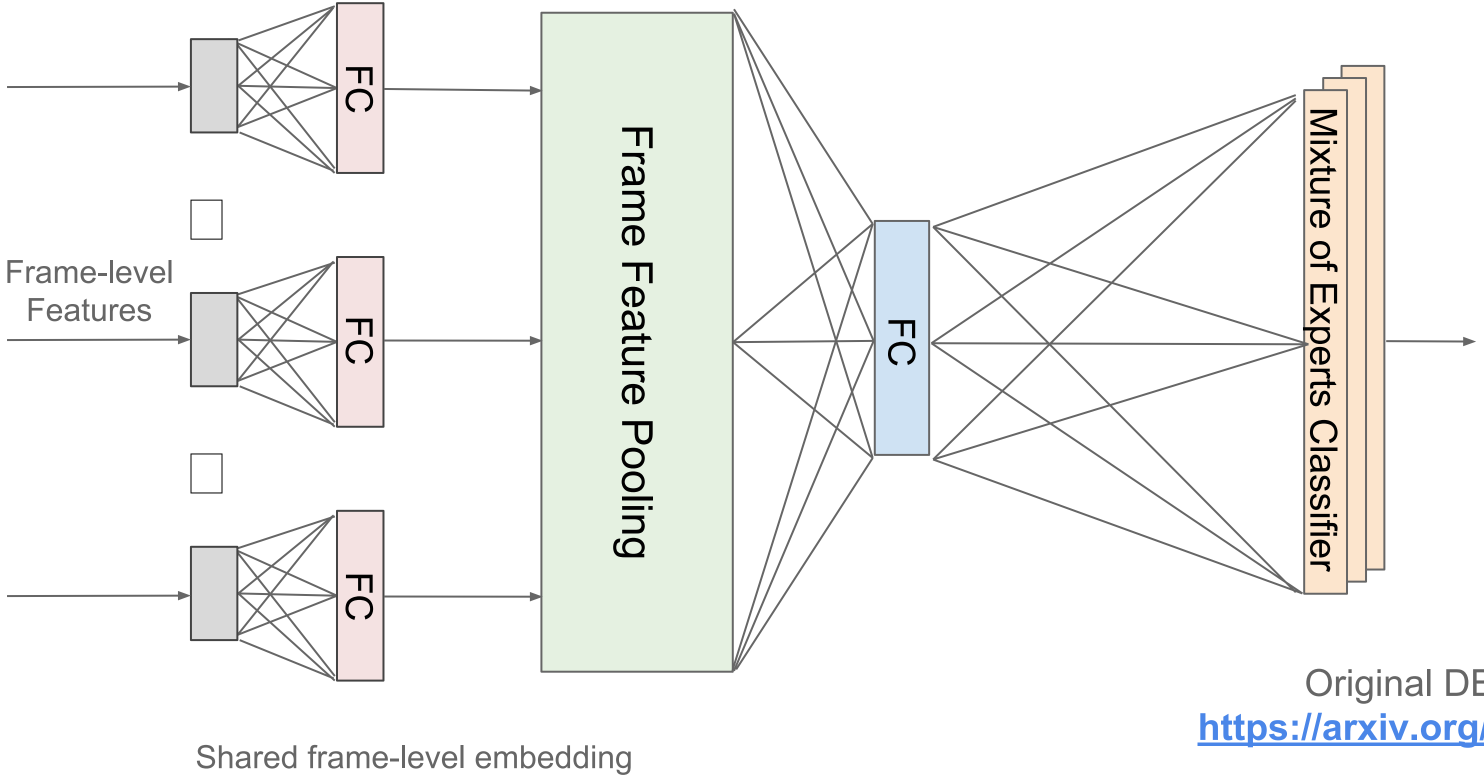
Context-Gated DBoF Models for YouTube-8M

Paul Natsev

natsev@google.com

Deep Bag of Frames (DBoF) Recap

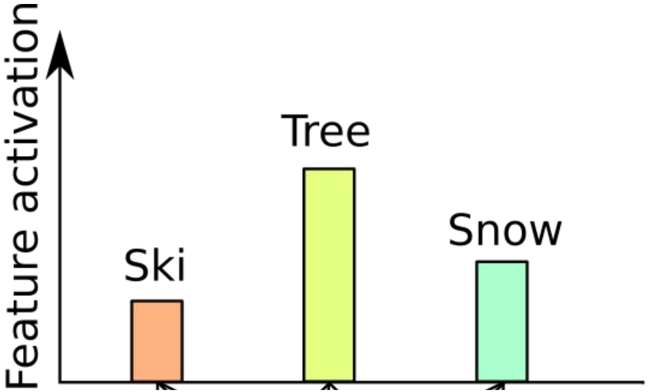
FC = Fully-connected layer + batch norm + Sigmoid activation



