Encoding Video and Label Priors for Multi-label Video Classification on YouTube-8M dataset

Team SNUVL X SKT (**8th Ranked**)





Seil Na[1] Youngjae Yu[1] Sangho Lee[1] Jisung Kim[2] Gunhee Kim[1]

1 SEOUL NATIONAL UNIV. VISION & LEARNING

2 SK telecom

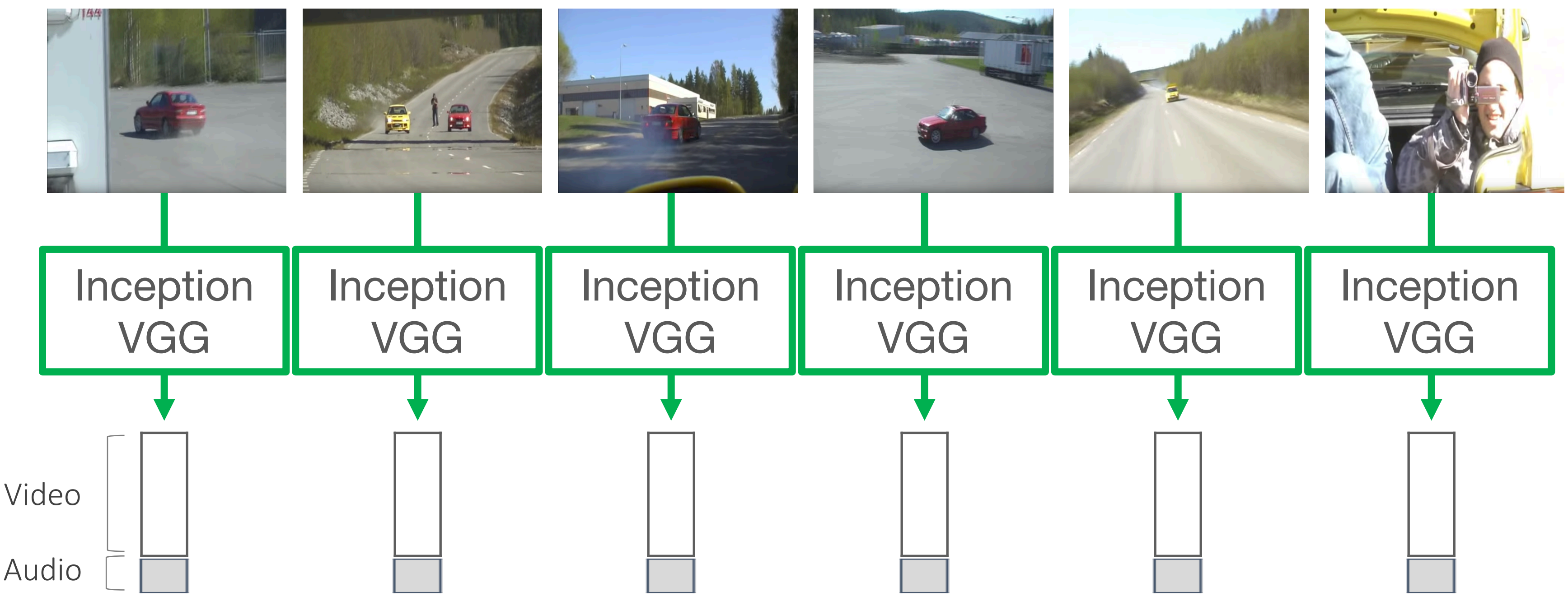
Code : https://github.com/seilna/youtube8m

Contents

- YouTube-8M Video Multi-label Classification
- Our approach
    - Video Pooling Layer
    - Classification Layer
    - Label Processing Layer
    - Loss Function
- Results

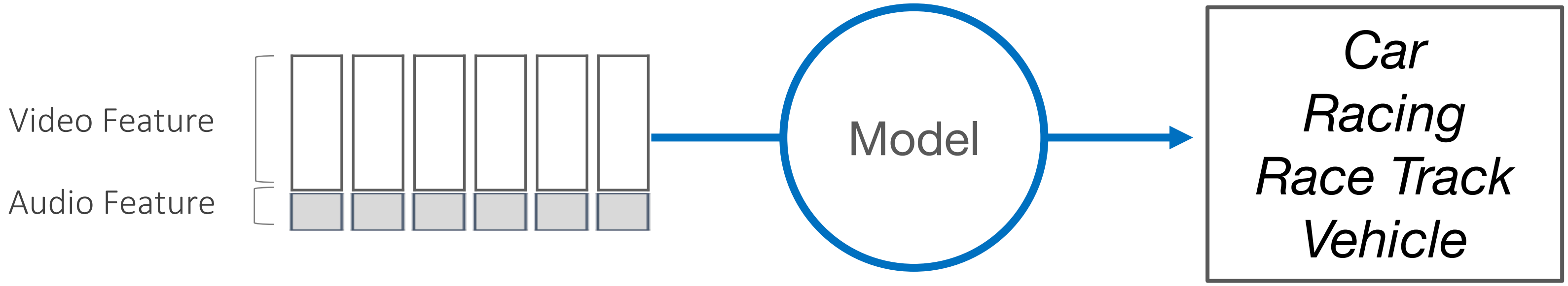
YouTube-8M Video Multi-label Classification

- Input: videos (with audio) with maximum 300 seconds long
- Video and audio are given in feature form, extracted using Inception Network and VGG



YouTube-8M Video Multi-label Classification

- Output: given a test video and audio feature, model produces a multi-label prediction score for 4,716 classes



YouTube-8M Video Multi-label Classification

- Evaluation: among scores for all classes, only top 20 scores are considered
- Google Average Precision (GAP) is used to evaluate performance of model

$$GAP = \sum_{i=1}^{N} p(i)\Delta r(i)$$

Three Key Issues

- Our approach tackles THREE issues

i) Video pooling method (representation)
ii) Label imbalance problem
iii) Correlation between labels

Three Key Issues

- Our approach tackles THREE issues

i) Video pooling method (Representation)
   - Encode T frame features into a compact vector
   - Encoder should capture the content distribution of frames and temporal information of the sequence

ii) Label imbalance problem

iii) Correlation between labels

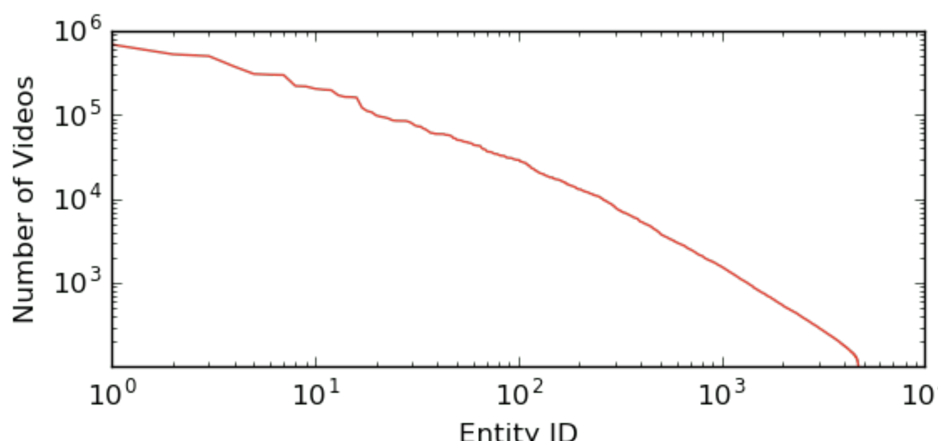
Three Key Issues

- Our approach tackles THREE issues

i) Video pooling method

ii) Label imbalance problem

  - In YouTube-8M dataset, the numbers of instances for each class are very different
  - How can we generalize well on small sets in the validation/test dataset?



