Deep Learning YouTube Video Tags

Travis Addair

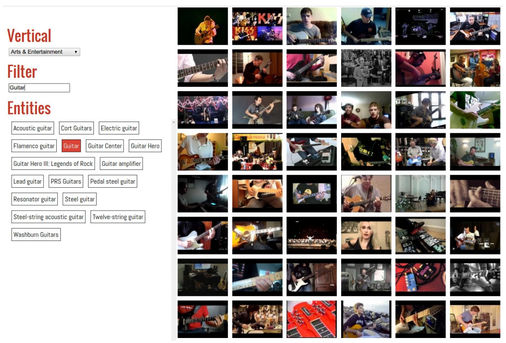
Introduction

Video tagging is a complex problem combining single-image feature extraction with arbitrarily long sequence understanding. Given a video containing image and audio features, the goal is to assign the video a set of tags that describe its content to humans.

Using Google's YouTube-8M video dataset, and named entity vectors provided by Google using word2vec, we propose a CNN-RNN architecture that structurally predicts the labels in relation to both the raw audio/visual features, and the correlation between the labels themselves.

Dataset

- 7 million YouTube videos.
- 4716 classes (tags) from Knowledge Graph.
- 1 to 31 tags per video.
- Entity vectors for tags mined from Freebase.
  - 100 billion entities.
  - 1000-dimensional.
  - 845 tags missing entity vectors.
- Tags generated with human curation.

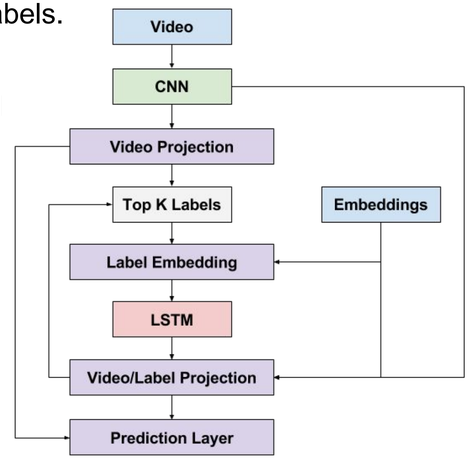


Technical Approach

CNN-RNN architectures have successfully been applied to the related problem of image captioning in the past. The primary differences between the problem of video tagging and image captioning are twofold:

1. Instead of a single frame of visual features, we have a sequence of visual and audio features.
2. Captioning implies an ordered sequence of labels, whereas tagging is an unordered set of distinct labels.

Our model:



- **CNN**: The input video is passed through an Inception network for each frame, and the frame-level features are aggregated into 1024 visual features and 128 audio features per video.
- **Video Projection:** Video Features are projected using a series of dense layers into label space, where every value is a rough confidence score for each label.
- **Top K Labels**: The top scoring labels are extracted, and pushed onto a stack for downstream evaluation.
- **Label Embedding**: The top label on the stack is removed and embedded into an entity (word) vector.
- **LSTM**: We perform one step forward for the LSTM to predict the next label given the current label.
- **Video/Label Projection**: The video features from the CNN are projected into entity vector space along with the output from the CNN, and their projected vectors are added together, then multiplied back with the embedding matrix to project back into label space and provide the new set of confidence values.
- **Prediction Layer**: The output of the Video/Label Projection is fed back to the Top K Labels layer as part of an iterative beam search algorithm that stops at a given depth. All resulting vectors of label confidence scores are aggregated in this layer to produce the final predictions.

Evaluation

We use Global Average Precision to assess the performance of our model:

$$\mathrm{AP} = \sum_{k=1}^{N} p(k)\Delta r(k)$$

Where p(i) is the precision of prediction i and r(i) is the recall.

Baseline models include both frame-level and video-level feature classifiers:

- Logistic Model (video): Single fully connected layer.
- Mixture of Experts (video): Ensemble classifier with a configurable number of logistic classifiers.
- Dense Model (video): Multiple fully-connected layers with ReLU activations, batch normalization, and dropout.
- Deep Bag of Features (frame): Clusters frame features and pools across frames.
- LSTM (frame): Each frame into an RNN.

Model	**GAP**
Logistic (video)	0.72
MoE (video)	0.76
Dense (video)	0.73
Deep Bag (frame)	0.71
LSTM (frame)	0.65
CNN-RNN	TBD
CNN-RNN + Audio	TBD