Original DBoF paper:
https://arxiv.org/abs/1609.08675

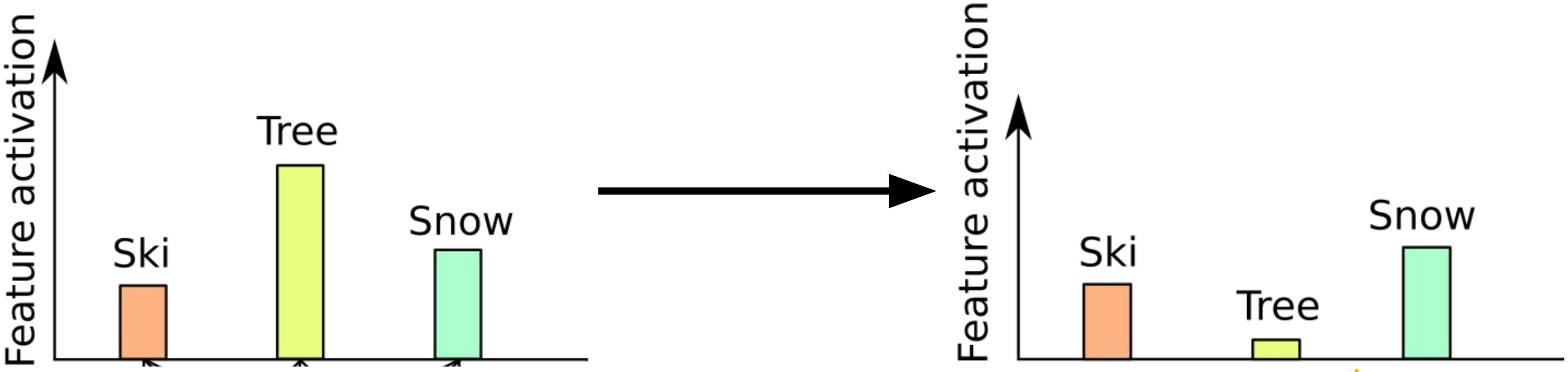
Context Gating Recap

Miech et al., "*Learnable pooling with Context Gating for Video Classification*", arxiv.org/abs/1706.06905



Context Gating Recap

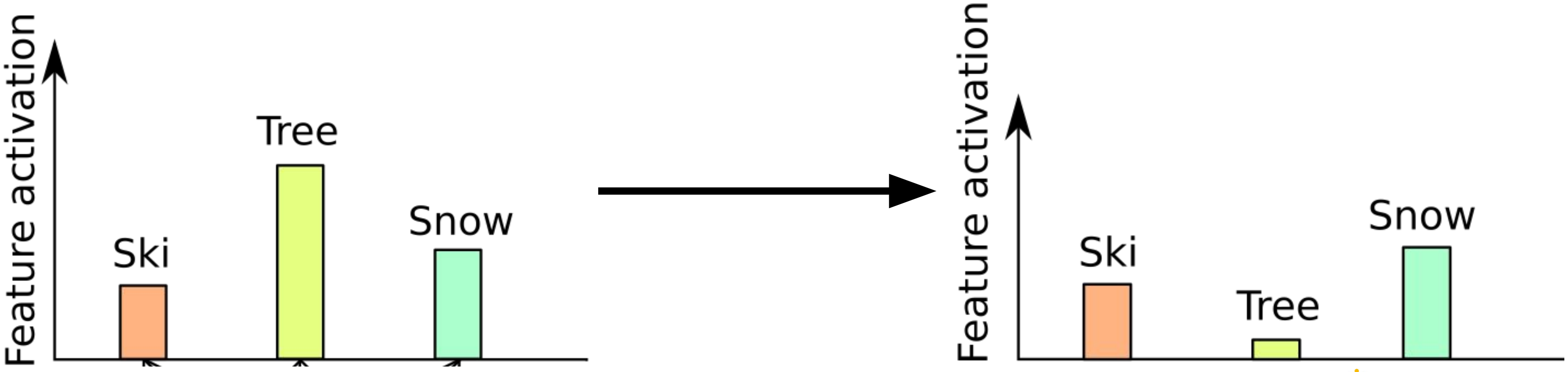
Miech et al., "*Learnable pooling with Context Gating for Video Classification*", arxiv.org/abs/1706.06905



