AI Workbench instance

检查我的进度 Step Completed!

Learn

PTPPF066 Challenge lab overview 10/100

Setup and requirements

Challenge scenario

Task 1. Create a Cloud Storage bucket to store your Vertex AI model data

Task 2. Deploy a Vertex AI Workbench instance

Task 3. Copy the Jupyter Notebook template for the lab

Task 4. Generate the training and validation dataset files

Task 5. Start a supervised tuning job using Gemini

Task 6. Test the tuned model with a prompt

Congratulations!

第二步：打开jupyterlab，构建自定义jupyter

1. 下载文本和图片数据

加载模型、下载辅助函数，并从 Cloud Storage 中获取文档和图片 Download custom Python utilities & required files

代码块

```
1 !gsutil -m rsync -r gs://github-
repo/rag/intro_multimodal_rag/intro_multimodal_rag_old_version .
2 print("Download completed")
```

The screenshot shows a Jupyter Notebook interface. On the left, there is a file browser with a list of files: data, class_a_share.png, intro_multimodal_rag_utils.py, intro_multimodal_rag-v2.0.ipynb, notebook_template.ipynb, requirements.txt, and tac_table_revenue.png. The 'data' folder is expanded. In the main area, there is a code cell with the following content:

```
from vertexai.generative_models import GenerationConfig, GenerativeModel, Image
Load the Gemini model
[4]: text_model = GenerativeModel("gemini-2.0-flash")
multimodal_model = text_model
multimodal_model_flash = text_model

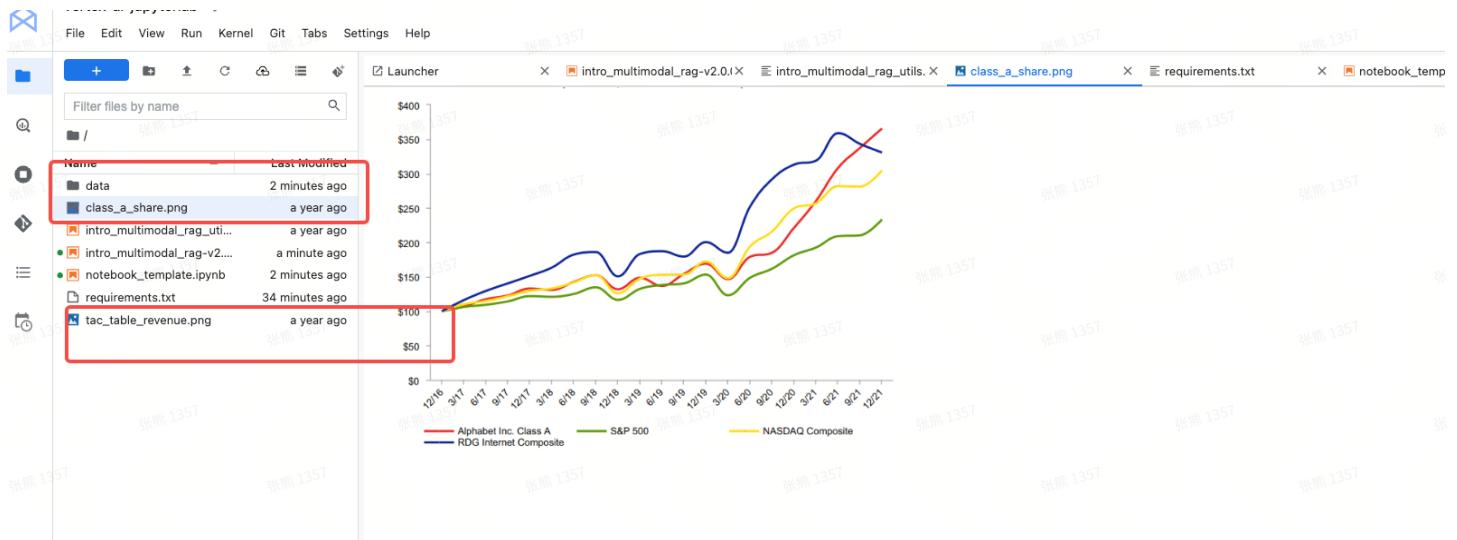
Download custom Python utilities & required files
The cell below will download a helper functions needed for this notebook, to improve readability. It also downloads other required files. You can also view the code for the utils here: (intro_multimodal_rag_utils.py) directly on GitHub.

[5]: # download documents and images used in this notebook
!gsutil -m rsync -r gs://github-repo/rag/intro_multimodal_rag/intro_multimodal_rag_old_version .
print("Download completed")
```

