This query uses the LOAD DATA statement to load the customer_reviews.csv file from Cloud Storage to a BigQuery table with the given column names and data types.

```
LOAD DATA OVERWRITE CustomerReviewGenAI.customer_reviews
(customer_review_id INT64, customer_id INT64, location_id INT64, review_datetime
DATETIME, review_text STRING, social_media_source STRING, social_media_handle STRING)
FROM FILES (
 format = 'CSV',
 uris = ['gs://coffeeonwheels/customer_reviews.csv']);
```

Create a External connection with vertex ai say named as "gemini_conn"
-- Create connection from vertex AI federation bigquery

Connection info

Connection ID	projects/gcpdataengineering-467713/locations/us/connections/gemini_conn
Friendly name	
Created	6 Aug 2025, 22:02:58 UTC+5:30
Last modified	6 Aug 2025, 22:02:58 UTC+5:30
Data location	us
Description	
Connection type	Vertex AI remote models, remote functions and BigLake (Cloud Resource)
Service account ID	bqcx-897867797155-2nup@gcp-sa-bigquery-condel.iam.gserviceaccount.com

The review_images object table is added to the gemini_demo dataset in BigQuery, storing the Cloud Storage URIs for each audio review in the sample dataset. This allows BigQuery to reference and analyze unstructured data like audio files.

```
CREATE OR REPLACE EXTERNAL TABLE
 `CustomerReviewGenAI.review_images`
WITH CONNECTION `us.gemini_conn`
OPTIONS (
 object_metadata = 'SIMPLE',
 uris = ['gs://coffeeonwheels/images/*']
 );
```

Filter | Something went wrong.

	Field name	Type	Mode
☐	uri	STRING	NULLABLE
☐	generation	INTEGER	NULLABLE
☐	content_type	STRING	NULLABLE
☐	size	INTEGER	NULLABLE
☐	md5_hash	STRING	NULLABLE
☐	updated	TIMESTAMP	NULLABLE
☐ ▼	metadata	RECORD	REPEATED
☐	name	STRING	NULLABLE
☐	value	STRING	NULLABLE
☐ ▼	ref	RECORD	NULLABLE
☐	uri	STRING	NULLABLE
☐	version	STRING	NULLABLE
☐	authorizer	STRING	NULLABLE
☐	details	JSON	NULLABLE

▦ review_images | BigLake | 🔍 Query | Open in ▾ | ⊕ Share | 🗑 Delete

Schema | **Details** | Insights | Lineage | Data profile | Data Quality

Table ID	gcpdataengineering-467713.CustomerReviewGenAI.review_images
Created	6 Aug 2025, 22:08:53 UTC+5:30
Last modified	6 Aug 2025, 22:08:53 UTC+5:30
Table expiry	NEVER
Data location	US
Case insensitive	false
Description	
Labels	
Primary key(s)	
Tags	

External data configuration

Source URI(s)	gs://coffeeonwheels/images/*
Auto-detect schema	true
Source format	—
Connection ID	gcpdataengineering-467713.us.gemini_conn

```
-- BigQuery ML statement used to create or update a remote machine learning model i
```

```
CREATE OR REPLACE MODEL `CustomerReviewGenAI.gemini_2_0_flash`
REMOTE WITH CONNECTION `us.gemini_conn`
OPTIONS (endpoint = 'gemini-2.0-flash')
```

Cloud AI API is not enabled: Vertex AI API has not been used in project gcpdataengineering-467713 before or it is disabled. Enable it by visiting https://console.developers.google.com/apis/api/aiplatform.googleapis.com/overview?project=gcpdataengineering-467713 then retry. If you enabled this API recently, wait a few minutes for the action to propagate to our systems and retry..

Enable Vertex AI API form api& Services

bqcx-897867797155-2nup@gcp-sa-bigquery-condel.iam.gserviceaccount.com does not have the permission to access or use the endpoint. Please grant the Vertex AI user role to the service account following https://cloud.google.com/bigquery/docs/generate-text-tutorial#grant-permissions. If issue persists, contact bqml-feedback@google.com for help.

