Imporved Deep Metric Learning with Multi-class N-pair Loss Objective

outline

1. Preliminary: Distance Metric Learning
2. Deep Metric Learning with Multiple Negative Examples
- Learning to identify from multiple negative examples
- N-pair loss for efficient deep metric learning
3. Experimental Results

Preliminary: Distance Metric Learning

- contrastive loss

$$\mathcal{L}^{m}_{\text{cont}}(x_i, x_j; f) = \mathbf{1}\{y_i = y_j\}\|f_i - f_j\|_2^2 + \mathbf{1}\{y_i \neq y_j\} \max\left(0, m - \|f_i - f_j\|_2\right)^2$$

Preliminary: Distance Metric Learning

- triplet loss

$$\mathcal{L}^{m}_{\text{cont}}(x_i, x_j; f) = \mathbf{1}\{y_i = y_j\}\|f_i - f_j\|_2^2 + \mathbf{1}\{y_i \neq y_j\} \max\left(0, m - \|f_i - f_j\|_2\right)^2$$

Preliminary: Distance Metric Learning

- \- contrastive loss

- Preliminary: Distance Metric Learning

- \- triplet loss

Deep Metric Learning with Multiple Negative Examples

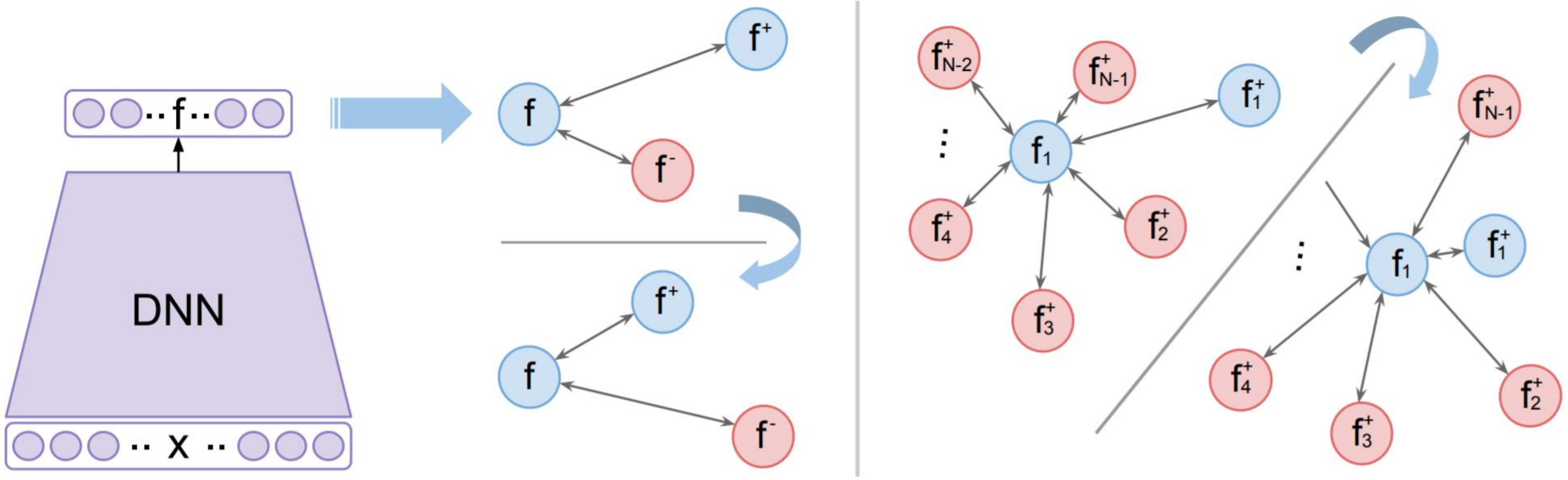
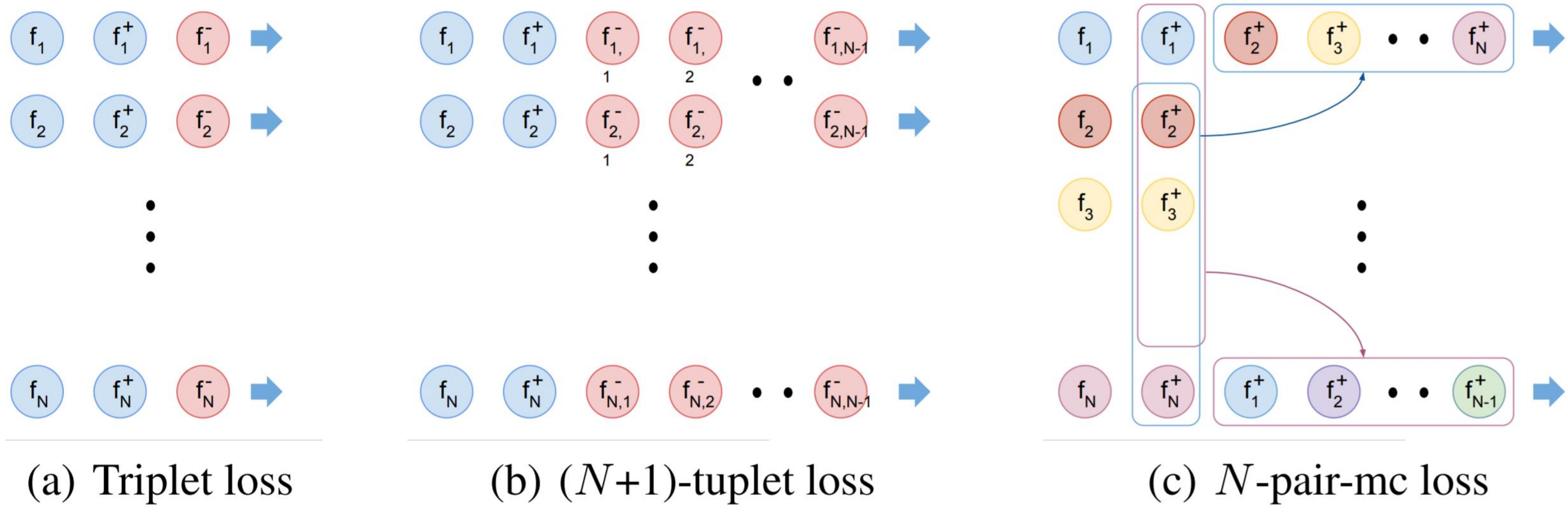


