

# Google Cloud & YouTube-8M Video Understanding Challenge

- Starter Code -

김성현

서울대학교 통계학과

2017년 5월 4일

# 목차

- 1 Model
  - Data
  - Video-Level Model
  - Frame-Level Model
- 2 Code
  - Overview
  - Demo
- 3 Execution

# Outline

- 1 Model
  - Data
    - Video-Level Model
    - Frame-Level Model
- 2 Code
  - Overview
  - Demo
- 3 Execution

# Model

## Data

- 데이터는 비디오 ID, 라벨, 영상 feature와 음성 feature로 이루어짐.
- Video-Level 데이터에는 하나의 데이터 당 1024개의 영상 feature가 있다.
- Frame-Level 데이터에는 초당 1프레임, 프레임당 1024개의 feature가 있다. 최대 300프레임.
- Starter Code에서는 영상 feature만을 사용.

# Model Data

- 데이터 형식

# Outline

- 1 Model
  - Data
  - **Video-Level Model**
  - Frame-Level Model
- 2 Code
  - Overview
  - Demo
- 3 Execution

# Model

## Video-Level Model

- Video-Level 데이터에는 시간의 흐름이 없음.
- 로지스틱 모형과 Mixture of expert(Moe) 모형 사용.

# Model

## Video-Level Model: Mixture of expert(Moe)

- Moe는 여러 개의 모형  $f_1, \dots, f_K$ 와 그 모형들에 입력값에 따라 다른 가중치를 주는 함수  $g$ 로 이루어진다.

$$\sum_i^K g(i|x) f_i(x)$$



# Outline

- 1 Model
  - Data
  - Video-Level Model
  - **Frame-Level Model**
- 2 Code
  - Overview
  - Demo
- 3 Execution

# Model

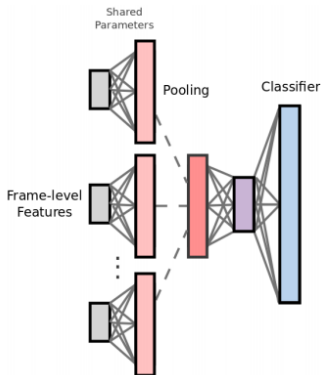
## Frame-Level Model

- Frame-Level Model는 Frame-Level feature를 Video-Level feature로 변환.
- 변환된 Video-Level feature를 Video-Level model에 적용시킴.
- Deep Bag of Frame (DBoF), Long Short-Term Memory (LSTM) 사용.

# Model

## Frame-Level Model: Deep Bag of Frame (DBoF)

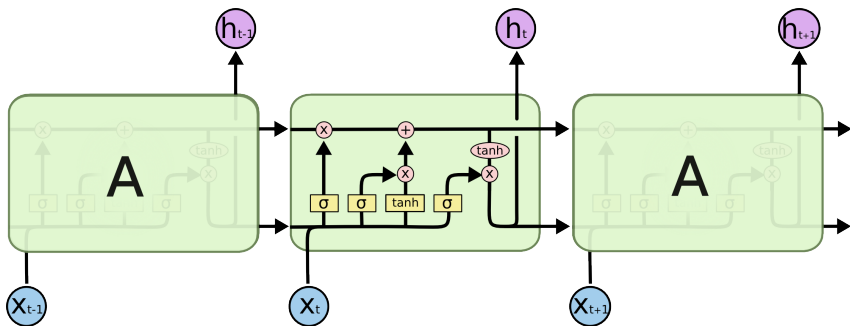
- DBoF에서는 프레임을 랜덤하게 뽑아 프레임별로 fully connected layer를 적용한 후 pooling한다. 이 때, 모든 프레임에서 fully connected layer를 공유한다.



# Model

## Frame-Level Model: Long Short-Term Memory (LSTM)

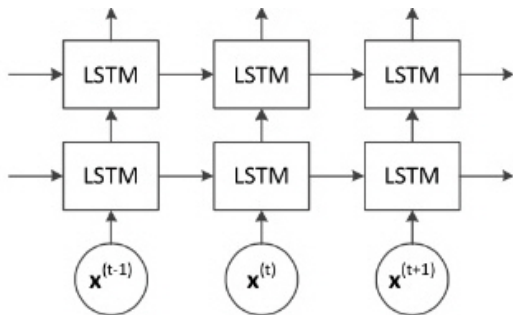
- 시간의 흐름을 반영한 모형으로 LSTM을 사용한다. LSTM cell의 구조는 다음과 같다.



# Model

## Frame-Level Model: Long Short-Term Memory (LSTM)

- 여러 층의 LSTM을 사용하기도 한다.
- 마지막 프레임, 마지막 층의  $h_t$ 를 Video-Level feature로 사용한다.