Output of the code cell:

```
Building synchronization state...
Starting synchronization...
Copying gs://github-repo/rag/intro_multimodal_rag/intro_multimodal_rag_old_version/tac_table_revenue.png...
Copying gs://github-repo/rag/intro_multimodal_rag/intro_multimodal_rag_old_version/class_a_share.png...
Copying gs://github-repo/rag/intro_multimodal_rag/intro_multimodal_rag_old_version/data/google-10k-sample-part1.pdf...
Copying gs://github-repo/rag/intro_multimodal_rag/intro_multimodal_rag_old_version/data/google-10k-sample-part2.pdf...
Copying gs://github-repo/rag/intro_multimodal_rag/intro_multimodal_rag_old_version/intro_multimodal_rag_utils.py...
- [5/5 files] 882.3 KiB/882.3 KiB 100% Done
Operation completed over 5 objects@882.3 KiB.
Download completed
```

Below the code cell, there is a section titled "Building metadata of documents containing text and images".



2 数据embedding，从文档中提取并存储文本和图像元数据。

代码块

```
1 text_model = GenerativeModel("gemini-2.0-flash")
2 multimodal_model = text_model
3 multimodal_model_flash = text_model
4
```

```
5
6 from intro_multimodal_rag_utils import get_document_metadata
7 # Specify the PDF folder with multiple PDF
8
9 # pdf_folder_path = "/content/data/" # if running in Google Colab/Colab
Enterprise
10 pdf_folder_path = "data/" # if running in Vertex AI Workbench.
11
12 # Specify the image description prompt. Change it
13 image_description_prompt = """Explain what is going on in the image.
14 If it's a table, extract all elements of the table.
15 If it's a graph, explain the findings in the graph.
16 Do not include any numbers that are not mentioned in the image.
17 """
18
19 # Extract text and image metadata from the PDF document
20 text_metadata_df, image_metadata_df = get_document_metadata(
21     multimodal_model, # we are passing Gemini 2.0 model
22     pdf_folder_path,
23     image_save_dir="images",
24     image_description_prompt=image_description_prompt,
25     embedding_size=1408,
26     # add_sleep_after_page = True, # Uncomment this if you are running into
API quota issues
27     # sleep_time_after_page = 5,
28     # generation_config = # see next cell
29     # safety_settings = # see next cell
30 )
31
32 print("\n\n --- Completed processing. ---")
```

```
Processing the file: ----- data/google-10k-sample-part1.pdf
```

```
Processing page: 1  
Processing page: 2  
Extracting image from page: 2, saved as: images/google-10k-sample-part1.pdf_image_1_0_11.jpeg  
Processing page: 3  
Extracting image from page: 3, saved as: images/google-10k-sample-part1.pdf_image_2_0_15.jpeg  
Processing page: 4  
Extracting image from page: 4, saved as: images/google-10k-sample-part1.pdf_image_3_0_18.jpeg  
Processing page: 5  
Extracting image from page: 5, saved as: images/google-10k-sample-part1.pdf_image_4_0_21.jpeg  
Processing page: 6  
Processing page: 7
```

```
Processing the file: ----- data/google-10k-sample-part2.pdf
```

```
Processing page: 1  
Extracting image from page: 1, saved as: images/google-10k-sample-part2.pdf_image_0_0_6.jpeg  
Extracting image from page: 1, saved as: images/google-10k-sample-part2.pdf_image_0_1_8.jpeg  
Processing page: 2  
Extracting image from page: 2, saved as: images/google-10k-sample-part2.pdf_image_1_0_13.jpeg  
Processing page: 3  
Processing page: 4  
Extracting image from page: 4, saved as: images/google-10k-sample-part2.pdf_image_3_0_19.jpeg  
Processing page: 5  
Extracting image from page: 5, saved as: images/google-10k-sample-part2.pdf_image_4_0_22.jpeg  
Extracting image from page: 5, saved as: images/google-10k-sample-part2.pdf_image_4_1_23.jpeg
```

