YouTube-8M: A Large-Scale Video Classification Benchmark

S Abu-El-Haija (2016)

발표자: 김사라

2017.04.24

YouTube-8M Dataset

- A large-scale benchmark dataset for general multi-label video classification.
- Here, they treat the task of video classification as that of producing labels that are relevant to a video given its frame.
- YouTube-8M is not restricted to action classes alone.

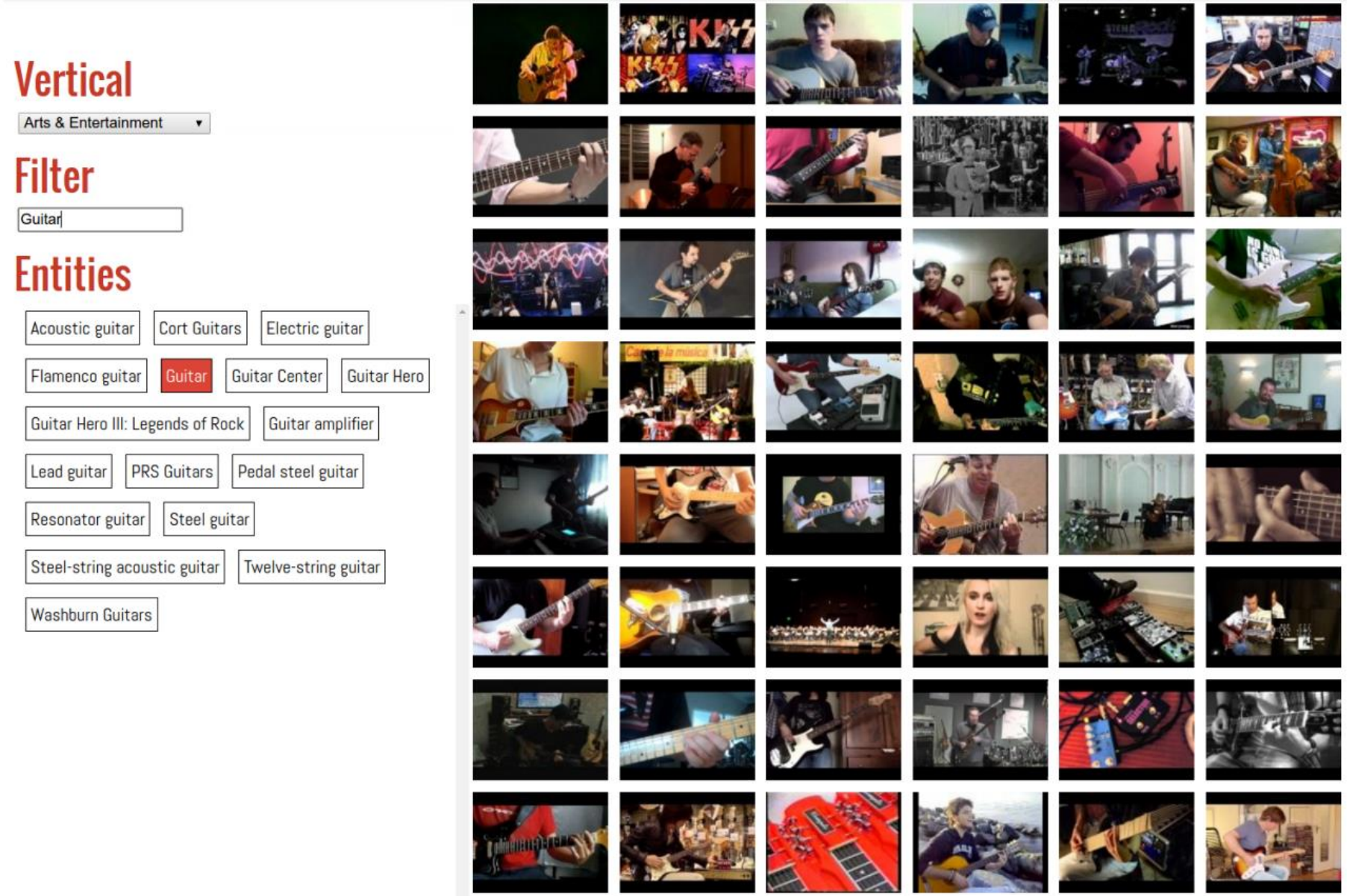
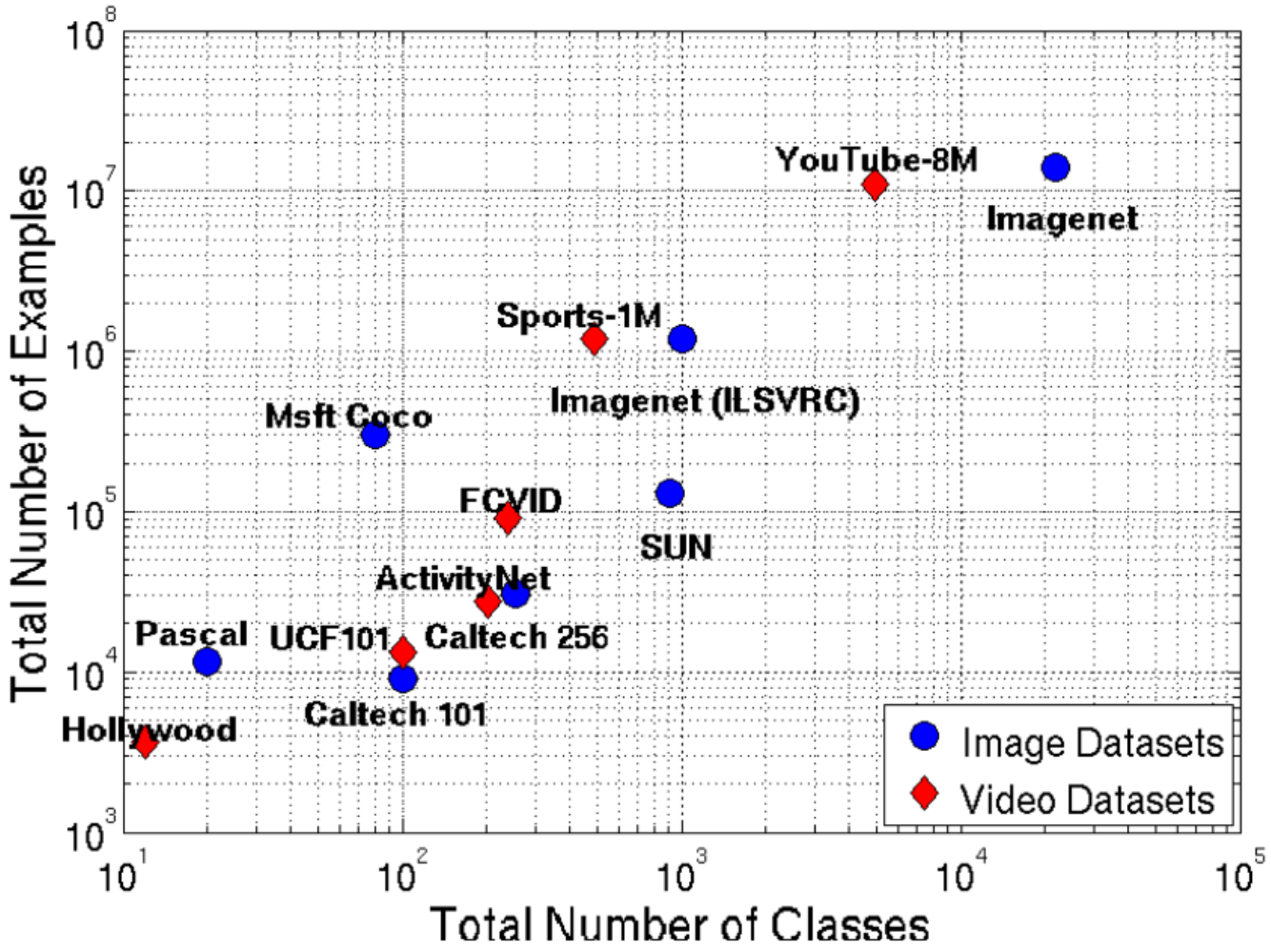


