Multimodal embeddings API

The Multimodal embeddings API generates vectors based on the input you provide, which can include a combination of image, text, and video data. The embedding vectors can then be used for subsequent tasks like image classification or video content moderation.

For additional conceptual information, see Multimodal embeddings (/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings).

Supported Models:

Model	Code
Embeddings for Multimodal	`multimodalembedding@001`

Example syntax

Syntax to send a multimodal embeddings API request.

curl Python (#python)
(#curl)

```
curl -X POST \
-H "Authorization: Bearer $(gcloud auth print-access-token)" \
-H "Content-Type: application/json" \

https://${LOCATION}-aiplatform.googleapis.com/v1/projects/${PROJECT_ID}/
-d '{
"instances": [
  ...
],
}'
```

Parameter list

See examples (#examples) for implementation details.

Request Body

```
{
  "instances": [
    {
      "text": string,
      "image": {
        // Union field can be only one of the following:
        "bytesBase64Encoded": string,
        "gcsUri": string,
        // End of list of possible types for union field.
        "mimeType": string
      },
      "video": {
        // Union field can be only one of the following:
        "bytesBase64Encoded": string,
        "gcsUri": string,
        // End of list of possible types for union field.
        "videoSegmentConfig": {
          "startOffsetSec": integer,
          "endOffsetSec": integer,
          "intervalSec": integer
        }
      },
      "parameters": {
        "dimension": integer
      }
    }
  ]
}
```

Parameters

`image`	Optional: **`Image`** The image to generate embeddings for.
`text`	Optional: **`String`** The text to generate embeddings for.
`video`	Optional: **`Video`** The video segment to generate embeddings for.

`dimension`	Optional: `Int` The dimension of the embedding, included in the response. Only applies to text and image input. Accepted values: **128**, **256**, **512**, or **1408**.

Image

Parameters	
`bytesBase64Encoded`	Optional: `String` Image bytes encoded in a base64 string. Must be one of **`bytesBase64Encoded`** or **`gcsUri`**.
`gcsUri`	Optional. `String` The Cloud Storage location of the image to perform the embedding. One of **`bytesBase64Encoded`** or **`gcsUri`**.
`mimeType`	Optional. `String` The MIME type of the content of the image. Supported values: **`image/jpeg`** and **`image/png`**.

Video

Parameters	
`bytesBase64Encoded`	Optional: `String` Video bytes encoded in base64 string. One of **`bytesBase64Encoded`** or **`gcsUri`**.
`gcsUri`	Optional: `String` The Cloud Storage location of the video on which to perform the embedding. One of **`bytesBase64Encoded`** or **`gcsUri`**.
`videoSegmentConfig`	Optional: `VideoSegmentConfig` The video segment config.

VideoSegmentConfig

Parameters	
`startOffsetSec`	Optional: **`Int`** The start offset of the video segment in seconds. If not specified, it's calculated with **`max(0, endOffsetSec - 120)`**.
`endOffsetSec`	Optional: **`Int`** The end offset of the video segment in seconds. If not specified, it's calculated with **`min(video length, startOffSec + 120)`**. If both **`startOffSec`** and **`endOffSec`** are specified, **`endOffsetSec`** is adjusted to **`min(startOffsetSec+120, endOffsetSec)`**.
`intervalSec`	Optional. **`Int`** The interval of the video the embedding will be generated. The minimum value for **`interval_sec`** is 4. If the interval is less than **4**, an **`InvalidArgumentError`** is returned. There are no limitations on the maximum value of the interval. However, if the interval is larger than **`min(video length, 120s)`**, it impacts the quality of the generated embeddings. Default value: **16**.

Response body

```
{
  "predictions": [
    {
      "textEmbedding": [
        float,
        // array of 128, 256, 512, or 1408 float values
        float
      ],
      "imageEmbedding": [
        float,
        // array of 128, 256, 512, or 1408 float values
        float
      ],
      "videoEmbeddings": [
        {
          "startOffsetSec": integer,
          "endOffsetSec": integer,
          "embedding": [
            float,
            // array of 1408 float values
```

```
            float
          ]
        }
      ]
    }
  ],
  "deployedModelId": string
}
```

Response element	Description
`imageEmbedding`	128, 256, 512, or 1408 dimension list of floats.
`textEmbedding`	128, 256, 512, or 1408 dimension list of floats.
`videoEmbeddings`	1408 dimension list of floats with the start and end time (in seconds) of the video segment that the embeddings are generated for.

Examples

Basic use case

Generate embeddings from image

Use the following sample to generate embeddings for an image.