text_metadata_df.head()							
[7]:	file_name	page_num	text	text_embedding_page	chunk_number	chunk_text	text_embedding_chunk
0	google-10k-sample-part1.pdf	1	source: https://abc.xyz/assets/investor/static...	[0.06604167073965073, 0.054615460336208344, -0...	1	source: https://abc.xyz/assets/investor/static...	[0.05712887644767761, 0.06096643581986427, -0...
1	google-10k-sample-part1.pdf	1	source: https://abc.xyz/assets/investor/static...	[0.06604167073965073, 0.054615460336208344, -0...	2	of Record]nAs of December 31, 2021, there wer...	[0.030928857624530792, 0.02287178672850132, -0...
2	google-10k-sample-part1.pdf	2	Issuer Purchases of Equity Securities\nThe fol...	[0.035787057131528854, 0.008179700933396816, -...	1	Issuer Purchases of Equity Securities\nThe fol...	[0.035787057131528854, 0.008179700933396816, -...
3	google-10k-sample-part1.pdf	3	Stock Performance Graphs\nThe graph below matc...	[0.04338429123163223, 0.024151558056473732, -0...	1	Stock Performance Graphs\nThe graph below matc...	[0.04338429123163223, 0.024151558056473732, -0...
4	google-10k-sample-part1.pdf	4	The graph below matches Alphabet Inc. Class A'...	[0.05739395692944527, 0.02924434281885624, -0...	1	The graph below matches Alphabet Inc. Class A'...	[0.05739395692944527, 0.02924434281885624, -0...

Inspect the processed image metadata

Inspect the processed image metadata

The following cell will produce a metadata table which describes the different parts of image metadata, including:

- **img_desc:** Gemini-generated textual description of the image.
- **mm_embedding_from_text_desc_and_img:** Combined embedding of image and its description, capturing both visual and textual information.
- **mm_embedding_from_img_only:** Image embedding without description, for comparison with description-based analysis.
- **text_embedding_from_image_description:** Separate text embedding of the generated description, enabling textual analysis and comparison.

image_metadata_df.head()							
[8]:	file_name	page_num	img_num	img_path	img_desc	mm_embedding_from_img_only	text_embedding_from_image_description
0	google-10k-sample-part1.pdf	2	1	images/google-10k-sample-part1.pdf_image_1_0_1...	The image is a table that shows share repurchase...	[0.0232506078, -0.000799618, 0.0152797047, -0...	[0.034899670630693436, 0.012614904902875423, -...
1	google-10k-sample-part1.pdf	3	1	images/google-10k-sample-part1.pdf_image_2_0_1...	The image is a graph comparing the cumulative ...	[0.00688917283, 0.0237245914, -0.00902850833, -...	[0.036557674407958984, 0.028885135427117348, -...
2	google-10k-sample-part1.pdf	4	1	images/google-10k-sample-part1.pdf_image_3_0_1...	The image is a graph comparing the cumulative ...	[0.00658653351, 0.010452861, -0.00870141853, 0...	[0.036007337272167206, 0.04106100648641586, -...
3	google-10k-sample-part1.pdf	5	1	images/google-10k-sample-part1.pdf_image_4_0_2...	The image is a table showing financial data fo...	[0.0166553948, 0.01998970866, -0.0204031914, -0...	[0.04557782784104347, 0.0320768803358078, -0...
4	google-10k-sample-part2.pdf	1	1	images/google-10k-sample-part2.pdf_image_0_0_6...	The image is a table showing data for the year...	[0.0357288234, 0.032441318, 0.0125657981, -0...	[0.011768336407840252, 0.030940620228648186, -...