Three Key Issues

- Our approach tackles THREE issues

i) Video pooling method

ii) Label imbalance problem

iii) Correlation between labels

Vertical

All

Filter

mario

Entities

Mario Kart (3658)
Super Mario Bros. (3136)
Super Mario World (1232)
New Super Mario Bros (1152)
Super Mario Galaxy (936)
New Super Mario Bros. Wii (700)

Three Key Issues

- Our approach tackles THREE issues

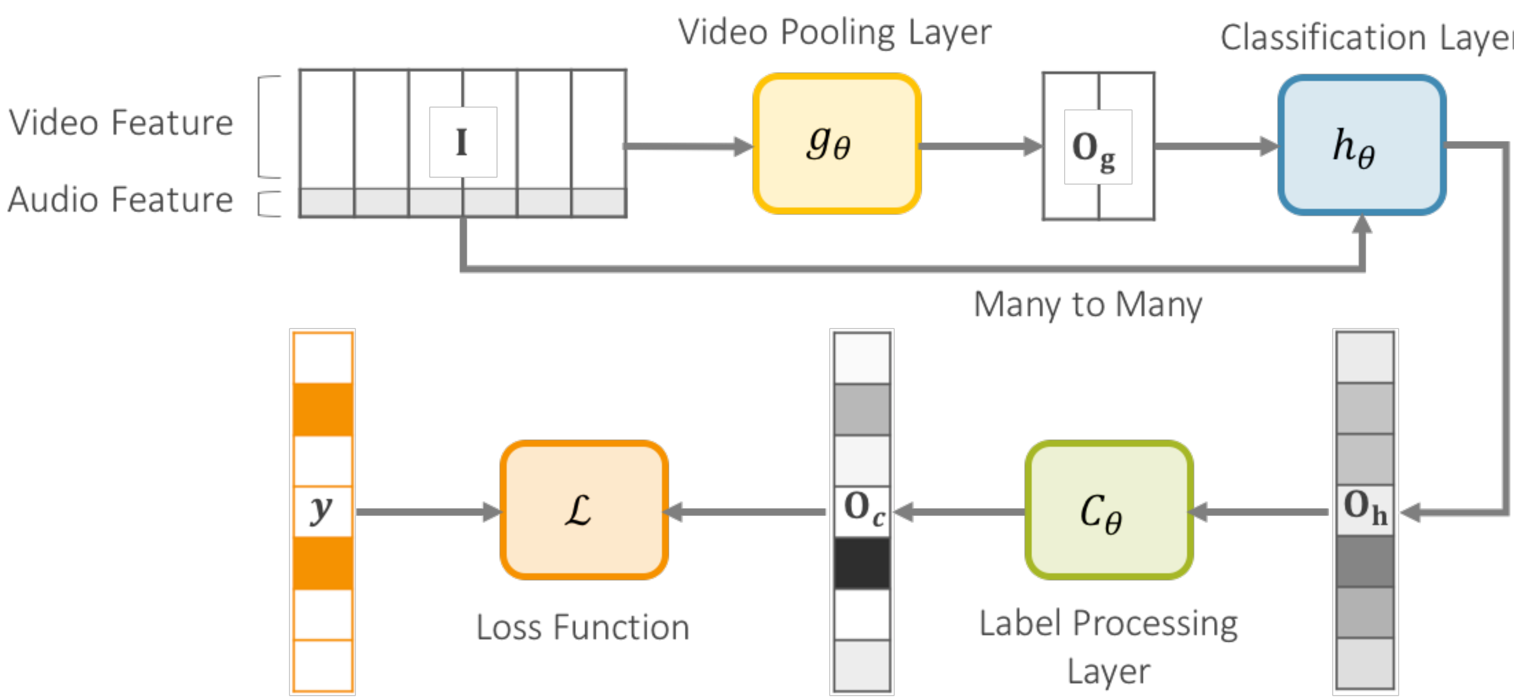
i) Video pooling method

ii) Label imbalance problem

iii) Correlation between labels
   - Some labels are semantically interrelated
   - Connected labels tend to appear in the same video
   - How can we use this prior to improve classification performance?

Our approach

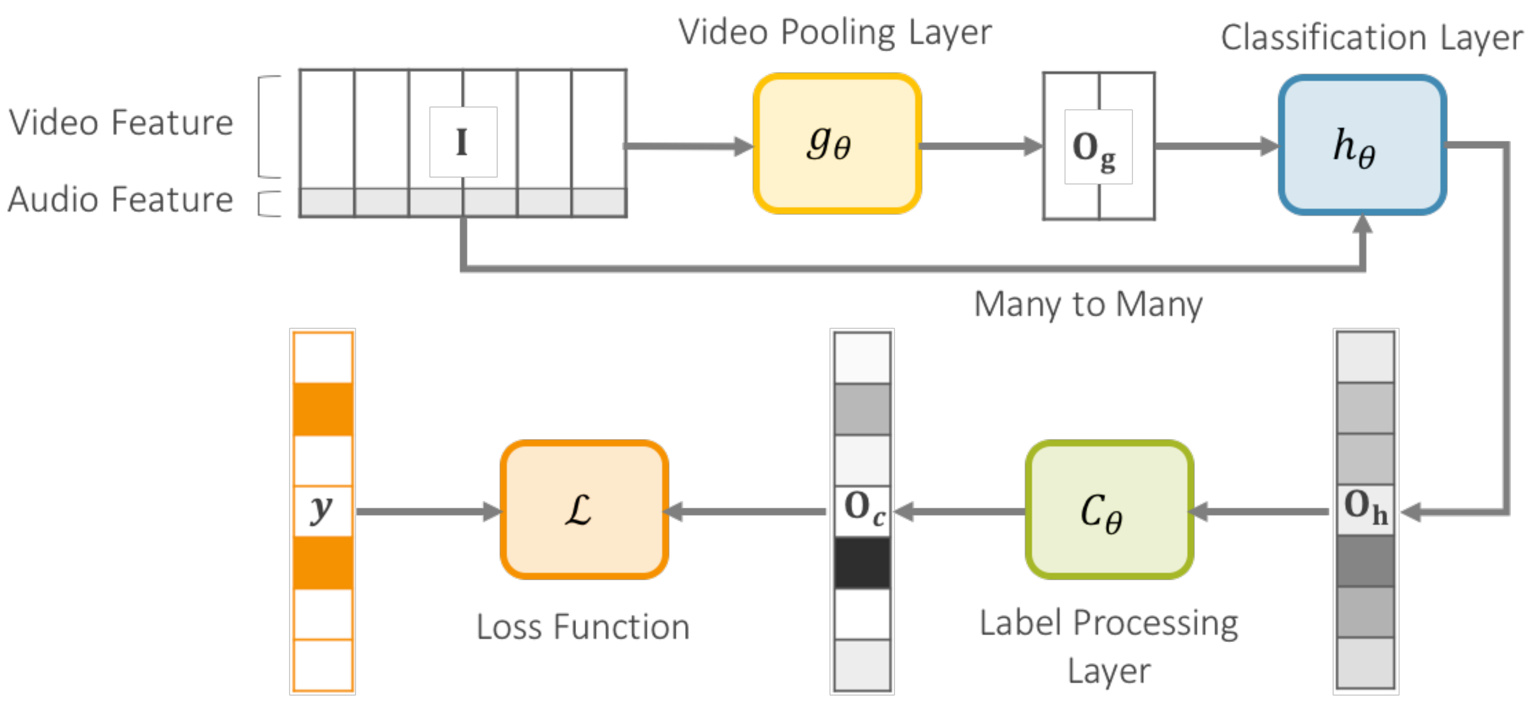
- Our model consists of FOUR components
  I. Video pooling layer
  II. Classification layer
  III. Label processing layer
  IV. Loss function