Context Gating Recap

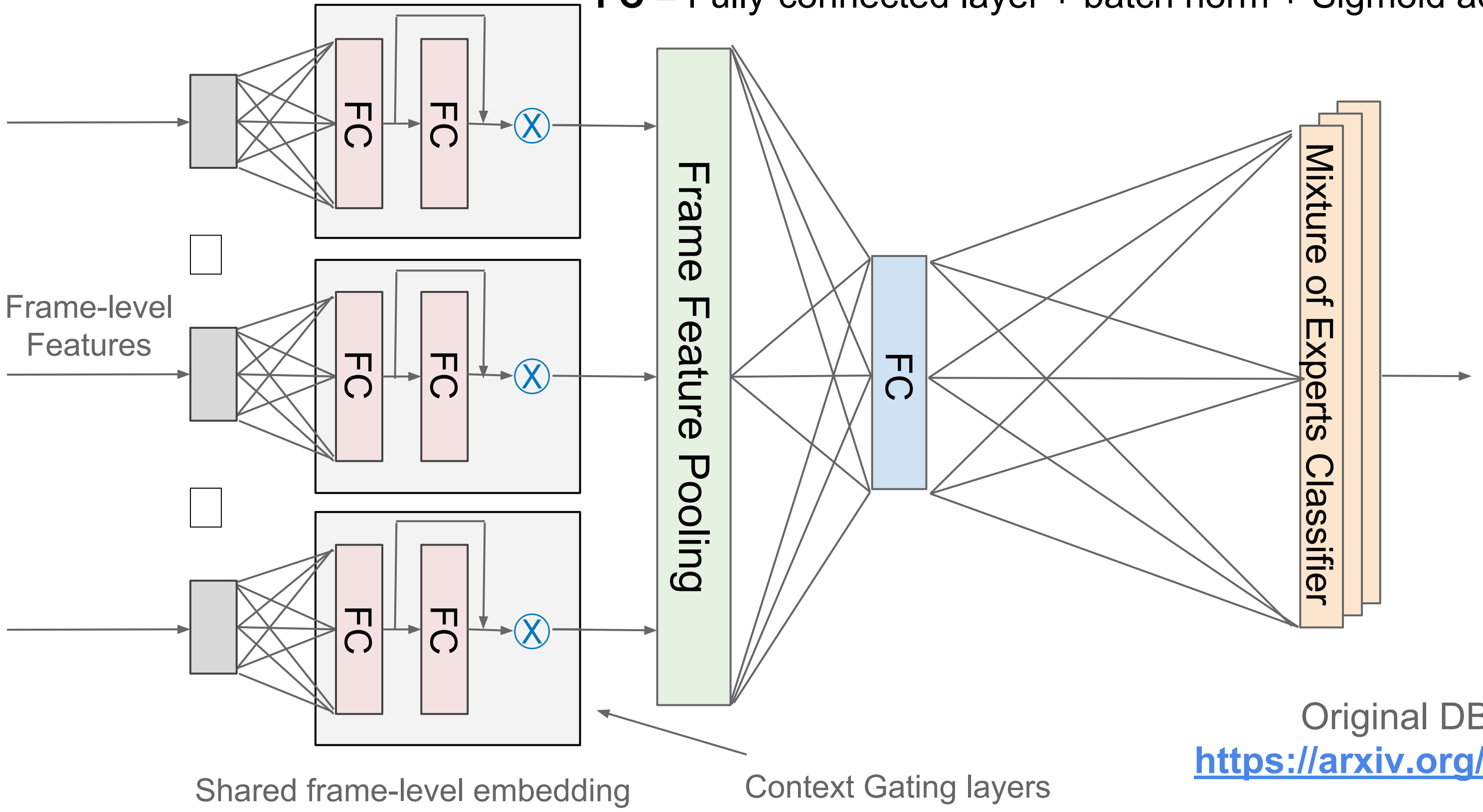
Miech et al., "*Learnable pooling with Context Gating for Video Classification*", arxiv.org/abs/1706.06905

$$Y = \sigma(WX + b) \circ X$$



Context-Gated Deep Bag of Frames (DBoF)