Import the helper functions to implement RAG

3 文本检索：

```
1 query = "I need details for basic and diluted net income per share of Class A,  
Class B, and Class C share for google?"  
2  
3 # Matching user text query with "chunk_embedding" to find relevant chunks.  
4 matching_results_text = get_similar_text_from_query(  
5     query,  
6     text_metadata_df,  
7     column_name="text_embedding_chunk",  
8     top_n=3,  
9     chunk_text=True,  
10 )  
11  
12 # Print the matched text citations  
13 print_text_to_text_citation(matching_results_text, print_top=False,  
    chunk_text=True)  
14  
15  
16  
17 print("\n **** Result: ***** \n")  
18  
19 # All relevant text chunk found across documents based on user query  
20 context = "\n".join(  
21     [value["chunk_text"] for key, value in matching_results_text.items()])  
22 )  
23  
24 instruction = f"""Answer the question with the given context.  
25 If the information is not available in the context, just return "not available  
in the context".  
26 Question: {query}  
27 Context: {context}  
28 Answer:  
29 """  
30  
31 # Prepare the model input  
32 model_input = instruction  
33  
34 # Generate Gemini response with streaming output  
35 get_gemini_response(  
36     text_model, # we are passing Gemini  
37     model_input=model_input,  
38     stream=True,  
39     generation_config=GenerationConfig(temperature=0.2),  
40 )  
41  
42 print("\n **** Result: ***** \n")  
43  
44
```

```
45 query = "I need details for basic and diluted net income per share of Class A,  
46 Class B, and Class C share for google?"  
47 matching_results_image = get_similar_image_from_query(  
48     text_metadata_df,  
49     image_metadata_df,  
50     query=query,  
51     column_name="text_embedding_from_image_description", # Use image  
description text embedding  
52     image_emb=False, # Use text embedding instead of image embedding  
53     top_n=3,  
54     embedding_size=1408,  
55 )  
56  
57 # Markdown(print_text_to_image_citation(matching_results_image,  
58 print_top=True))  
58 print("\n **** Result: ***** \n")  
59  
60 # Display the top matching image  
61 display(matching_results_image[0]["image_object"])  
62  
63 print("\n **** Result: ***** \n")  
64  
65 # All relevant text chunk found across documents based on user query  
66 context = f"""Image: {matching_results_image[0]['image_object']}  
67 Description: {matching_results_image[0]['image_description']}  
68 """  
69  
70 instruction = f"""Answer the question in JSON format with the given context of  
Image and its Description. Only include value.  
71 Question: {query}  
72 Context: {context}  
73 Answer:  
74 """  
75  
76 # Prepare the model input  
77 model_input = instruction  
78  
79 # Generate Gemini response with streaming output  
80 Markdown(  
81     get_gemini_response(  
82         multimodal_model_flash, # we are passing Gemini 2.0 Flash  
83         model_input=model_input,  
84         stream=True,  
85         generation_config=GenerationConfig(temperature=1),  
86     )  
87 )
```

```
[12]: print("\n **** Result: **** \n")

# All relevant text chunk found across documents based on user query
context = "\n".join([
    [value["chunk_text"]] for key, value in matching_results_text.items()])
)

instruction = f"""Answer the question with the given context.
If the information is not available in the context, just return "not available in the context".
Question: {query}
Context: {context}
Answer:
"""

# Prepare the model input
model_input = instruction

# Generate Gemini response with streaming output
get_gemini_response(
    text_model, # we are passing Gemini
    model_input=model_input,
    stream=True,
    generation_config=GenerationConfig(temperature=0.2),
)

**** Result: ****
```

[12]: 'The net income per share amounts are the same for Class A, Class B, and Class C stock because the holders of each class are entitled to equal per share dividends or distributions in liquidation. The context provides that basic and diluted net income per share are computed using the two-class method. It also mentions that the computation of the diluted net income per share of Class A stock assumes the conversion of Class B stock, while the diluted net income per share of Class B stock does not assume the conversion of those shares. However, the specific details (actual values) for basic and diluted net income per share of Class A, Class B, and Class C shares for Google are not available in the context.\n'

You can see that it returned:

"The provided context does not include the details for basic and diluted net income per share of Class A, Class B, and Class C share for google."