REST Vertex AI SDK for Python... Node.js (#node.js) Java (#java) Go (#go)
(#rest)

Before using any of the request data, make the following replacements:

- ***LOCATION***: Your project's region. For example, `us-central1`, `europe-west2`, or `asia-northeast3`. For a list of available regions, see Generative AI on Vertex AI locations (/vertex-ai/generative-ai/docs/learn/locations-genai).
- ***PROJECT_ID***: Your Google Cloud project ID (/resource-manager/docs/creating-managing-projects#identifiers).
- ***TEXT***: The target text to get embeddings for. For example, `a cat`.
- ***B64_ENCODED_IMG***: The target image to get embeddings for. The image must be specified as a base64-encoded (/vertex-ai/generative-ai/docs/image/base64-encode) byte string.

HTTP method and URL:

```
POST https://LOCATION-aiplatform.googleapis.com/v1/projects/PROJECT_I
```

Request JSON body:

```
{
  "instances": [
    {
      "text": "TEXT",
      "image": {
        "bytesBase64Encoded": "B64_ENCODED_IMG"
      }
    }
  ]
}
```

To send your request, choose one of these options:

curl (#curl) PowerShell (#powershell)

Note: The following command assumes that you have logged in to the **gcloud** CLI with your user account by running **gcloud init** (/sdk/gcloud/reference/init) or **gcloud auth login** (/sdk/gcloud/reference/auth/login) . You can check the currently active account by running **gcloud auth list** (/sdk/gcloud/reference/auth/list).

Save the request body in a file named `request.json`, and execute the following command:

```
$cred = gcloud auth print-access-token
$headers = @{ "Authorization" = "Bearer $cred" }

Invoke-WebRequest `
    -Method POST `
    -Headers $headers `
    -ContentType: "application/json; charset=utf-8" `
```

```
    -InFile request.json `
    -Uri "https://LOCATION-aiplatform.googleapis.com/v1/projects
```

The embedding the model returns is a 1408 float vector. The following sample response is shortened for space.

```
{
  "predictions": [
    {
      "textEmbedding": [
        0.010477379,
        -0.00399621,
        0.00576670747,
        [...]
        -0.00823613815,
        -0.0169572588,
        -0.00472954148
      ],
      "imageEmbedding": [
        0.00262696808,
        -0.00198890246,
        0.0152047109,
        -0.0103145819,
        [...]
        0.0324628279,
        0.0284924973,
        0.011650892,
        -0.00452344026
      ]
    }
  ],
  "deployedModelId": "DEPLOYED_MODEL_ID"
}
```

Generate embeddings from video

Use the following sample to generating embeddings for video content.

REST Vertex AI SDK for Python... Go (#go)
(#rest)

The following example uses a video located in Cloud Storage. You can also use the **`video.bytesBase64Encoded`** field to provide a base64-encoded (/vertex-ai/generative-ai/docs/image/base64-encode) string representation of the video.

Before using any of the request data, make the following replacements:

- ***LOCATION***: Your project's region. For example, **`us-central1`**, **`europe-west2`**, or **`asia-northeast3`**. For a list of available regions, see Generative AI on Vertex AI locations (/vertex-ai/generative-ai/docs/learn/locations-genai).
- ***PROJECT_ID***: Your Google Cloud project ID (/resource-manager/docs/creating-managing-projects#identifiers).
- ***VIDEO_URI***: The Cloud Storage URI of the target video to get embeddings for. For example, **`gs://my-bucket/embeddings/supermarket-video.mp4`**.

  You can also provide the video as a base64-encoded byte string:

```
[...]
"video": {
  "bytesBase64Encoded": "B64_ENCODED_VIDEO"
}
[...]
```

- **`videoSegmentConfig`** (***START_SECOND***, ***END_SECOND***, ***INTERVAL_SECONDS***). Optional. The specific video segments (in seconds) the embeddings are generated for.

The value you set for **`videoSegmentConfig.intervalSec`** affects the pricing tier you are charged at. For more information, see the video embedding modes (/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings#video-modes) section and pricing (/vertex-ai/generative-ai/pricing) page.

For example:

```
[...]
"videoSegmentConfig": {
  "startOffsetSec": 10,
  "endOffsetSec": 60,
  "intervalSec": 10
```

```
    }
    [...]
```

Using this config specifies video data from 10 seconds to 60 seconds and generates embeddings for the following 10 second video intervals: [10, 20), [20, 30), [30, 40), [40, 50), [50, 60). This video interval (`"intervalSec": 10`) falls in the Standard video embedding mode (/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings#video-modes), and the user is charged at the Standard mode pricing rate (/vertex-ai/generative-ai/pricing).

