



PEGASUS 1.5 · TBM · 📄 📄

📄 📄 📄 📄 📄 📄

📄 📄 📄 📄

Pegasus 1.5 Time-Based Metadata API · 📄/📄 📄 📄 📄 📄

📄 📄 📄 📄 📄 📄 📄 TwelveLabs Pegasus 1.5 📄 📄 📄📄📄(TBM) API 📄 📄📄📄. 📄📄 📄📄 📄📄 📄📄 📄📄 📄📄
JSON 📄📄 📄📄 📄 📄📄, 📄📄📄📄 📄📄 📄📄 📄📄📄 UI📄 📄📄📄 📄 📄📄📄.

📄 📄: tbdemo-alpha.vercel.app · 📄📄 📄📄 📄📄-📄📄 📄📄 📄📄📄📄 📄📄 📄📄-📄📄 JSON📄 📄📄 📄📄📄📄📄 📄 📄📄📄. (📄📄 📄📄 📄📄 📄📄
📄📄 📄📄)

01 API 📄📄

📄📄	📄
Base URL	https://api.twelvelabs.io/v1.3
📄📄 📄📄 (📄📄📄)	POST /analyze/tasks
📄📄 📄📄	GET /analyze/tasks/{task_id} — status: ready 📄📄 polling
📄📄	x-api-key: <API_KEY>
📄📄	pegasus1.5
📄📄 📄📄	analysis_mode: "time_based_metadata"
📄📄 📄📄	docs.twelvelabs.io

📄📄 📄📄 task📄 `result.data` 📄 JSON 📄📄📄📄 📄📄 📄📄. 📄📄📄📄📄📄 📄 📄 📄📄📄📄 📄📄 📄📄 📄📄📄 📄📄.

02 □□(Request) □□

□□□ □□□ segment_definitions □ □□□□□. □ □□□ □□□□ □□ □□(description) □ □□ □□(fields) □□□□ □□□□, □□□□ □□□□ □□□ □ □□□□. □□□ □ □□ □□□ □□□ □□ □□ payload□□□□ (3□ □□□).

```
{
  "model_name": "pegasus1.5",
  "video": {
    "type": "asset_id",
    "asset_id": "6a170c7dc6d6bda7c4c9b08d"
  },
  "analysis_mode": "time_based_metadata",
  "response_format": {
    "type": "segment_definitions",
    "segment_definitions": [
      {
        "id": "L0_shot_boundaries",
        "description": "Detect all shot boundaries in this soccer broadcast.",
        "fields": [
          { "name": "shot_type", "type": "string",
            "enum": ["WIDE_PITCH", "PLAYER_CLOSE_UP", "CROWD", "BENCH", "OTHER"] },
          { "name": "description", "type": "string" },
          { "name": "main_actions", "type": "array", "items": {"type": "string"} },
          { "name": "emotional_elements", "type": "string" },
          { "name": "player_names", "type": "array", "items": {"type": "string"} }
        ]
      },
      {
        "id": "H16_soccer_events",
        "description": "Identify soccer gameplay events, live or replayed.",
        "fields": [
          { "name": "event_type", "type": "string",
            "enum": ["GOAL", "SHOT_ON_TARGET", "SHOT_OFF_TARGET", "HEADER",
              "SAVE", "FREE_KICK", "CORNER_KICK", "PENALTY_KICK",
              "FOUL", "YELLOW_CARD", "RED_CARD", "VAR_REVIEW", "CELEBRATION"] },
          { "name": "playback_status", "type": "string",
            "enum": ["LIVE_GAMEPLAY", "REPLAY"] },
          { "name": "match_start_time", "type": "string" },
          { "name": "match_end_time", "type": "string" },
          { "name": "description", "type": "string" },
          { "name": "player_names", "type": "array", "items": {"type": "string"} }
        ]
      },
      {
        "id": "H16_broadcast_structure",
        "description": "Classify non-live inserts: replays, halftime, interviews.",
        "fields": [
          { "name": "segment_type", "type": "string",
            "enum": ["REPLAY_ANALYSIS", "HALFTIME", "POST_MATCH",
              "PLAYER_SPOTLIGHT", "CROWD_REACTION", "INTERVIEW"] },
          { "name": "description", "type": "string" },
          { "name": "player_names", "type": "array", "items": {"type": "string"} },
          { "name": "emotional_state", "type": "string" }
        ]
      }
    ]
  }
}
```

□□ description □□(□□□□ □□ □□ □□) □ □□□ API Request □□□□ □□ □□□□ □□ □□□□.

03 响应(Response) 结构

响应结构包含 `segment_definition` 的 `id` 以及 `start_time / end_time` 元数据。

```
{
  "H16_soccer_events": [
    {
      "start_time": 53.0,
      "end_time": 84.0,
      "metadata": {
        "event_type": "GOAL",
        "playback_status": "LIVE_GAMEPLAY",
        "match_start_time": "00:53",
        "match_end_time": "00:58",
        "description": "Pereira scores for Fulham with a chip
          over the goalkeeper.",
        "player_names": ["Andreas Pereira", "Aaron Ramsdale"]
      }
    }
  ]
}
```

响应结构

名称 (ID)	包含的元数据
L0_shot_boundaries	shot_type, description, main_actions[], emotional_elements, player_names[]
H16_soccer_events	event_type, playback_status (LIVE/REPLAY), match_start_time, match_end_time, description, player_names[]
H16_broadcast_structure	segment_type, description, player_names[], emotional_state

响应结构包含 `Response JSON` 元数据。

04 响应/请求 (3部分)

响应/请求结构包含 `A` 和 `B` 元数据。

名称	包含的元数据	包含的元数据	包含的元数据
A - 请求	Pegasus 请求: 请求 JSON 元数据	请求元数据	请求元数据 (请求)
B - 请求	Pegasus + Bedrock Claude: JSON 元数据	Bedrock (请求元数据)	请求元数据
C - 请求	Jockey + Bedrock Claude: 请求元数据	请求元数据	请求元数据

B 请求 JSON 元数据 Claude 请求元数据 DB 请求元数据。

05 □□ □□□□□

가 □□ □□(□□ A) □□ □□□□□. □□□□□ □□□ □□ UI/□□ □□□ □□□ □□□□□□, AWS □□□(S3, signed URL □) □□ □ □□ □□□ □□□.

1 (□□□) □□□ □□□ S3□ □□□

2 (□□□) signed URL □□

3 Pegasus 1.5 TBM API □□ — `POST /analyze/tasks`

4 □□ JSON□ S3□ □□

5 (□□□) JSON □□ □ □□□□□/UI □□

□□ B□ 4□ □□□ "Bedrock Claude□ JSON □□" □□□ □□□□□.