Our approach

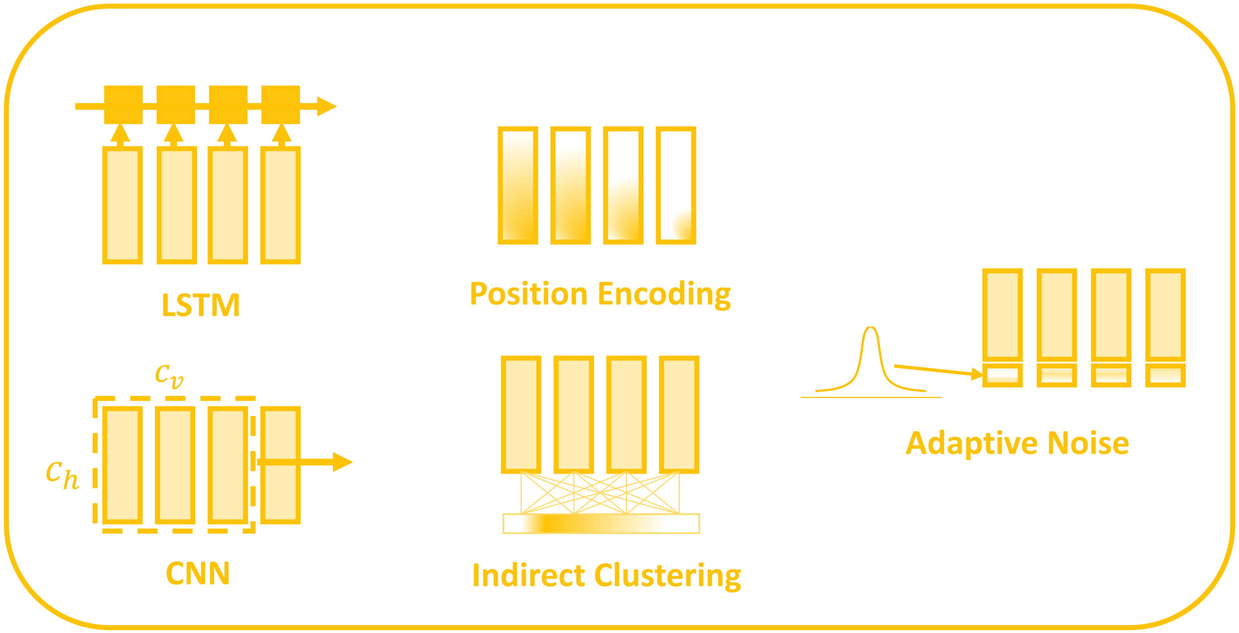
- Our model consists of FOUR components
  I. Video pooling layer 1,2
  II. Classification layer
  III. Label processing layer 3
  IV. Loss function 2

1. Video pooling method
2. Label imbalance problem
3. Correlation between labels



Video Pooling Layer

- Video pooling layer g_θ: $\mathbb{R}^{T \times 1,152} \rightarrow \mathbb{R}^d$ encodes T frame vectors into a compact vector
- Experiment following 5 methods