Figure1: This screenshot of a dataset explorer depicts a subset of videos in the dataset annotated with the entity "Guitar".

YouTube-8M Dataset

- YouTube-8M contains more than 8 million videos from 4,800 classes.



YouTube-8M Dataset

- We use the YouTube video annotation system to obtain topic annotations for a video, and to retrieve videos for a given topics.

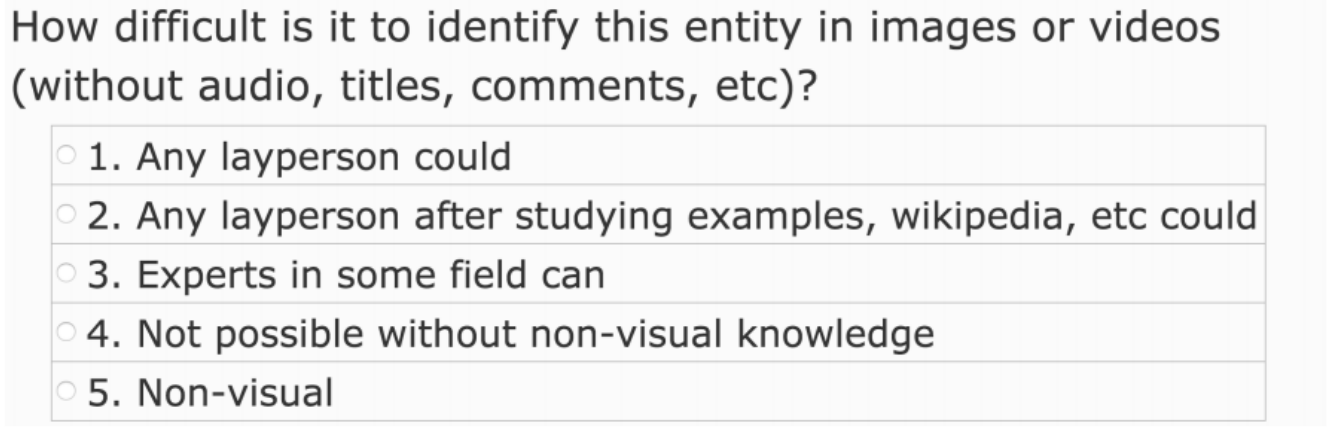
- The annotations are provided in the form of Knowledge Graph entities. We use Knowledge Graph entities to succinctly describe the main themes of a video, e.g., a video of biking on dirt roads and cliffs would have a central topic/theme of Mountain Biking, not Dirt, Road, Person, Sky, and so on.

- The aim of the dataset is not only to understand what is present in each frame of the video, but also identify the few key topics that best describe what the video is about.

YouTube-8M Dataset

1. Vocabulary Construction
   - every label in the dataset should be distinguishable using visual information alone,
   - each label should have sufficient number of videos for training models and for computing reliable metrics on the test set.

Entity Name	Entity URL	Entity Description
Thunderstorm	http://www.freebase.com/m/0jb2l	A thunderstorm, also known as an electrical storm, a lightning storm, or a thundershower, is a type of storm characterized by the presence of lightning and its acoustic effect on the Earth's atmosphere known as thunder. The meteorologically assigned cloud type associated with the thunderstorm is the cumulonimbus. Thunderstorms are usually accompanied by strong winds, heavy rain and sometimes snow, sleet, hail, or no precipitation at all...



(a) Screenshot of the question displayed to human raters.

YouTube-8M Dataset

2. Collecting Videos
   - have at least 1,000 views.
   - We exclude too short (< 120 secs) or too long (> 500 secs) videos.
   - Randomly sample 10 million videos among them.
   - Obtained all entities for the sampled 10 million videos using the YouTube video annotation system.
   - Filtered out entities with less than 200 videos, and videos with no remaining entities. This reduced the size of our data to 8,264,650 videos.

YouTube-8M Dataset

3. Features
   - We pre-process the videos and extract frame-level features using a state-of-art deep model: the publicly available Inception network trained on ImageNet.

   - We decode each video at 1 frame-per-second up to the first 360 seconds, feed the decoded frames into the Inception network, and fetch the ReLu activation of the last hidden layer, before the classification layer.

   - The feature vector is 2048-dimensional per second of video. Afterwards, we apply PCA (+ whitening) to reduce feature dimensions to 1024.

YouTube-8M Dataset

- ✓ While this removes motion information from the videos, recent work shows diminishing returns from motion features as the size and diversity of the video data increase.

- ✓ We hope to extend the dataset with audio and motion features in the futures.

YouTube-8M Dataset

- The YouTube-8M dataset contains 4,800 classes and a total of 8,264,650 videos.
- The average length of a video in the dataset is 229.6 seconds, which amounts to ~1.9 billion frames across the dataset.