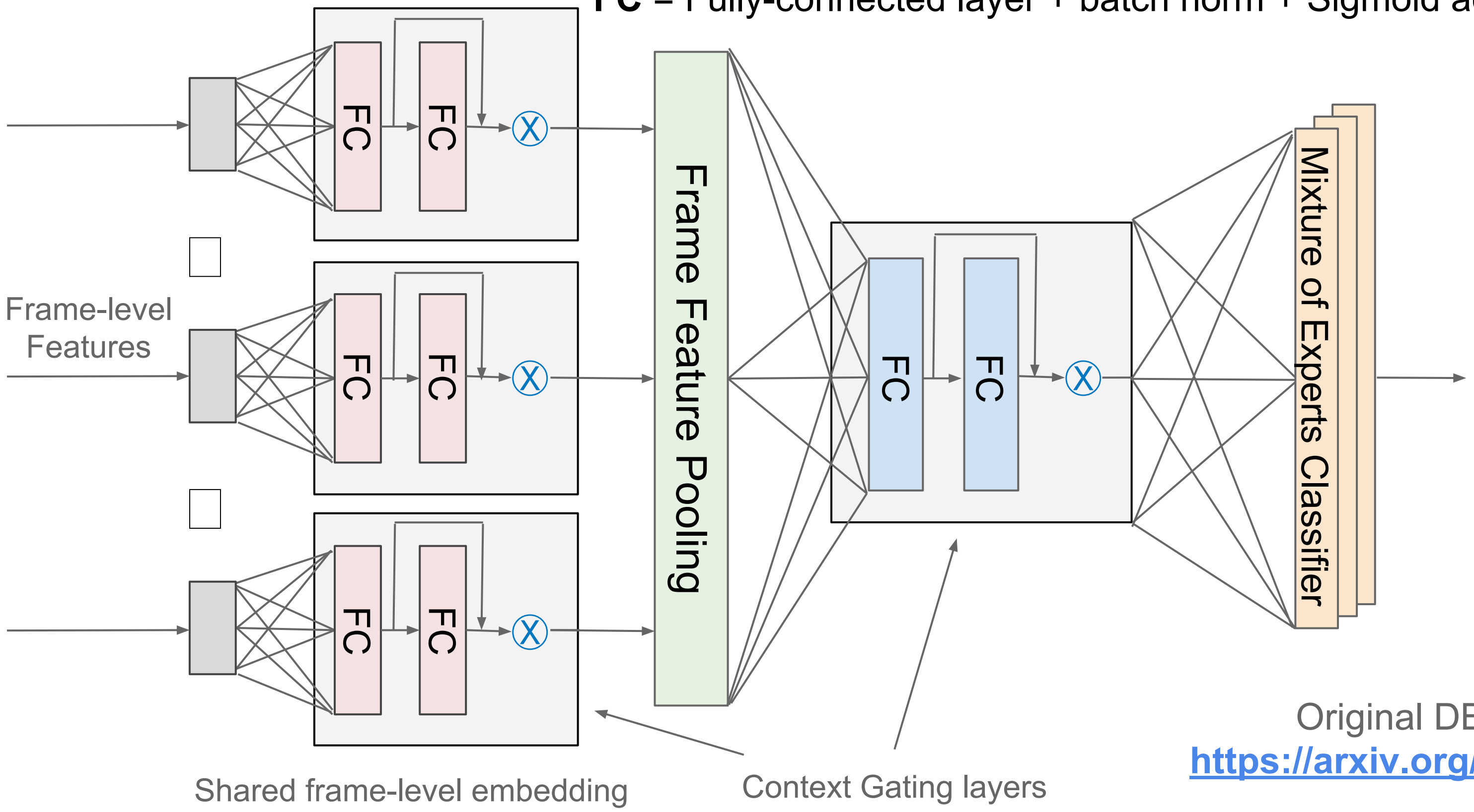
FC = Fully-connected layer + batch norm + Sigmoid activation



Original DBoF paper:
https://arxiv.org/abs/1609.08675

Context-Gated Deep Bag of Frames (DBoF)