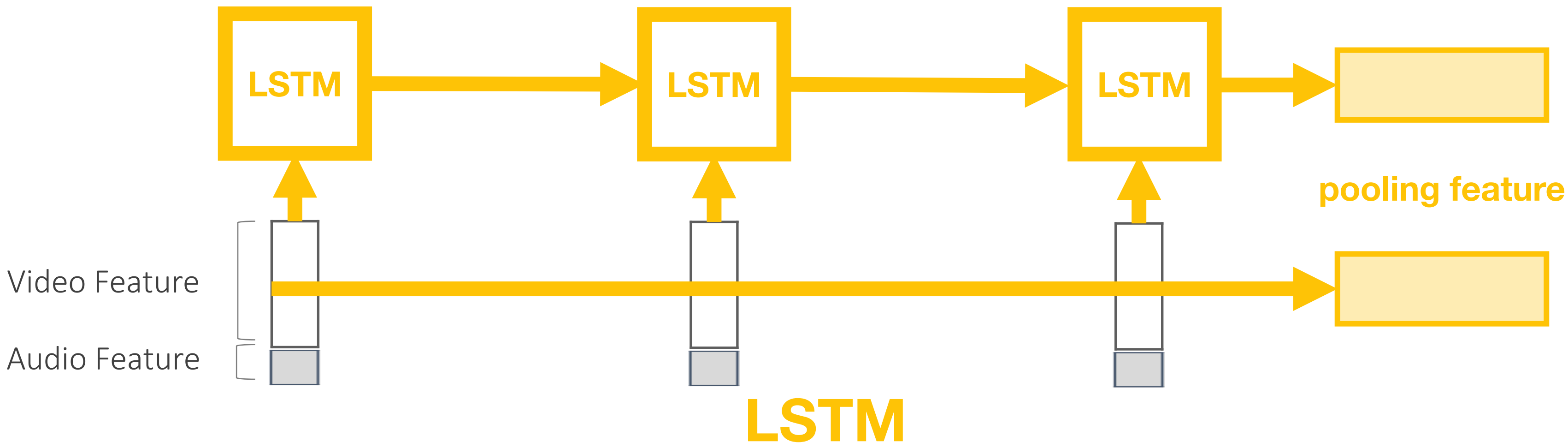


(a) Video Pooling Layer g_θ

Video Pooling Layer

1. LSTM

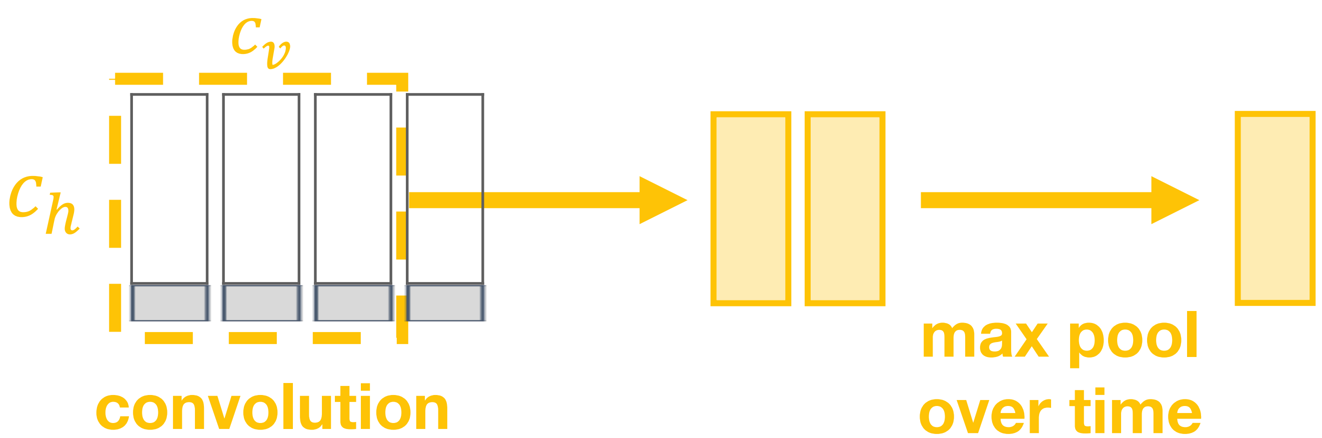
- Each frame vector is the input of LSTM
- All states vectors and the average of input vectors are used



Video Pooling Layer

2. CNN

- Use convolution operation like [Kim 2014].
- Adjacent frame vectors are regarded together

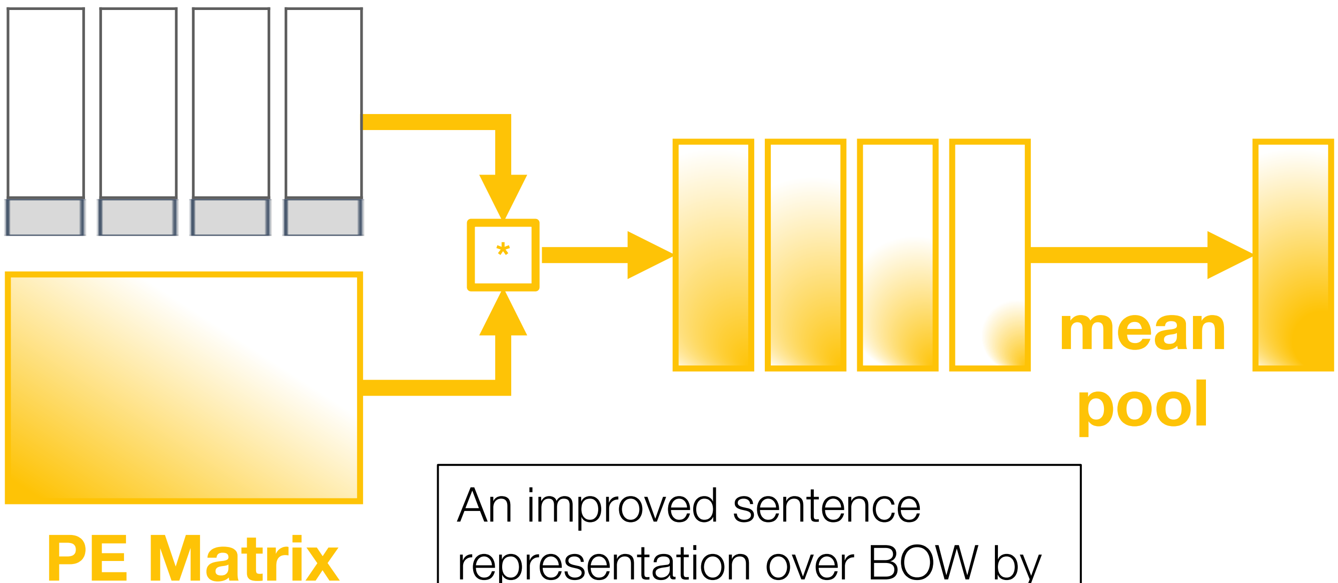


Kim, Yoon. "Convolutional neural networks for sentence classification."arXiv:1408.5882, 2014

Video Pooling Layer

3. Position Encoding

- Use the position encoding matrix [E2EMN] to represent the sequence order



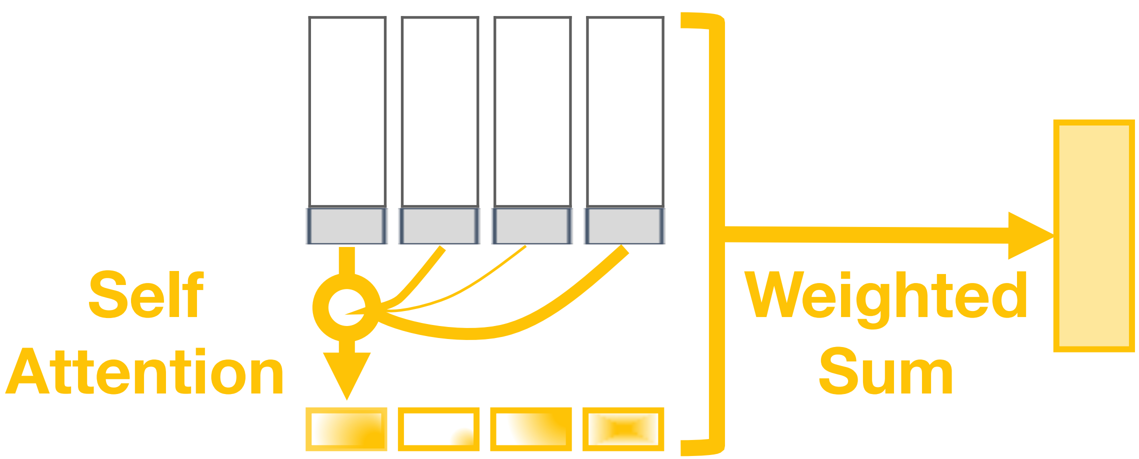
An improved sentence representation over BOW by considering word order

Sukhbaatar et al. "End-to-end memory networks." NIPS 2015.