If you omit `videoSegmentConfig`, the service uses the following default values: `"videoSegmentConfig": { "startOffsetSec": 0, "endOffsetSec": 120, "intervalSec": 16 }`. This video interval (`"intervalSec": 16`) falls in the Essential video embedding mode (/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings#video-modes), and the user is charged at the Essential mode pricing rate (/vertex-ai/generative-ai/pricing).

HTTP method and URL:

```
POST https://LOCATION-aiplatform.googleapis.com/v1/projects/PROJECT_I
```

Request JSON body:

```
{
  "instances": [
    {
      "video": {
        "gcsUri": "VIDEO_URI",
        "videoSegmentConfig": {
          "startOffsetSec": START_SECOND,
          "endOffsetSec": END_SECOND,
          "intervalSec": INTERVAL_SECONDS
        }
      }
    }
```

```
  ]
}
```

To send your request, choose one of these options:

curl (#curl)PowerShell (#powershell)

Note: The following command assumes that you have logged in to the **gcloud** CLI with your user account by running **gcloud init** (/sdk/gcloud/reference/init) or **gcloud auth login** (/sdk/gcloud/reference/auth/login) . You can check the currently active account by running **gcloud auth list** (/sdk/gcloud/reference/auth/list).

Save the request body in a file named **request.json**, and execute the following command:

```
$cred = gcloud auth print-access-token
$headers = @{ "Authorization" = "Bearer $cred" }

Invoke-WebRequest `
    -Method POST `
    -Headers $headers `
    -ContentType: "application/json; charset=utf-8" `
    -InFile request.json `
    -Uri "https://LOCATION -aiplatform.googleapis.com/v1/projects
```

The embedding the model returns is a 1408 float vector. The following sample responses are shortened for space.

Response (7 second video, no `videoSegmentConfig` specified):

```
{
  "predictions": [
    {
      "videoEmbeddings": [
        {
          "endOffsetSec": 7,
          "embedding": [
            -0.0045467657,
```

```
            0.0258095954,
            0.0146885719,
            0.00945400633,
            [...]
            -0.0023291884,
            -0.00493789,
            0.00975185353,
            0.0168156829
          ],
          "startOffsetSec": 0
        }
      ]
    }
  ],
  "deployedModelId": "DEPLOYED_MODEL_ID"
}
```

Response (59 second video, with the following video segment config: `"videoSegmentConfig": { "startOffsetSec": 0, "endOffsetSec": 60, "intervalSec": 10 }`):

```
{
  "predictions": [
    {
      "videoEmbeddings": [
        {
          "endOffsetSec": 10,
          "startOffsetSec": 0,
          "embedding": [
            -0.00683252793,
            0.0390476175,
            [...]
            0.00657121744,
            0.013023301
          ]
        },
        {
          "startOffsetSec": 10,
          "endOffsetSec": 20,
          "embedding": [
            -0.0104404651,
            0.0357737206,
            [...]
            0.00509833824,
            0.0131902946
```

```
        ]
      },
      {
        "startOffsetSec": 20,
        "embedding": [
          -0.0113538112,
          0.0305239167,
          [...]
          -0.00195809244,
          0.00941874553
        ],
        "endOffsetSec": 30
      },
      {
        "embedding": [
          -0.00299320649,
          0.0322436653,
          [...]
          -0.00993082579,
          0.00968887936
        ],
        "startOffsetSec": 30,
        "endOffsetSec": 40
      },
      {
        "endOffsetSec": 50,
        "startOffsetSec": 40,
        "embedding": [
          -0.00591270532,
          0.0368893594,
          [...]
          -0.00219071587,
          0.0042470959
        ]
      },
      {
        "embedding": [
          -0.00458270218,
          0.0368121453,
          [...]
          -0.00317760976,
          0.00595594104
        ],
        "endOffsetSec": 59,
        "startOffsetSec": 50
      }
    ]
  }
```

```
  ],
  "deployedModelId": "DEPLOYED_MODEL_ID"
}
```

Advanced use case

Use the following sample to get embeddings for video, text, and image content.

For video embedding, you can specify the video segment and embedding density.

REST Vertex AI SDK for Python... Go (#go)
(#rest)

The following example uses image, text, and video data. You can use any combination of these data types in your request body.

This sample uses a video located in Cloud Storage. You can also use the `video.bytesBase64Encoded` field to provide a base64-encoded (/vertex-ai/generative-ai/docs/image/base64-encode) string representation of the video.

Before using any of the request data, make the following replacements:

- ***LOCATION***: Your project's region. For example, `us-central1`, `europe-west2`, or `asia-northeast3`. For a list of available regions, see Generative AI on Vertex AI locations (/vertex-ai/generative-ai/docs/learn/locations-genai).
- ***PROJECT_ID***: Your Google Cloud project ID (/resource-manager/docs/creating-managing-projects#identifiers).
- ***TEXT***: The target text to get embeddings for. For example, `a cat`.
- ***IMAGE_URI***: The Cloud Storage URI of the target image to get embeddings for. For example, `gs://my-bucket/embeddings/supermarket-img.png`.