Top-level Category	1st Entity	2nd Entity	3rd Entity	4th Entity	5th Entity	6th Entity	7th Entity
Arts & Entertainment	Concert	Animation	Music video	Dance	Guitar	Disc jockey	Trailer
Autos & Vehicles	Vehicle	Car	Motorcycle	Bicycle	Aircraft	Truck	Boat
Beauty & Fitness	Fashion	Hair	Cosmetics	Weight training	Hairstyle	Nail	Mascara
Books & Literature	Book	Harry Potter	The Bible	Writing	Magazine	Alice	E-book
Business & Industrial	Train	Model aircraft	Fish	Water	Tractor pulling	Advertising	Landing
Computers & Electronics	Personal computer	Video game console	iPhone	PlayStation 3	Tablet computer	Xbox 360	Microsoft Windows
Finance	Money	Bank	Foreign Exchange	Euro	United States Dollar	Credit card	Cash
Food & Drink	Food	Cooking	Recipe	Cake	Chocolate	Egg	Eating
Games	Video game	Minecraft	Action-adventure game	Strategy video game	Sports game	Call of Duty	Grand Theft Auto V
Health	Medicine	Raw food	Ear	Glasses	Injury	Dietary supplement	Dental braces
Hobbies & Leisure	Fishing	Outdoor recreation	Radio-controlled model	Wedding	Christmas	Hunting	Diving
Home & Garden	Gardening	Home improvement	House	Kitchen	Garden	Door	Swimming pool
Internet & Telecom	Mobile phone	Smartphone	Telephone	Website	Sony Xperia	Google Nexus	World Wide Web
Jobs & Education	School	University	High school	Teacher	Kindergarten	Campus	Classroom
Law & Government	Tank	Firefighter	President of the U.S.A.	Soldier	President	Police officer	Fighter aircraft
News	Weather	Snow	Rain	News broadcasting	Newspaper	Mattel	Hail
People & Society	Prayer	Family	Play-Doh	Human	Dragon	Angel	Tarot
Pets & Animals	Animal	Dog	Horse	Cat	Bird	Aquarium	Puppy
Real Estate	House	Apartment	Condominium	Dormitory	Mansion	Skyscraper	Loft
Reference	Vampire	Bus	River	City	Mermaid	Village	Samurai
Science	Nature	Robot	Eye	Ice	Biology	Skin	Light
Shopping	Toy	LEGO	Sledding	Doll	Shoe	My Little Pony	Nike; Inc.
Sports	Motorsport	Football	Winter sport	Cycling	Basketball	Gymnastics	Wrestling
Travel	Amusement park	Hotel	Airport	Beach	Roller coaster	Lake	Resort
Full vocabulary	**Vehicle**	**Concert**	**Animation**	**Music video**	**Video game**	**Motorsport**	**Football**

Table 1: Most frequent entities for each of the top-level categories.

YouTube-8M Dataset

- Frame-level data (total size of 1.71 TB)
  Each video has
  a. “video_id”: unique id for the video,
  b. “label”: list of labels of that video,
  c. Each frame has “rgb”: float array of length 1024,
  d. Each frame has “audio”: float array of length 128.

  A video v is given by a sequence of frame-level features (rgb) $x^v_{1:F_v}$,
  where x^v_j is the feature of the j-th frame from video v.

YouTube-8M Dataset

- Video-level data (total size of 31GB)
  Each video has
  a. “video_id”
  b. “labels”
  c. “mean_rgb”:float array of length 1028
  d. “mean_audio”: float array of length 128

  Formally, a video-level feature $\phi\left(x_{1:F_v}^{v}\right)$ is a fixed-length representation at the video-level. We use a simple aggregation technique for getting these video-level representation:
  1. First, second order and ordinal statistics
  2. Feature normalization

YouTube-8M Dataset

- **Video-level data (total size of 31GB)**
  1. **First, second order and ordinal statistics:**
     we extract the mean $\mu^v \in R^{1024}$ and the standard-deviation $\sigma^v \in R^{1024}$,
     and the top 5 ordinal statistics $Top_5\left(x^v_{1:F_v}\right)$ for each dimension

$$\phi\left(x^v_{1:F_v}\right) = \begin{bmatrix} \mu\left(x^v_{1:F_v}\right) \\ \sigma\left(x^v_{1:F_v}\right) \\ Top_5\left(x^v_{1:F_v}\right) \end{bmatrix}$$

  2. **Feature normalization:**
     we substract the mean $\phi(\cdot)$ then use PCA to decorelate and whiten the features.

Evaluation Metric

- For each entity e, we consider $(x_i, g_i^e)_{i=1,\dots,N}$ for a binary classifier, where N is the number of videos, $g_i^e \in \{0,1\}$ is the ground-truth which is 1 if labels of i-th video contains entity e, and zero otherwise.

- Mean Average Precision (mAP)
  - Let y_i be the annotation scores for i-th video.
  - For each entity e, we sort all the non-zero annotations according to the model score. (denoted as y_i^e)
  - At a given threshold τ, the precision $P^e(\tau)$ and recall $R^e(\tau)$ are given by

$$P^e(\tau) = \frac{\sum_i I(y_i^e \geq \tau) g_i^e}{\sum_i I\left(y_i^e \geq \tau\right)},$$

$$R^e(\tau) = \frac{\sum_i I(y_i^e \geq \tau) g_i^e}{\sum_i g_i^e}.$$

Evaluation Metric

- **Mean Average Precision (mAP)**
  - **The average precision, approximating the area under the precision-recall curve, can then be computed as**

$$AP^e = \sum_{j=1}^{10000} P(\tau_j)\left[R(\tau_j) - R(\tau_{j+1})\right],$$

where $\tau_j = j/10000.$

  - **The mean average precision is computed as the unweighted mean of all the per-class average precisions:**

$$mAP = \frac{\sum_e AP^e}{\#\ of\ entities}.$$