Video Pooling Layer

4. Indirect Clustering

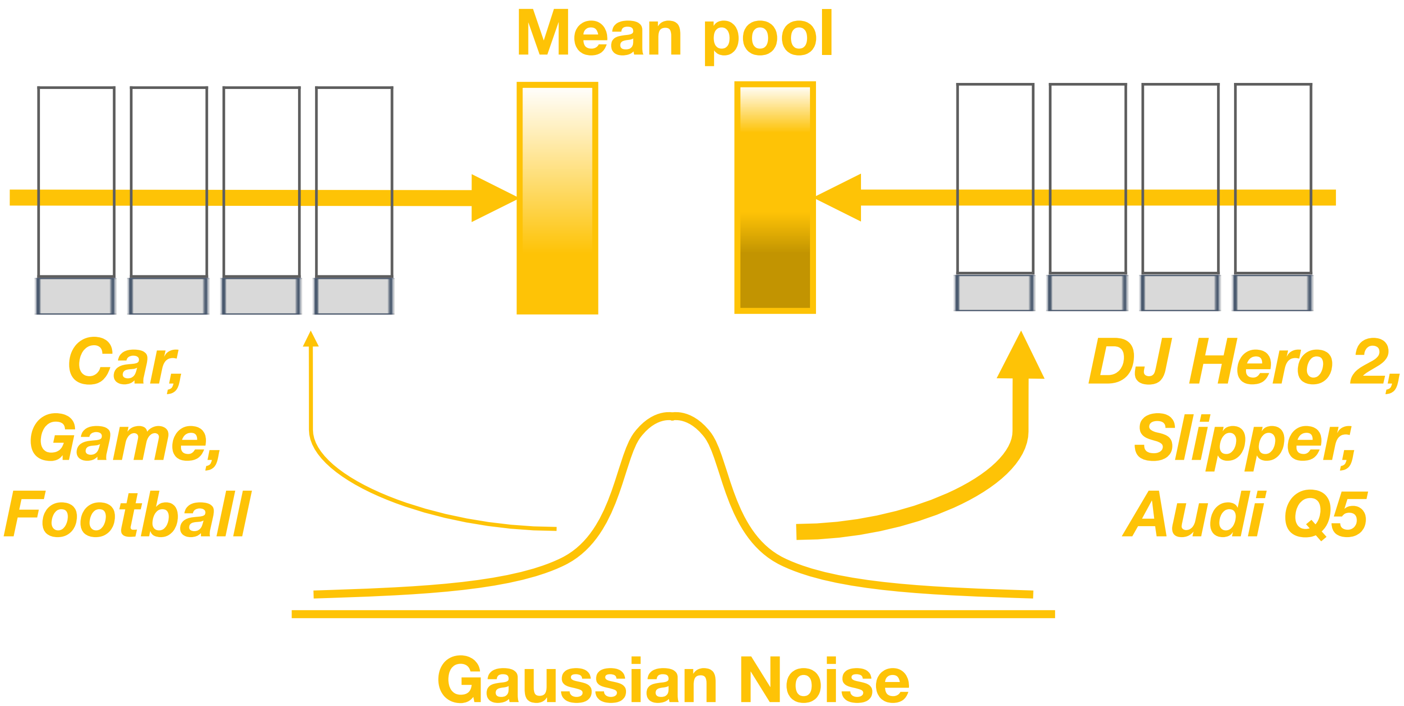
- We implicitly cluster frames via self-attention mechanism



Video Pooling Layer

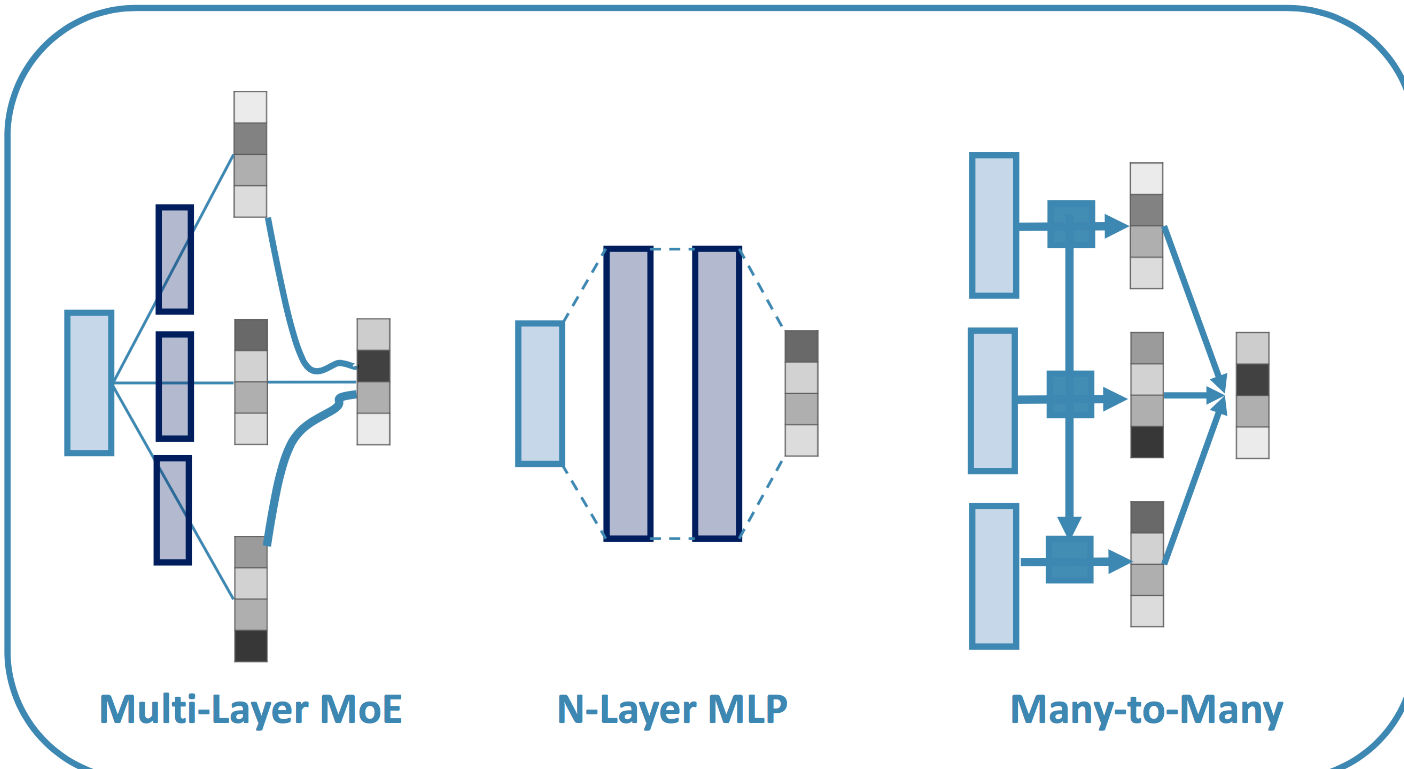
5. Adaptive Noise

- To deal with label imbalance, inject more noise to features of a video with rare labels, and less noise to videos with common labels