Grant Vertex AI user role to the service account created
Also Grant bucket permission for storage object admin to the external connection service account

⚙ gemini_2_0_flash   ↻ Refresh

Details   Inference   Registry

Model type	Model ID	Date created	Model location
Unknown	gcpdataengineering-467713.CustomerReviewGenAI.gemini_2_0_flash	6 Aug 2025, 22:33:33 UTC+5:30	US

Model details ✏ Edit

Date modified	6 Aug 2025, 22:33:33 UTC+5:30
Model expiry	Never
Description	-
Remote connection	gcpdataengineering-467713.us.gemini_conn
Remote endpoint	projects/gcpdataengineering-467713/locations/us-central1/publishers/google/models/gemini-2.0-flash

Training options

Training options are the parameters that were added in the script to create this model.

Actual iterations	1

Prompt for Keyword Extraction

This query takes customer reviews from the customer_reviews table, constructs prompts for the gemini_2_0_flash model to identify keywords within each review. The results are then stored in a new table customer_reviews_keywords.

```
CREATE OR REPLACE TABLE
`CustomerReviewGenAI.customer_reviews_keywords` AS (
SELECT ml_generate_text_llm_result, social_media_source, review_text, customer_id,
location_id, review_datetime
FROM
ML.GENERATE_TEXT(
MODEL `CustomerReviewGenAI.gemini_2_0_flash`,
(
  SELECT social_media_source, customer_id, location_id, review_text, review_datetime,
CONCAT(
    'For each review, provide keywords from the review. Answer in JSON format with one
key: keywords. Keywords should be a list.',
    review_text) AS prompt
  FROM `CustomerReviewGenAI.customer_reviews`
),
STRUCT(
  0.2 AS temperature, TRUE AS flatten_json_output)));
```

Filter Something went wrong.		
Field name	Type	Mode
☐ ml_generate_text_llm_result	STRING	NULLABLE
☐ social_media_source	STRING	NULLABLE
☐ review_text	STRING	NULLABLE
☐ customer_id	INTEGER	NULLABLE
☐ location_id	INTEGER	NULLABLE
☐ review_datetime	TIMESTAMP	NULLABLE

customer_revi...   🔍 Query   Open in ▾   +👤 Share   📋 Copy   ⊞ Snapshot   🗑 Delete   ⬆ Export   ↻ Refresh

Schema | Details | **Preview** | Table explorer `Preview` | Insights | Lineage | Data profile | Data Quality

Row	ml_generate_text_llm_result	social_media_source	review_text	customer_id	location_id	review_datetime

1	```json
	{
	"keywords": ["great experience", "short wait time", "friendly staff", "fast", "brewed
		Facebook	I had a great experience at this location! The wait time was short, and the staff were friendly and fast. I ordered a brewed coffee and it was	6620	8399	2024-09-06 15:07:10 UTC

2	```json
	{
	"keywords": ["friendly", "service", "fast", "coffee", "excellent", "recommend"]
		Facebook	The staff was very friendly and the service was fast. The coffee was excellent and I would definitely recommend this place to others.	4457	8704	2024-03-08 05:29:10 UTC

3	```json
	{
	"keywords": ["good service", "convenient", "cramped", "average seating", "decent
		Facebook	Good service. This location is convenient but cramped. The seating is pretty average. Still, they have a decent selection of drinks. The staff seemed	1612	8088	2024-03-27 04:15:10 UTC

4	```json	Facebook	The service at this location	7077	6004	2024-04-10 00:33:10 UTC

-- Analyze the customer reviews for positive and negative sentiment
--This query takes customer reviews from the customer_reviews table, constructs prompts for the gemini_2_0_flash model to classify the sentiment of each review. The results are then stored in a new table customer_reviews_analysis, so that you may use it later for further analysis.

```sql
CREATE OR REPLACE TABLE
`CustomerReviewGenAI.customer_reviews_analysis` AS (
SELECT ml_generate_text_llm_result, social_media_source, review_text, customer_id,
location_id, review_datetime
FROM
ML.GENERATE_TEXT(
MODEL `CustomerReviewGenAI.gemini_2_0_flash`,
(
  SELECT social_media_source, customer_id, location_id, review_text, review_datetime,
CONCAT(
    'Classify the sentiment of the following text as positive or negative.',
    review_text, "In your response don't include the sentiment explanation. Remove all
extraneous information from your response, it should be a boolean response either
positive or negative.") AS prompt
  FROM `CustomerReviewGenAI.customer_reviews`
),
STRUCT(
  0.2 AS temperature, TRUE AS flatten_json_output)));
```

Row	ml_generate_text_llm_result	social_media_source	review_text	customer_id	location_id	review_datetime
1	Negative	Google	The service could be improved. I think it's disorganized. The staff was very friendly and attentive. But we ordered two items and	995	8206	2024-01-16 20:00:10 UTC
2	Negative	Twitter	The coffee was too strong for my taste and it gave me a headache.	85	4034	2024-07-25 03:20:10 UTC
3	Negative	Twitter	Noisy Location'	877	3347	2024-05-01 14:39:10 UTC
4	negative	Google	I had a very disappointing experience at this coffee truck. The service was terrible - the staff were rude and inattentive, and we had to wait	3998	8844	2024-02-12 09:03:10 UTC
5	negative	Twitter	The baristas were incredibly rude to me for no reason. One	1224	9745	2024-09-09 20:26:10 UTC

Filter   Something went wrong.