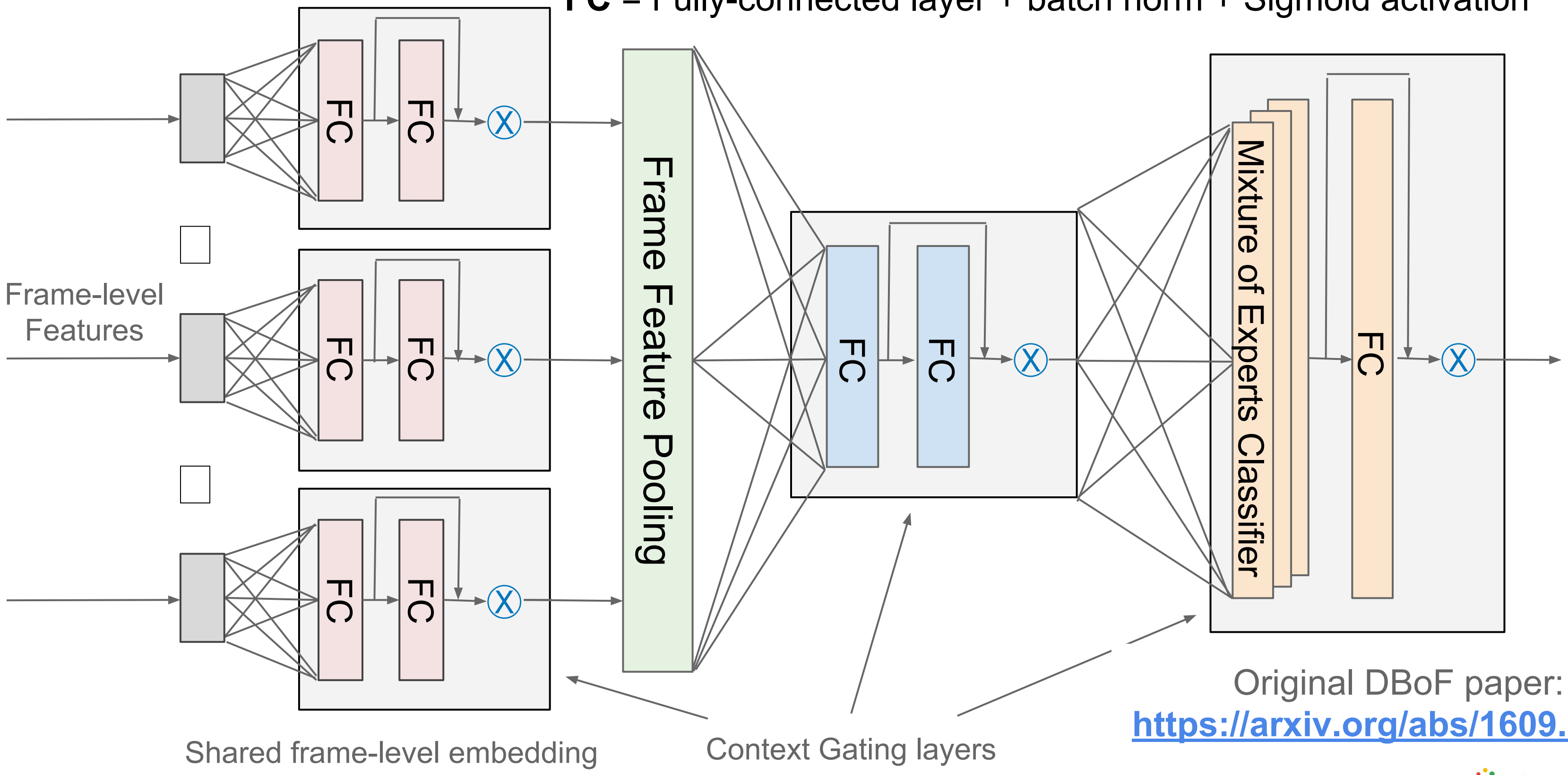
FC = Fully-connected layer + batch norm + Sigmoid activation



Original DBoF paper:
https://arxiv.org/abs/1609.08675

Context-Gated Deep Bag of Frames (DBoF)

FC = Fully-connected layer + batch norm + Sigmoid activation



Original DBoF paper:
https://arxiv.org/abs/1609.08675

The Various Roles of Context Gating

When applied before (temporal) pooling:

- Context gating functions as a frame-specific and feature-specific attention model

When applied after pooling but before classification:

- Context gating adds more capacity / depth to the model
- Resembles ResNet-like skip connections but with multiplicative layer interactions
- But since both branches are required, the ResNet “shortcut” intuition doesn’t apply

When applied after classification:

- Context gating performs fusion across classes, exploits semantic correlations

Recap of feature pooling methods

MAX Pooling

$$f_{Max}(X) = \left[\max_{i=1..N} x_i(j) \right]_{j=1..D}$$

AVG Pooling

$$f_{Avg}(X) = \left[\frac{1}{N} \sum_{i=1}^{N} x_i(j) \right]_{j=1..D}$$

L2 Pooling

$$f_{L2}(X) = \left[\sqrt{\frac{1}{N} \sum_{i=1}^{N} x_i(j)^2} \right]_{j=1..D}$$

