Gemini Decisions API explained: scoring choices instead of generating text

21 November 2026 · 7 min read · blogs.iictp.org/gemini-decisions-api-explained

Short answer: the Gemini Decisions API is a Gemini API endpoint, announced by Google on 11 November 2026, for closed-set decisions such as yes/no checks, multiple choice and rubric scores. It runs a new model, Gemini Decision 1 (gemini-decision-1), which scores the candidate options in a single forward pass and returns probabilities and confidence scores instead of generating answer text token by token. One multimodal input can be checked against up to 1,000 questions in one request.

TL;DR The short version

- The Gemini Decisions API, announced by Google on 11 November 2026, handles closed-set decisions such as yes/no checks, multiple choice and rubric scores.
- Its model, Gemini Decision 1, scores candidate options in one forward pass and returns probabilities instead of generating text.
- It only works when the right answer is among your options, and confidence should be a routing signal tuned on your own evaluation data.

Read online, listen to it, or change the text size: https://blogs.iictp.org/gemini-decisions-api-explained

What is the Gemini Decisions API?

Many production LLM calls are not really generation. Routing a query to a model tier, checking a policy rule, picking a tool or grading retrieved context all have a fixed set of possible answers. Teams usually handle these with a generative call and a parser, or with a custom classifier they have to train. The Decisions API offers a purpose-built primitive for this: you define the options, and the model returns how likely each one is.

How it works

Instead of decoding output tokens, Gemini Decision 1 scores your candidate options in one forward pass, which is why Google bills no output tokens. The input uses the same Content block format as the Interactions API and can mix text, images, audio, video and PDFs, up to 256K tokens according to Google. Image and video resolution can be tuned to trade detail against token cost.

Each request carries a shared input and a map of named questions. Google describes three question types:

- **Yes/no:** the probability that a condition holds, with optional criteria describing what counts as true and false. Google's sample code names this type noul.
- **choice:** picks from 2 to 255 options and returns the winner plus normalised probabilities for every option.

- **score:** grades against an ordered rubric starting at 0 and returns a continuous weighted score and per-level probabilities.

Every answer includes a confidence score from 0 to 1, so your code can send ambiguous cases to a fallback model or a human reviewer.

What is new compared with earlier approaches

The difference from asking a chat model for JSON is structural. A generative call produces one sampled answer, and any confidence you get is either absent or self-reported in text. Scoring options directly returns a probability for each one, and many independent questions share one pass over the input, which Google says keeps latency nearly flat from 1 to 100 questions per request. Compared with training your own classifier, you change behaviour by editing option descriptions rather than collecting labels and retraining.

Results reported by Google

- Median model compute of 50 to 100 ms and 150 to 250 ms end to end, for multimodal inputs and contexts up to 16K tokens, according to Google.
- Google says Gemini Decision 1 leads its benchmark suite spanning agent routing, reasoning, ranking and multimodal understanding, with the lowest calibration error. The post does not name the individual benchmarks or competitor scores in the text we read.
- Google says it is nearly 5 times cheaper across its shared benchmark suite than Gemini 3.5 Flash-Lite through the Interactions API.
- Price: $0.10 per million input tokens, $0 for output tokens, with a no-cost tier in Google AI Studio for prototyping. The shared input is billed once per request regardless of the number of questions.
- Internal uses Google describes: model routing in Gemini Notebook (up to 40% lower latency for everyday queries, per Google), rapid wireframes in Stitch, emoji selection in CC from Google Labs, and a safety layer for MCP tools in Google Search.

All of these are vendor claims from the launch post. Calibration in particular should be checked on your own data before you set thresholds.

Limitations and open questions

- **Closed sets only.** If the right answer is not among your options, the model will still distribute probability across the options you gave.
- **Calibration is data-dependent.** A model calibrated on Google’s suite may be over- or under-confident on your domain. Measure it with a labelled sample.
- **Option wording matters.** Criteria text is effectively the classifier definition, so changes to it need the same review and evaluation as code changes.
- **Preview surface.** The sample uses the v1beta API version. Expect changes before general availability.

How to use the Gemini Decisions API

1. Get an API key in Google AI Studio.
2. Install the google-genai SDK and create a client with the v1beta API version, as in Google’s sample.

3. Call client.decisions.create with model "gemini-decision-1", an input list of content blocks, and a questions map whose entries each have a type, instructions and, for choice and score, criteria.
4. Read the probability or score and the confidence for each question, and route low-confidence answers to a fallback.
5. Before relying on thresholds, build a small labelled set and compare predicted probabilities with actual outcomes.

Google lists Python, JavaScript and REST guides in the Decisions API documentation, and says integrations are rolling out for OpenRouter, LangChain, Vercel, LiteLLM, Pydantic AI, Pipecat, LiveKit and OpenCode.

What it means for engineers and technical teams

- Move routing, guardrail and grading calls off generative models where the answer set is fixed. That is where latency and output-token spend add up.
- Treat confidence as a routing signal, not a truth label. Set thresholds from your own evaluation data. See how to evaluate an LLM application.
- Version your question definitions alongside code, because they define behaviour.

Frequently asked questions

- **What model does the Gemini Decisions API use?** Gemini Decision 1, model ID gemini-decision-1.
- **How many questions can one request ask?** Up to 1,000 independent questions against one shared input, according to Google.
- **Does it generate text?** No. It returns probabilities, scores and confidence values for the options you define, and output tokens are not billed.
- **Which inputs does it accept?** Text, images, audio, video and PDFs, up to 256K tokens per Google.

Related reading

- How to evaluate an LLM application
- LLMOps: running AI in production
- Multimodal AI explained
- AI Practitioner certification

Sources were read on 21 November 2026. Preview APIs, model IDs, prices and access rules change often, so check the linked documentation before you build on them.

> Related credential: AI Practitioner. Explore the exam facts and syllabus.

Sources and further reading

1. Google (Choi, M., Yu, F.). Gemini Decisions API: Fast, calibrated multimodal decisions. 2026-11-11. blog.google/innovation-and-ai/technology/developers-tools/gemini-decisions-api

Published by IICTP, an independent technical credential body, on 21 November 2026. The current version of this post is at https://blogs.iictp.org/gemini-decisions-api-explained. More guides: blogs.iictp.org.