	Field name	Type	Mode
☐	ml_generate_text_llm_result	STRING	NULLABLE
☐	social_media_source	STRING	NULLABLE
☐	review_text	STRING	NULLABLE
☐	customer_id	INTEGER	NULLABLE
☐	location_id	INTEGER	NULLABLE
☐	review_datetime	TIMESTAMP	NULLABLE

Table info

Table ID	gcpdataengineering-467713.CustomerReviewGenAI.customer_reviews_analysis
Created	6 Aug 2025, 23:46:24 UTC+5:30
Last modified	6 Aug 2025, 23:46:24 UTC+5:30
Table expiry	NEVER
Data location	US
Default collation	
Default rounding mode	ROUNDING_MODE_UNSPECIFIED
Case insensitive	false
Description	
Labels	
Primary key(s)	
Tags	

--Create a view to sanitize the records
--The query creates the view, cleaned_data_view and includes the sentiment results

```sql
CREATE OR REPLACE VIEW CustomerReviewGenAI.cleaned_data_view AS
SELECT
REPLACE(REPLACE(REPLACE(LOWER(ml_generate_text_llm_result), '.', ''), ' ', ''), '\n',
'') AS sentiment,
REGEXP_REPLACE(REGEXP_REPLACE(REGEXP_REPLACE(social_media_source,
r'Google(\+|\sReviews|\sLocal|\sMy\sBusiness|\sreviews|\sMaps)?',
     'Google'), 'YELP', 'Yelp'), r'SocialMedia1?', 'Social Media') AS
social_media_source,
review_text,
customer_id,
location_id,
review_datetime
FROM
CustomerReviewGenAI.customer_reviews_analysis;
```

Filter   Something went wrong.

	Field name	Type	Mode
☐	sentiment	STRING	NULLABLE
☐	social_media_source	STRING	NULLABLE
☐	review_text	STRING	NULLABLE
☐	customer_id	INTEGER	NULLABLE
☐	location_id	INTEGER	NULLABLE
☐	review_datetime	TIMESTAMP	NULLABLE

View info

View ID	gcpdataengineering-467713.CustomerReviewGenAI.cleaned_data_view
Created	6 Aug 2025, 23:52:49 UTC+5:30
Last modified	6 Aug 2025, 23:52:49 UTC+5:30
View expiry	NEVER
Use Legacy SQL	false
Description	
Labels	
Primary key(s)	
Tags	

ow	sentiment ▾		social_media_source ▾		review_text ▾		customer_id ▾		location_id ▾
1	negative		Google		The service could be improved. I think it's disorganized. The staff was very friendly and attentive. But we ordered two items and		995		8206
2	negative		Twitter		The coffee was too strong for my taste and it gave me a headache.		85		4034
3	negative		Twitter		Noisy Location'		877		3347
4	negative		Google		I had a very disappointing experience at this coffee truck. The service was terrible - the		3998		8844

--Create a report of positive and negative review counts

```sql
SELECT sentiment, COUNT(*) AS count
FROM `CustomerReviewGenAI.cleaned_data_view`
WHERE sentiment IN ('positive', 'negative')
GROUP BY sentiment;
```

Job information	Results	Chart	JSON	Execution details	Execution graph

Row		sentiment ▾		count ▾	
1		negative		9	
2		positive		41	

-- Create a count of positive and negative reviews by social media source

```sql
SELECT sentiment, social_media_source, COUNT(*) AS count
FROM `CustomerReviewGenAI.cleaned_data_view`
WHERE sentiment IN ('positive') OR sentiment IN ('negative')
GROUP BY sentiment, social_media_source
ORDER BY sentiment, count;
```

Job information	Results	Chart	JSON	Execution details	Execution graph

Row		sentiment ▾		social_media_source ▾		count ▾	
1		negative		Yelp		1	
2		negative		Google		2	
3		negative		Twitter		6	
4		positive		Tiktok		1	