SWAP (Self-Weighted Avg Pooling)

$$f_{Swap}(X) = \left[\sum_{i=1}^{N} w_{ij}\, x_i(j) \right]_{j=1..D}, \text{ with } w_{ij} = \frac{|x_i(j)|}{\sum_{i=1}^{N} |x_i(j)|}$$

Attention-Weighted Avg Pooling

$$f_{AttnAvg}(X) = \left[\sum_{i=1}^{N} w_i\, x_i(j) \right]_{j=1..D}, \text{ with } w_i = Softmax(W[x_i]) = \frac{e^{W[x_i]}}{\sum_{i=1}^{N} e^{W[x_i]}}$$

Context-Gated Weighted Avg Pooling

$$f_{CgateAvg}(X) = \left[\frac{1}{N} \sum_{i=1}^{N} w_{ij}\, x_i(j) \right]_{j=1..D}, \text{ with } w_{ij} = \sigma(W_j[x_i]) = \frac{e^{W_j[x_i]}}{1+e^{W_j[x_i]}}$$

Experiments

1. Assess effect of adding context gating after various DBoF layers
2. Assess model size vs. performance by varying bottleneck size & MoE mixtures

Experiments

1. Assess effect of adding context gating after various DBoF layers
2. Assess model size vs. performance by varying bottleneck size & MoE mixtures

Metrics (computed on 10% of the validation set):

- ***Global Average Precision*** *(gAP)* - AUC of global P-R curve across all classes

$$gAP = \frac{1}{|E|N} \sum_{(e,i)=1}^{|E|N} P_e(i) \, \Delta R_e(i)$$

Dominated by frequent classes

Experiments

1. Assess effect of adding context gating after various DBoF layers
2. Assess model size vs. performance by varying bottleneck size & MoE mixtures

Metrics (computed on 10% of the validation set):

- ***Global Average Precision*** *(gAP)* - AUC of global P-R curve across all classes
- ***Mean Average Precision*** *(mAP)* - Mean per-class AUC of P-R curves

$$gAP = \frac{1}{|E|N} \sum_{(e,i)=1}^{|E|N} P_e(i)\, \Delta R_e(i)$$

Dominated by frequent classes

$$mAP = \frac{1}{|E|} \sum_{e} AP(e) = \frac{1}{|E|} \sum_{e=1}^{|E|} \sum_{i=1}^{N} P_e(i)\, \Delta R_e(i)$$

Dominated by rare (fine-grained) classes

Default DBoF Parameters

- Frame feature pooling method: **SWAP**
- L2 normalization before and after frame pooling: **On**
- Batch normalization on all fully-connected and context gating layers: **On**
- Fully-connected and context gating layers activation: **Sigmoid**
- Frame embedding size (before pooling): **4096**
- Video embedding size (after pooling): **4096**
- Context gating layers: **On (before pooling), Off (after pooling), On (after classifier)**
- Classification layer: **Mixture-of-Experts (MoE) with 5 mixtures**

Training Parameters

- Optimizer: **Adam (ε = 0.0001)**
- Batch size: **512 examples**
- Learning rate: **0.005 initially, scaled by 0.95 every ~3K steps (1.5M examples)**
- L2 regularization penalty weight: **0.00001 (on all weights)**
- Clip gradient norm above threshold: **1.0**
- Data augmentation: **sample random 30 frames** per video in each training step

Results - Context Gating

Context Gating Configuration	CGate frame embedding	CGate video embedding	CGate classifier	Est. # model parameters **	Model size on disk (GB) **	YT8M-2017		YT8M-2018	
						gAP *	mAP *	gAP *	mAP *
000				42 M	0.2 GB	0.8293	0.4836	0.8710	0.5643
001			✓	64 M	0.2 GB	0.8336	0.4965	0.8728	0.5697
010		✓		46 M	0.2 GB	0.8325	0.4900	0.8734	0.5707
100	✓			59 M	0.2 GB	**0.8348**	**0.5021**	**0.8749**	**0.5780**

Context gating before frame pooling works better than after pooling or after classification!
This is a DBoF differentiator since early context gating is not feasible with NetVLAD or NetFV

* Global Average Precision (gAP) and Mean Average Precision (mAP) computed on 10% of validation data
** Default model params, except for: video embedding size = 2048, classifier = 2-mixture MoE