Classification Layer

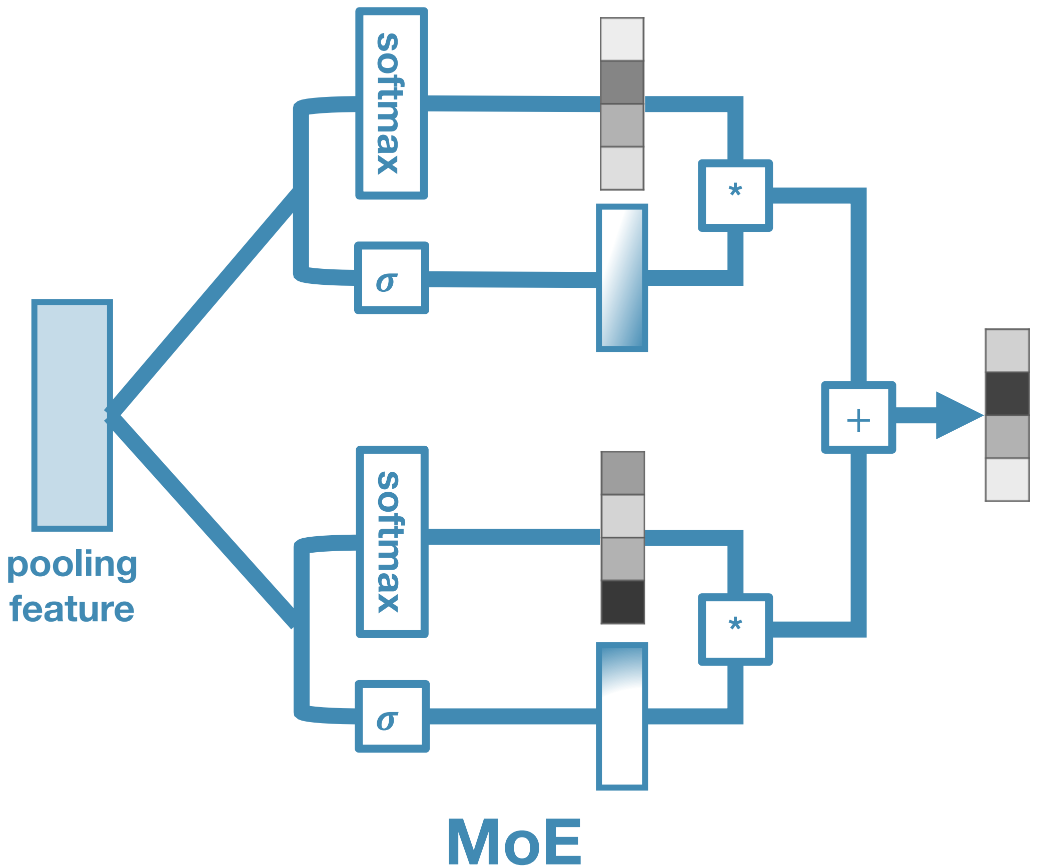
- Given pooled video features, the Classification Layer h_θ: $\mathbb{R}^d \rightarrow \mathbb{R}^{4,716}$ outputs a class score
- Experiment following 3 methods