Generating Market Suggestions

-- Respond to customer reviews

-- Note: Refer to zero-shot vs. few-shot prompts within the Google AI for Developers documentation for more information.

```sql
-- This table will contain the original review data along with the generated
marketing suggestions, allowing you to easily analyze and act upon them.
```

```sql
CREATE OR REPLACE TABLE
`CustomerReviewGenAI.customer_reviews_marketing` AS (
SELECT ml_generate_text_llm_result, social_media_source, review_text, customer_id,
location_id, review_datetime
FROM
ML.GENERATE_TEXT(
MODEL `CustomerReviewGenAI.gemini_2_0_flash`,
(
  SELECT social_media_source, customer_id, location_id, review_text, review_datetime,
CONCAT(
    'You are a marketing representative. How could we incentivise this customer with
this positive review? Provide a single response, and should be simple and concise, do
not include emojis. Answer in JSON format with one key: marketing. Marketing should be a
string.', review_text) AS prompt
  FROM `CustomerReviewGenAI.customer_reviews`
),
STRUCT(
  0.2 AS temperature, TRUE AS flatten_json_output)));
```

	Job information	**Results**	Chart	JSON	Execution details	Execution graph		
Row	ml_generate_text_llm_result ▼	social_media_source ▼	review_text ▼	customer_id ▼	location_id ▼	review_datetime ▼

1	```json
	{
	  "marketing": "Thank you for the
	great review! Enjoy a free
	coffee on your next visit as a
		Facebook	Great coffee	6093	3498	2024-11-10 12:57:10 UTC

2	```json
	{
	  "marketing": "Thank you for
	your review! Enjoy a free
	pastry with your next coffee
		Facebook	Good location and delicious cof...	5006	3174	2024-02-05 18:02:10 UTC

3	```json {	Facebook	Great coffee and friendly staff, ...	260	9464	2024-05-15 23:07:10 UTC

For Specific customer -> Apply where clause

```sql
CREATE OR REPLACE TABLE
`gemini_demo.customer_reviews_marketing` AS (
SELECT ml_generate_text_llm_result, social_media_source, review_text, customer_id,
location_id, review_datetime
FROM
ML.GENERATE_TEXT(
MODEL `gemini_demo.gemini_2_0_flash`,
(
    SELECT social_media_source, customer_id, location_id, review_text, review_datetime,
CONCAT(
      'You are a marketing representative. How could we incentivise this customer with
this positive review? Provide a single response, and should be simple and concise, do
not include emojis. Answer in JSON format with one key: marketing. Marketing should be a
string.', review_text) AS prompt
    FROM `gemini_demo.customer_reviews`
    WHERE customer_id = 5576
),
STRUCT(
```

```
    0.2 AS temperature, TRUE AS flatten_json_output)));

SELECT * FROM `gemini_demo.customer_reviews_marketing`
```

For Better Readability :-
Use:-

```
CREATE OR REPLACE TABLE
`CustomerReviewGenAI.customer_reviews_marketing_formatted` AS (
SELECT
  review_text,
  JSON_QUERY(RTRIM(LTRIM(results.ml_generate_text_llm_result, " ```json"), "```"),
"$.marketing") AS marketing,
  social_media_source, customer_id, location_id, review_datetime
FROM
  `CustomerReviewGenAI.customer_reviews_marketing` results )

select * from `CustomerReviewGenAI.customer_reviews_marketing_formatted` ;
```

| Job information | Results | Chart | JSON | Execution details | Execution graph |

Row	review_text ▼	marketing ▼	social_media_source ▼	customer_id ▼	location_id ▼	review_datetime ▼
1	The service at this location was great. The staff was friendly and helpful, and they made sure my coffee was made just the way I liked it.	"Offer a free drink or small discount on their next purchase as a thank you for their positive feedback and continued patronage."	Facebook	3079	4433	2024-06-26 10:23:10 U'
2	The service was great and the coffee was even better. I especially liked the variety of milk alternatives.	"Thank you for your kind words! We'd love to offer you a free coffee on your next visit to show our appreciation."	Facebook	4478	4364	2024-03-10 23:41:10 U'
3	I enjoyed my fresh brewed coffee. The barista was very friendly. I will make this a	"Thank you for your kind words! We'd love to offer you a free coffee on your next visit."	Facebook	1339	5634	2024-06-07 07:48:10 U'

Github link for the sources : -