Results - Context Gating

Context Gating Configuration	CGate frame embedding	CGate video embedding	CGate classifier	Est. # model parameters **	Model size on disk (GB) **	YT8M-2017		YT8M-2018	
						gAP *	mAP *	gAP *	mAP *
000				42 M	0.2 GB	0.8293	0.4836	0.8710	0.5643
001			✓	64 M	0.2 GB	0.8336	0.4965	0.8728	0.5697
010		✓		46 M	0.2 GB	0.8325	0.4900	0.8734	0.5707
100	✓			59 M	0.2 GB	0.8348	0.5021	0.8749	0.5780
011		✓	✓	69 M	0.3 GB	0.8348	0.4988	0.8737	0.5727
110	✓	✓		63 M	0.2 GB	0.8365	0.5017	**0.8758**	**0.5807**
101	✓		✓	81 M	0.3 GB	0.8365	**0.5083**	0.8748	0.5796
111	✓	✓	✓	85 M	0.3 GB	**0.8372**	0.5074	0.8750	0.5800

Context gating at multiple depths works even better!

* Global Average Precision (gAP) and Mean Average Precision (mAP) computed on 10% of validation data
** Default model params, except for: video embedding size = 2048, classifier = 2-mixture MoE

Results - Model Size (with 110 Context Gating)

Model size variant: (bottleneck, classifier)	Bottleneck layer size	# MoE mixtures	Est. # model parameters	Model size on disk (GB)	YT8M-2017		YT8M-2018	
					gAP *	mAP *	gAP *	mAP *
(small, small)	1024	1	39M	0.1 GB	0.8325	0.4852	0.8738	0.5679
(small, medium)		5	70M	0.3 GB	0.8353	0.4938	0.8759	0.5745
(medium, small)	2048	1	58M	0.2 GB	0.8363	0.5030	0.8759	0.5804
(medium, medium)		5	121M	0.5 GB	0.8393	0.5116	0.8779	0.5874
(large, small)	4096	1	103M	0.4 GB	0.8381	0.5149	0.8757	0.5894
(large, medium)		5	229M	0.9 GB	**0.8413**	**0.5226**	**0.8785**	**0.5961**

* Global Average Precision (gAP) and Mean Average Precision (mAP) computed on 10% of validation data

Results - Model Size (with 110 Context Gating)

Model size variant: (bottleneck, classifier)	Bottleneck layer size	# MoE mixtures	Est. # model parameters	Model size on disk (GB)	YT8M-2017		YT8M-2018	
					gAP *	mAP *	gAP *	mAP *
(small, small)	1024	1	39M	0.1 GB	0.8325	0.4852	0.8738	0.5679
(small, medium)		5	70M	0.3 GB	0.8353	0.4938	0.8759	0.5745
(medium, small)	2048	1	58M	0.2 GB	0.8363	0.5030	0.8759	0.5804
(medium, medium)		5	121M	0.5 GB	0.8393	0.5116	0.8779	0.5874
(large, small)	4096	1	103M	0.4 GB	0.8381	0.5149	0.8757	0.5894
(large, medium)		5	229M	0.9 GB	**0.8413**	**0.5226**	**0.8785**	**0.5961**
2017 Kaggle 1st Place (Team WILLOW)	Best 1 model (NetVLAD)		350M	> 1 GB	0.8320	-	-	-
	Best ensemble		?	> 25 GB?	**0.8497**	-	-	-