# Outline

- 1 Model
  - Data
  - Video-Level Model
  - Frame-Level Model
- 2 **Code**
  - **Overview**
  - Demo
- 3 Execution

# Overview

- main 함수 포함: train, eval, inference
- 기타 모듈:
  - 1.eval\_util
  - 2.export\_model
  - 3.losses
  - 4.frame\_level\_models
  - 5.video\_level\_models
  - 6.readers
  - 7.utils

# Outline

- 1 Model
  - Data
  - Video-Level Model
  - Frame-Level Model
- 2 Code
  - Overview
  - Demo
- 3 Execution

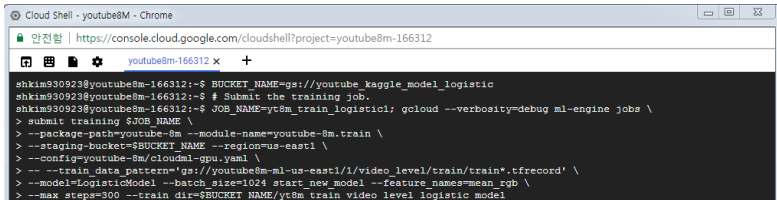


# Demo

# Execution

- 파이썬 2 형식으로 작성되었으며 파이썬 2.7+ 버전과 텐서플로우 1.0.0 이상의 버전이 필요하다.
- 구글클라우드를 이용해 시험한 결과 로컬 머신으로 실행할 경우 video-level 로지스틱 모형을 제외한 모형에서 프로세스가 종료됨.
- cloud에서는 모든 모형의 학습이 가능하고 yaml파일을 이용하면 GPU를 사용할 수 있다.
- 텐서보드와 로그로 학습 상황을 알 수 있음.

# Execution:shell



```
Cloud Shell - youtube8M - Chrome
안전함 | https://console.cloud.google.com/cloudshell?project=youtube8m-166312
youtube8m-166312 x +
shkim930923@youtube8m-166312:~$ BUCKET_NAME=gs://youtube_kaggle_model_logistic
shkim930923@youtube8m-166312:~$ # Submit the training job.
shkim930923@youtube8m-166312:~$ JOB_NAME=yt8m_train_logistic1; gcloud --verbosity=debug ml-engine jobs \
> submit training $JOB_NAME \
> --package-path=youtube-8m --module-name=youtube-8m.train \
> --staging-bucket=$BUCKET_NAME --region=us-east1 \
> --config=youtube-8m/cloudml-gpu.yaml \
> -- --train_data_pattern='gs://youtube8m-ml-us-east1/1/video_level/train/train*.tfrecord' \
> --model=LogisticModel --batch_size=1024 start_new_model --feature_names=mean_rgb \
> --max_steps=300 --train_dir=$BUCKET_NAME/yt8m_train_video_level_logistic_model
```

# Execution:log

Google Cloud Platform ▶ youtube8M

Stackdriver  
로그 기록

로그

로그 기반 측정 항목  
나보내기  
리소스 사용량

측정 항목 만들기  
나보내기 만들기

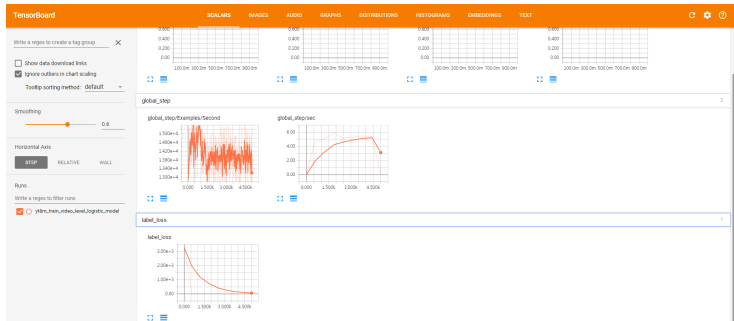
하셀 또는 워크스페이스 기준 필터링

Cloud ML 작업 yt8m\_train\_0 | 최근 로그 | 최근 로그 수준 | 이동할 날짜

2017-05-03 JST

▶	06:10:16.337	training step 49		Loss: 2268.02	Examples/sec: 614.70
▶	06:10:16.638	training step 50		Loss: 2247.89	Examples/sec: 588.06   Hit@1: 0.63 FERR: 0.47 GAP: 0.39
▶	06:10:17.308	training step 51		Loss: 2218.85	Examples/sec: 790.97
▶	06:10:17.518	training step 52		Loss: 2214.31	Examples/sec: 612.45
▶	06:10:17.726	training step 53		Loss: 2185.72	Examples/sec: 618.04
▶	06:10:17.933	training step 54		Loss: 2172.43	Examples/sec: 617.48
▶	06:10:18.141	training step 55		Loss: 2151.19	Examples/sec: 617.62
▶	06:10:18.350	training step 56		Loss: 2136.83	Examples/sec: 615.51
▶	06:10:18.558	training step 57		Loss: 2117.95	Examples/sec: 617.63
▶	06:10:18.773	training step 58		Loss: 2107.17	Examples/sec: 597.63
▶	06:10:18.975	training step 59		Loss: 2093.14	Examples/sec: 634.23
▶	06:10:19.287	training step 60		Loss: 2067.29	Examples/sec: 617.77   Hit@1: 0.73 FERR: 0.55 GAP: 0.46

# Execution:tensor board



# Reference

- YouTube8M: A Large-Scale Video Classification Benchmark.