Classification Layer

1. Multi-layer Mixture of Experts

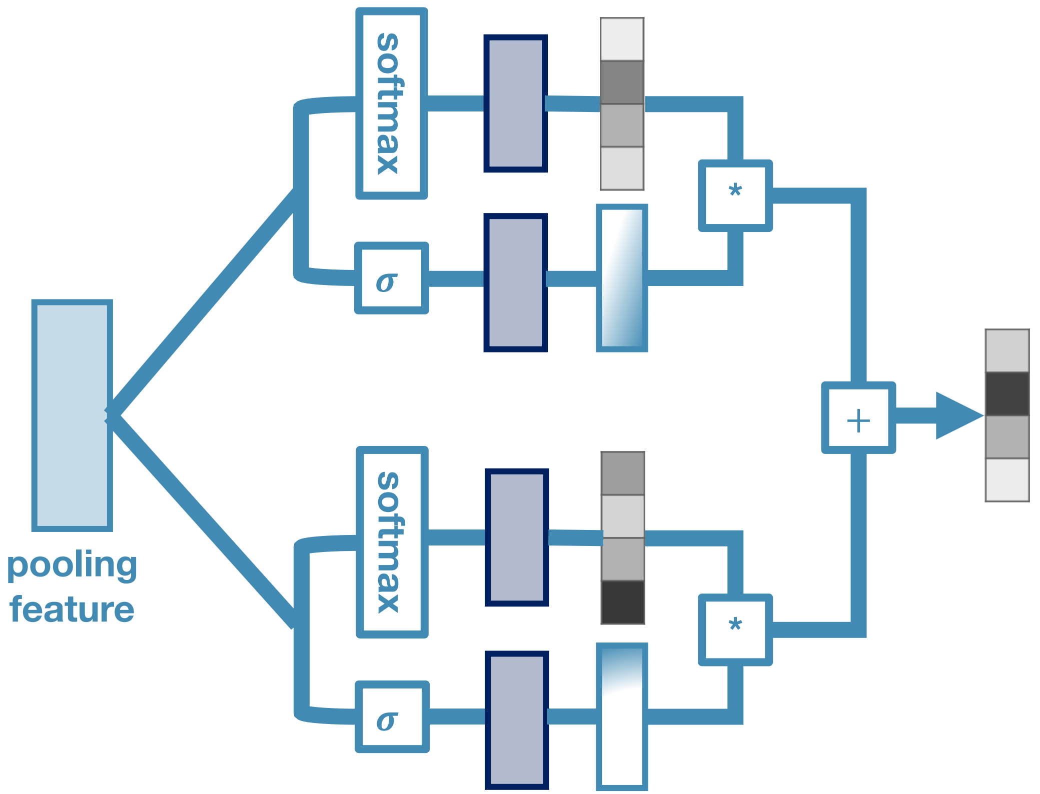
- Simply expand the existing MoE model



Classification Layer

1. Multi-layer Mixture of Experts

- Simply expand the existing MoE model

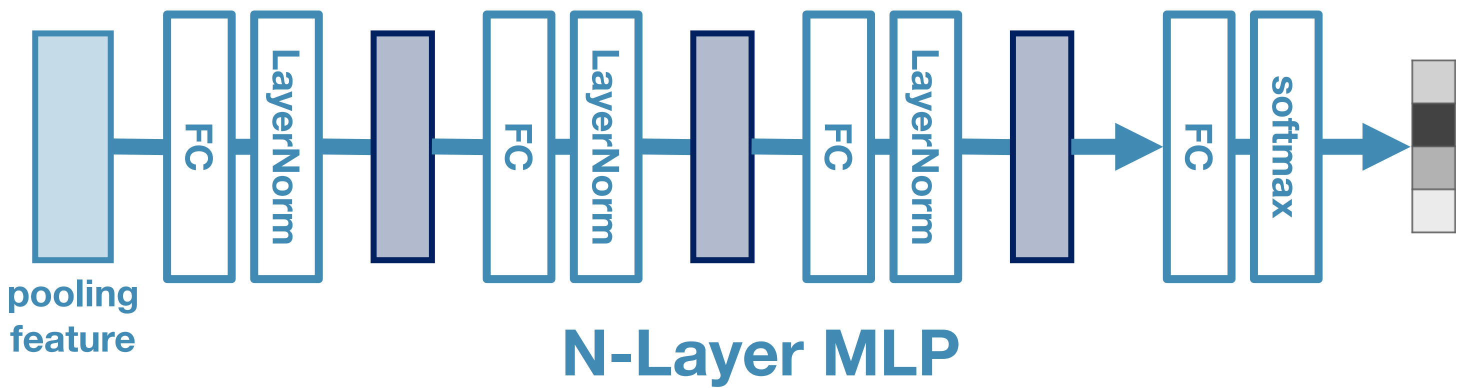


Multi-layer MoE

Classification Layer

2. N-Layer MLP

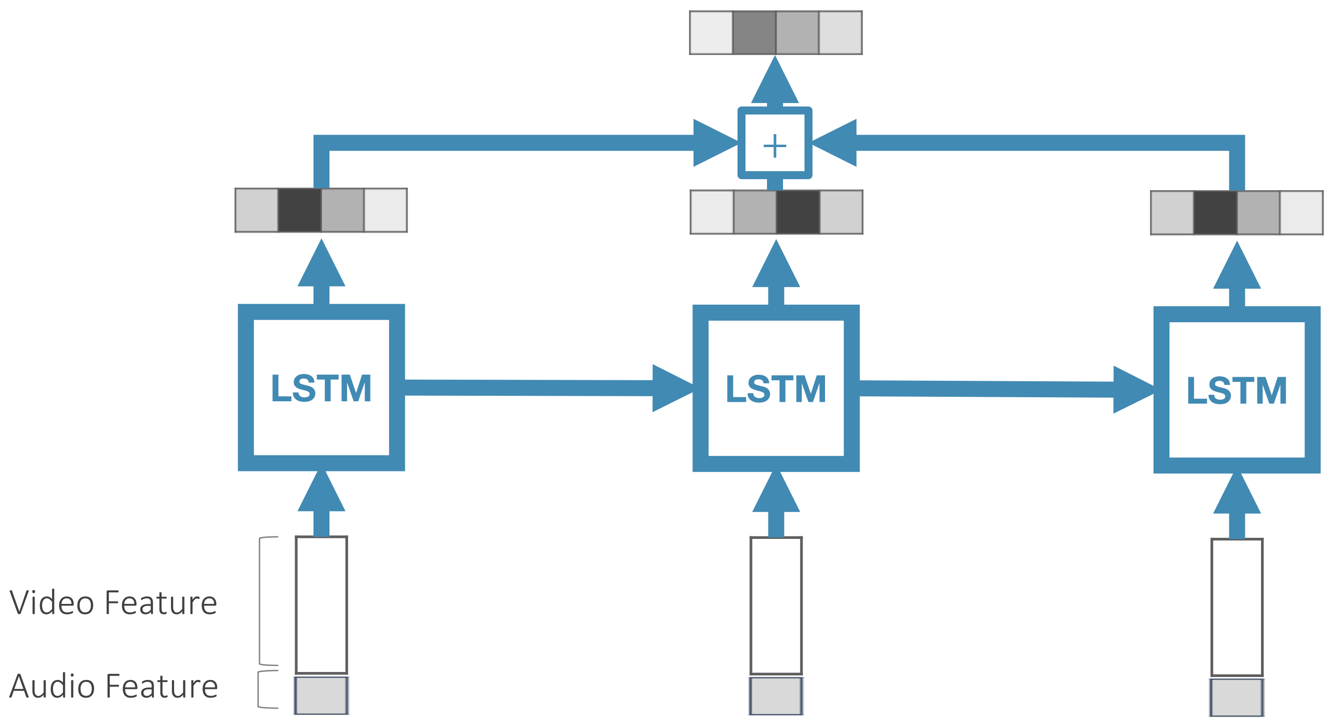
- A stack of fully connected layer
- Empirically, three layers with layer normalization



Classification Layer

3. Many-to-Many

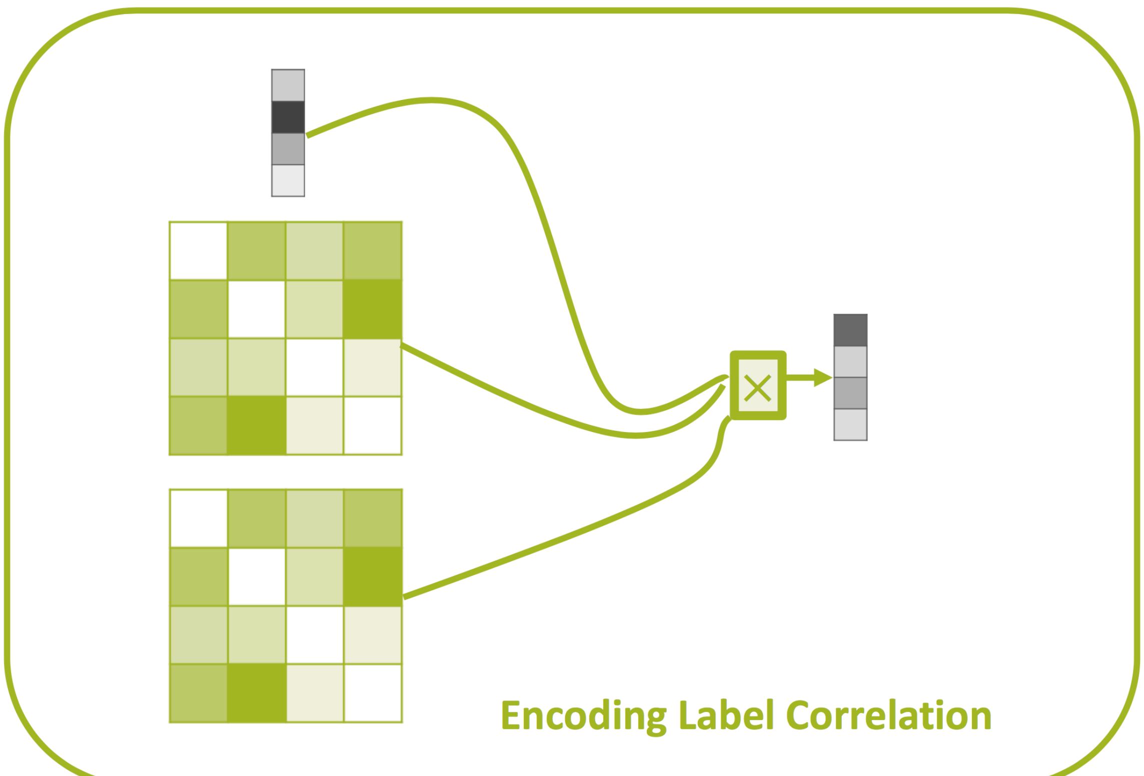
- Each frame vector is the input of LSTM
- Output is an average of score for each time step



Many-to-Many

Label Processing Layer

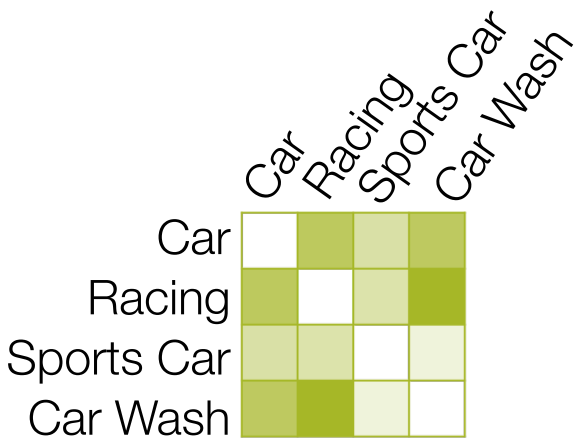
- Label Processing Layer C_θ update the class score using prior for correlation between labels
- Experiment following 1 method