This is expected as discussed previously. No other text chunk (total 3) had the information you sought. This is because the information is only available in the images rather than in the text part of the document. Next, let's see if you can solve this problem by leveraging Gemini and Multimodal Embeddings.

Note: We handcrafted examples in our document to simulate real-world cases where information is often embedded in charts, table, graphs, and other image-based elements and unavailable as plain text.

```
[13]: query = "I need details for basic and diluted net income per share of Class A, Class B, and Class C share for google?"
```

```
[14]: matching_results_image = get_similar_image_from_query(
    text_metadata_df,
    image_metadata_df,
    query=query,
    column_name="text_embedding_from_image_description", # Use image description text embedding
    image_emb=False, # Use text embedding instead of image embedding
    top_n=3,
    embedding_size=1408,
)

# Markdown(print_text_to_image_citation(matching_results_image, print_top=True))
print("\n **** Result: **** \n")

# Display the top matching image
display(matching_results_image[0]["image_object"])
```

**** Result: ****

Year Ended December 31,			
2020			
	Class A	Class B	Class C
Basic net income per share:			
Numerator			
Allocation of undistributed earnings	\$ 17,733	\$ 2,732	\$ 19,804
Denominator			
Number of shares used in per share computation	299,815	46,182	334,819
Basic net income per share	\$ 59.15	\$ 59.15	\$ 59.15
Diluted net income per share:			
Numerator			
Allocation of undistributed earnings for basic computation	\$ 17,733	\$ 2,732	\$ 19,804
Reallocation of undistributed earnings as a result of conversion of Class B to Class A shares	2,732	0	0
Reallocation of undistributed earnings	(180)	(25)	180
Allocation of undistributed earnings	\$ 20,285	\$ 2,707	\$ 19,984
Denominator			
Number of shares used in basic computation	299,815	46,182	334,819
Weighted-average effect of dilutive securities			
Add:			
Conversion of Class B to Class A shares outstanding	46,182	0	0
Restricted stock units and other contingently issuable shares	87	0	6,125

```
***** Result: *****  
[15]: {  
    "Basic net income per share - Class A": "$59.15",  
    "Basic net income per share - Class B": "$59.15",  
    "Basic net income per share - Class C": "$59.15",  
    "Diluted net income per share - Class A": "$58.61",  
    "Diluted net income per share - Class B": "$58.61",  
    "Diluted net income per share - Class C": "$58.61"  
}  
  
[ 1]: ## you can check the citations to probe further.  
## check the "image description:" which is a description extracted through Gemini which helped search our query.  
Markdown(print_text_to_image_citation(matching_results_image, print_top=True))
```

Image Search

4 图片搜索-Search similar image with image query

代码块

```
1 # You can find a similar image as per the images you have in the metadata.  
2 # In this case, you have a table (picked from the same document source) and  
# you would like to find similar tables in the document.  
3 image_query_path = "tac_table_revenue.png"  
4  
5 # Print a message indicating the input image  
6 print("****Input image from user:****")  
7  
8 # Display the input image  
9 Image.load_from_file(image_query_path)  
10  
11  
12 matching_results_image = get_similar_image_from_query(  
13     text_metadata_df,  
14     image_metadata_df,  
15     query=query, # Use query text for additional filtering (optional)  
16     column_name="mm_embedding_from_img_only", # Use image embedding for  
# similarity calculation  
17     image_emb=True,  
18     image_query_path=image_query_path, # Use input image for similarity  
# calculation  
19     top_n=3, # Retrieve top 3 matching images  
20     embedding_size=1408, # Use embedding size of 1408  
21 )  
22  
23 print("\n **** Result: **** \n")  
24  
25 # Display the Top Matching Image  
26 display(  
27     matching_results_image[0]["image_object"]  
28 )  
29  
30 print_text_to_image_citation(  
31     matching_results_image, print_top=True  
32 ) # Print citation details for the top matching image
```

```

33 # Display the top matching image object (Pillow Image)
34
35 # Check Other Matched Images (Optional)
36 # You can access the other two matched images using:
37
38 print("-----Matched Images-----\n")
39 display_images(
40 [
41     matching_results_image[0]["img_path"],
42     matching_results_image[1]["img_path"],
43 ],
44 resize_ratio=0.5,
45 )