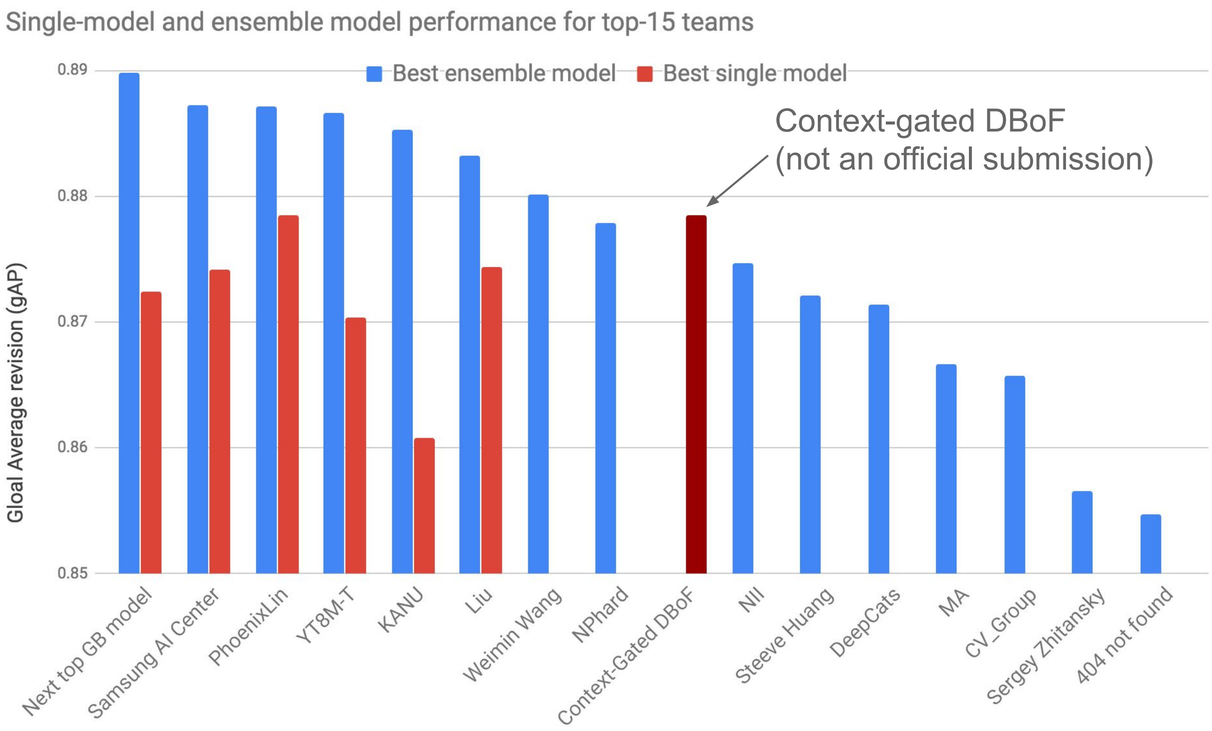
* Global Average Precision (gAP) and Mean Average Precision (mAP) computed on 10% of validation data

Results - Model Size (with 110 Context Gating)

Model size variant: (bottleneck, classifier)	Bottleneck layer size	# MoE mixtures	Est. # model parameters	Model size on disk (GB)	YT8M-2017 gAP *	YT8M-2017 mAP *	YT8M-2018 gAP *	YT8M-2018 mAP *
(small, small)	1024	1	39M	0.1 GB	0.8325	0.4852	0.8738	0.5679
(small, medium)		5	70M	0.3 GB	0.8353	0.4938	0.8759	0.5745
(medium, small)	2048	1	58M	0.2 GB	0.8363	0.5030	0.8759	0.5804
(medium, medium)		5	121M	0.5 GB	0.8393	0.5116	0.8779	0.5874
(large, small)	4096	1	103M	0.4 GB	0.8381	0.5149	0.8757	0.5894
(large, medium)		5	229M	0.9 GB	**0.8413**	**0.5226**	**0.8785**	**0.5961**
2017 Kaggle 1st Place (Team WILLOW)	Best 1 model (NetVLAD)		350M	> 1 GB	0.8320	-	-	-
	Best ensemble		?	> 25 GB?	**0.8497**	-	-	-
2018 Kaggle 1st Place (Next top GB model)	Best 1 model (NetVLAD)		350M	> 1 GB	-	-	0.8724	-
	Best ensemble		?	1.0 GB	-	-	**0.8898**	0.5964
2018 Kaggle 3rd Place (PhoenixLin)	Best 1 model (NeXtVLAD)		79M	0.3 GB	-	-	**0.8785**	-
	Best ensemble		237M	0.9 GB	-	-	0.8871	**0.5968**

* Global Average Precision (gAP) and Mean Average Precision (mAP) computed on 10% of validation data

Ensemble vs. single model scores for top-15 teams (2018)



kaggle.com/c/youtube8m-2018

400+ teams participating

Top 15 teams shown on left
(after model size verification)

Conclusions

- Context-gated DBoF among top-performing single-model architectures on YT-8M:
  - 2017 gAP: 0.8413 (best previously reported: 0.8320 from NetVLAD)
  - 2018 gAP: 0.8785 (best previously reported: 0.8785 from NeXtVLAD)

Model architectures	**Model capacity (#parameters)**	**2018 gAP**
NetVLAD	350M	0.8724
Small DBoF	70M	0.8759
Medium DBoF	121M	0.8779
Large DBoF	229M	**0.8785**
NeXtVLAD	**79M**	**0.8785**

- Ensembling adds just ~0.01 gAP on top of the best individual models
  - **Can we get the ensemble performance with a single model?**

Thank you for your attention.