Label Processing Layer

1. Encoding Label Correlation

- Construct a correlation matrix by counting the labels that appear in the same videos

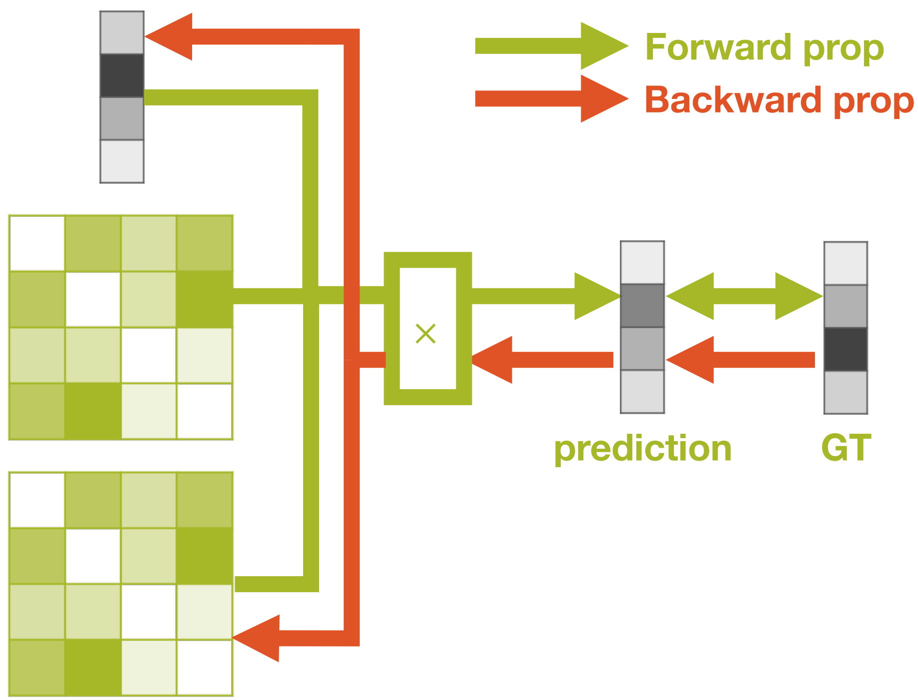


Label Processing Layer

1. Encoding Label Correlation

- Update the score using the correlation matrix

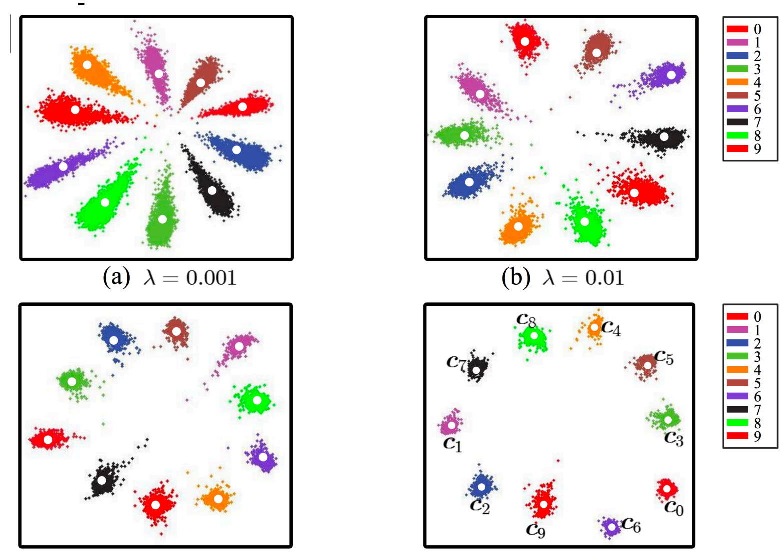
$$O_c = \alpha \cdot O_h + \beta \cdot M_c O_h + \gamma \cdot M_c{}' O_h$$



Loss Function

1. Center Loss

- Assign a penalty for the embedding of video belonging to the same label
- Add the center loss term to cross-entropy label loss at a predefined



(a) $\lambda = 0.001$ (b) $\lambda = 0.01$ (c) $\lambda = 0.1$ (d) $\lambda = 1$

Wen et al. "A discriminative feature learning approach for deep face recognition." ECCV 2016.

Loss Function

2. Huber Loss

- A combination of L1 and L2 loss to be robust against noisy labels
- Use pseudo-huber loss of cross entropy for fully-differentiable form
- $$\mathcal{L} = \delta^2 \left(\sqrt{1 + \left(\frac{\mathcal{L}_{CE}}{\delta} \right)^2} - 1 \right)$$

Results – Video Pooling Layer

Method	GAP@20
LSTM	0.811
LSTM-M	0.815
LSTM-M-O	**0.820**
LSTM-M-O-LN	0.815
CNN-64	0.704
CNN-256	0.753
CNN-1024	-
Position Encoding	0.782
Indirect Clustering	0.801
Adaptive Noise	0.782
mean pooling	0.747

- The LSTM family showed the best accuracies
- The more the distribution information is in the LSTM state, the better the performance is

Results – Classification Layer

Method	GAP@20
Many-to-Many	0.791
2 Layer MoE-2	0.424
2 Layer MoE-16	0.421
3 Layer MLP-4096	0.802
3 Layer MLP-4096-LN	**0.809**
MoE-2	0.747
MoE-16	0.796

- Multi-layer MLP showed the best performance
- LN made an improvement unlike LSTM in the video pooling layer

Results – Label Processing Layer

Method	GAP@20
`MoE - (1.0, 0.3, 0.0)`	0.784
`MoE - (1.0, 0.1, 0.0)`	0.787
`MoE - (1.0, 0.0, 0.1)`	0.788
`MoE - (1.0, 0.01, 0.0)`	**0.790**
`MoE - (1.0, 0.0, 0.01)`	**0.790**
`MoE - (1.0, 0.01, 0.01)`	0.788

- In all combinations, label processing had little impact on performance improvement
- It implies that a more sophisticated model is needed to deal with correlation between labels

Results – Loss Function

Method	GAP@20
$\mathcal{L}_{CE}$	0.798
$\mathcal{L}_{CE}$ + $\mathcal{L}_c(\lambda = 0.001)$	0.799
$\text{Huber}_{CE}(\delta = 0.5)$	**0.803**
$\text{Huber}_{CE}(\delta = 1.0)$	0.801
$\text{Huber}_{CE}(\delta = 2.0)$	0.798
$\text{Huber}_{CE}(\delta = 3.0)$	0.794

- The Huber loss is helpful to handle noisy labels or label imbalance problems

Conclusion

Video Pooling Layer

- Even for the "video" classification, the content distribution information of the frame vectors had a great impact on performance
- Future Work
  1. How to incorporate temporal information well?
  2. A better pooling method for both distribution and temporal information (e.g. RNN-FV)?

Lev et al. "RNN Fisher Vectors for Action Recognition and Image Annotation." ECCV 2016.

Conclusion

Label Processing Layer

- Correlation between labels was treated too naively in our work
- Future work
    1. A more sophisticated approach for it?

Loss function

- With the same label distribution in the current train/val/test split, there may be no need to address the label imbalance issue (for final accuracy)