```

`image.load_from_file(image_query_path)`

Input image from user:

	Year Ended December 31,	
	2020	2021
TAC	\$ 32,778	\$ 45,566
Other cost of revenues	51,954	65,373
Total cost of revenues	\$ 84,732	\$ 110,939
Total cost of revenues as a percentage of revenues	46.4 %	43.1 %

You expect to find tables (as images) that are similar in terms of "Other/Total cost of revenues."

You expect to find tables (as images) that are similar in terms of "Other/Total cost of revenues."

你期望找到在“其他/总收入成本”方面相似的表格（以图片形式呈现）。

**** Result: *****

	Year Ended December 31,	
	2020	2021
TAC	\$ 32,778	\$ 45,566
Other cost of revenues	51,954	65,373
Total cost of revenues	\$ 84,732	\$ 110,939
Total cost of revenues as a percentage of revenues	46.4 %	43.1 %

It did find a similar-looking image (table), which gives more detail about different revenue, expenses, income, and a few more details based on the given image. More importantly, both tables show numbers related to the "cost of revenue."

You can also print the citation to see what it has matched.

It did find a similar-looking image (table), which gives more detail about different revenue, expenses, income, and a few more details based on the given image. More importantly, both tables show numbers related to the "cost of revenue."

You can also print the citation to see what it has matched.

它确实找到了一张外观相似的图片（表格），该表格根据给定图片提供了关于不同收入、费用、收益的更多细节，还有一些其他详情。更重要的是，两张表格都展示了与“收入成本”相关的数据。

你也可以打印引用内容，查看它匹配到了什么。

代码块

```

1 # Check Other Matched Images (Optional)
2 # You can access the other two matched images using:
3
4 print("-----Matched Images-----\n")

```

```

5     display_images(
6         [
7             matching_results_image[0]["img_path"],
8             matching_results_image[1]["img_path"],
9         ],
10        resize_ratio=0.5,
11    )

```

-----Matched Images-----

	Year Ended December 31,	
	2020	2021
TAC	\$ 32,778	\$ 45,566
Other cost of revenues	\$ 51,954	\$ 65,373
Total cost of revenues	\$ 84,732	\$ 110,939
Total cost of revenues as a percentage of revenues	46.4 %	43.1 %

	Year Ended December 31,		\$ Change	% Change
	2020	2021		
Consolidated revenues	\$ 182,527	\$ 257,637	\$ 75,110	41 %
Change in consolidated constant currency revenues				39 %
Cost of revenues	\$ 84,732	\$ 110,939	\$ 26,207	31 %
Operating expenses	\$ 56,571	\$ 67,984	\$ 11,413	20 %
Operating income	\$ 41,224	\$ 78,714	\$ 37,490	91 %
Operating margin	23 %	31 %		8 %
Other income (expense), net	\$ 6,858	\$ 12,020	\$ 5,162	75 %
Net Income	\$ 40,269	\$ 76,033	\$ 35,764	89 %
Diluted EPS	\$ 58.61	\$ 112.20	\$ 53.59	91 %
Number of Employees	135,301	156,500	21,199	16 %

案例视频：

对话历史 +

穿扬科技 | Google Cloud Partner

AI助手在线 Powered by Vertex AI

暂无对话历史
点击右上角 + 开始新的对话

首页 对话 文件管理

AI智能问答助手

基于 Google Vertex AI 的企业级智能检索问答系统

智能对话 文档管理 实时检索

试试这些问题

什么是定金？定金和订金有什么区别？ → 赔偿的构成要件是什么？ →

合同违约的法律后果有哪些？ → 如何处理劳动争议？ →

开始使用:

Powered by Vertex AI