  You can also provide the image as a base64-encoded byte string:

```
[...]
"image": {
  "bytesBase64Encoded": "B64_ENCODED_IMAGE"
}
[...]
```

- ***VIDEO_URI***: The Cloud Storage URI of the target video to get embeddings for. For example, `gs://my-bucket/embeddings/supermarket-video.mp4`.

  You can also provide the video as a base64-encoded byte string:

```
[...]
"video": {
  "bytesBase64Encoded": "B64_ENCODED_VIDEO"
}
[...]
```

- `videoSegmentConfig` (***START_SECOND***, ***END_SECOND***, ***INTERVAL_SECONDS***). Optional. The specific video segments (in seconds) the embeddings are generated for.

! The value you set for **`videoSegmentConfig.intervalSec`** affects the pricing tier you are charged at. For more information, see the video embedding modes (/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings#video-modes) section and pricing (/vertex-ai/generative-ai/pricing) page.

For example:

```
[...]
"videoSegmentConfig": {
  "startOffsetSec": 10,
  "endOffsetSec": 60,
  "intervalSec": 10
}
[...]
```

Using this config specifies video data from 10 seconds to 60 seconds and generates embeddings for the following 10 second video intervals: [10, 20), [20, 30), [30, 40), [40, 50), [50, 60). This video interval (`"intervalSec": 10`) falls in the Standard video embedding mode (/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings#video-modes), and the user is charged at the Standard mode pricing rate (/vertex-ai/generative-ai/pricing).

If you omit `videoSegmentConfig`, the service uses the following default values: `"videoSegmentConfig": { "startOffsetSec": 0, "endOffsetSec": 120, "intervalSec": 16 }`. This video interval (`"intervalSec": 16`) falls in the Essential video embedding mode (/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings#video-modes), and the user is charged at the Essential mode pricing rate (/vertex-ai/generative-ai/pricing).

HTTP method and URL:

```
POST https://LOCATION-aiplatform.googleapis.com/v1/projects/PROJECT_I
```

Request JSON body:

```
{
  "instances": [
    {
      "text": "TEXT",
      "image": {
        "gcsUri": "IMAGE_URI"
      },
      "video": {
        "gcsUri": "VIDEO_URI",
        "videoSegmentConfig": {
          "startOffsetSec": START_SECOND,
          "endOffsetSec": END_SECOND,
          "intervalSec": INTERVAL_SECONDS
        }
      }
    }
  ]
}
```

To send your request, choose one of these options:

curl (#curl) PowerShell (#powershell)

Note: The following command assumes that you have logged in to the **gcloud** CLI with your user account by running **gcloud init** (/sdk/gcloud/reference/init) or **gcloud auth**

login (/sdk/gcloud/reference/auth/login) . You can check the currently active account by running **gcloud auth list** (/sdk/gcloud/reference/auth/list).

Save the request body in a file named **request.json**, and execute the following command:

```
$cred = gcloud auth print-access-token
$headers = @{ "Authorization" = "Bearer $cred" }

Invoke-WebRequest `
    -Method POST `
    -Headers $headers `
    -ContentType: "application/json; charset=utf-8" `
    -InFile request.json `
    -Uri "https://LOCATION-aiplatform.googleapis.com/v1/project
```

The embedding the model returns is a 1408 float vector. The following sample response is shortened for space.

```
{
  "predictions": [
    {
      "textEmbedding": [
        0.0105433334,
        -0.00302835181,
        0.00656806398,
        0.00603460241,
        [...]
        0.00445805816,
        0.0139605571,
        -0.00170318608,
        -0.00490092579
      ],
      "videoEmbeddings": [
        {
          "startOffsetSec": 0,
          "endOffsetSec": 7,
          "embedding": [
            -0.00673126569,
            0.0248149596,
            0.0128901172,
```

```
            0.0107588246,
            [...]
            -0.00180952181,
            -0.0054573305,
            0.0117037306,
            0.0169312079
          ]
        }
      ],
      "imageEmbedding": [
        -0.00728622358,
        0.031021487,
        -0.00206603738,
        0.0273937676,
        [...]
        -0.00204976718,
        0.00321615417,
        0.0121978866,
        0.0193375275
      ]
    }
  ],
  "deployedModelId": "DEPLOYED_MODEL_ID"
}
```

What's next

For detailed documentation, see the following:

- Get multimodal embeddings
  (/vertex-ai/generative-ai/docs/embeddings/get-multimodal-embeddings)

Last updated 2025-03-21 UTC.