Figure 1: Deep metric learning with (left) triplet loss and (right) $(N+1)$-tuplet loss. Embedding vectors f of deep networks are trained to satisfy the constraints of each loss. Triplet loss pulls positive example while pushing one negative example at a time. On the other hand, $(N+1)$-tuplet loss pushes N-1 negative examples *all at once*, based on their similarity to the input example.

Deep Metric Learning with Multiple Negative Examples



Deep Metric Learning with Multiple Negative Examples
: Learning to identify from multiple negative examples

$$\mathcal{L}(\{x, x^+, \{x_i\}_{i=1}^{N-1}\}; f) = \log\left(1 + \sum_{i=1}^{N-1} \exp(f^\top f_i - f^\top f^+)\right)$$

Deep Metric Learning with Multiple Negative Examples

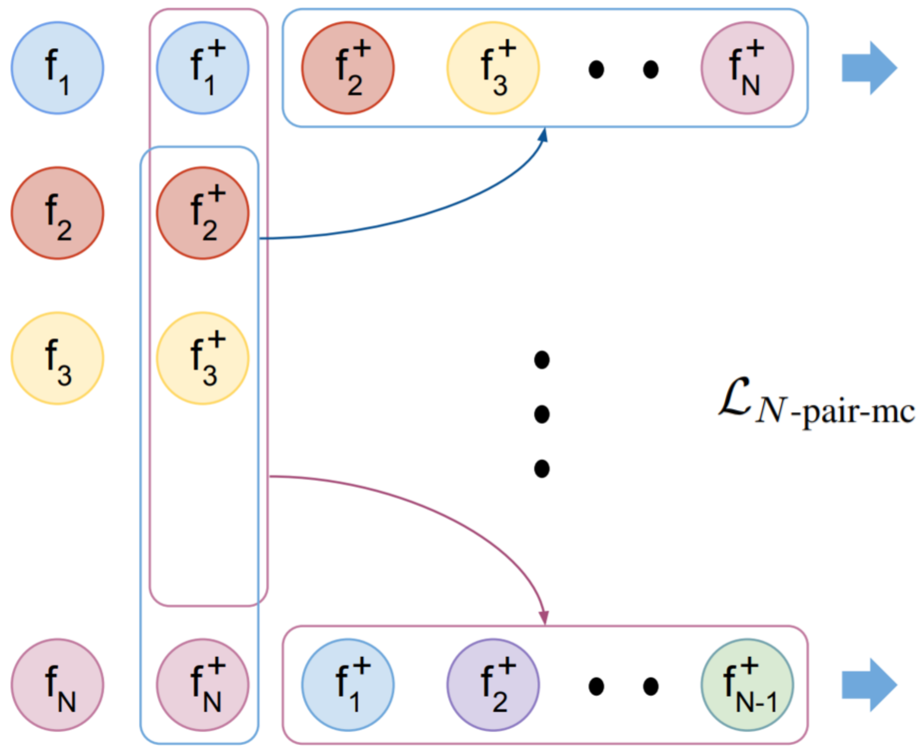
: Learning to identify from multiple negative examples

$$\mathcal{L}_{(2+1)\text{-tuplet}}(\{x, x^+, x_i\}; f) = \log\left(1 + \exp(f^\top f_i - f^\top f^+)\right);$$

$$\mathcal{L}_{\text{triplet}}(\{x, x^+, x_i\}; f) = \max\left(0, f^\top f_i - f^\top f^+\right).$$

$$\log\left(1 + \sum_{i=1}^{L-1} \exp(f^\top f_i - f^\top f^+)\right) = -\log \frac{\exp(f^\top f^+)}{\exp(f^\top f^+) + \sum_{i=1}^{L-1} \exp(f^\top f_i)}$$

Deep Metric Learning with Multiple Negative Examples
: N-pair loss for efficient deep metric learning



(c) N-pair-mc loss

$$\mathcal{L}_{N\text{-pair-mc}}(\{(x_i, x_i^+)\}_{i=1}^N; f) = \frac{1}{N}\sum_{i=1}^{N} \log\left(1 + \sum_{j \neq i} \exp(f_i^\top f_j^+ - f_i^\top f_i^+)\right)$$

Deep Metric Learning with Multiple Negative Examples
: N-pair loss for efficient deep metric learning

Hard Negative class mining

1. **Evaluate Embedding Vectors:** choose randomly a large number of output classes C; for each class, randomly pass a few (one or two) examples to extract their embedding vectors.
2. **Select Negative Classes:** select one class randomly from C classes from step 1. Next, greedily add a new class that violates triplet constraint the most w.r.t. the selected classes till we reach N classes. When a tie appears, we randomly pick one of tied classes [28].
3. **Finalize N-pair:** draw two examples from each selected class from step 2.

Experimental Results
: Fine-grained visual object recognition and verification

Database, evaluation metric		triplet	triplet-nm	72-pair-ovo	72-pair-mc	softmax
Car-333	Recognition	70.24±0.38	83.22±0.09	86.84±0.13	88.37±0.05	**89.21**±0.16 88.69±0.20†
	VRF (neg=1)	96.78±0.04	97.39±0.07	**98.09**±0.07	97.92±0.06	96.19±0.07
	VRF (neg=71)	48.96±0.35	65.14±0.24	73.05±0.25	**76.02**±0.30	55.36±0.30
Flower-610	Recognition	71.55±0.26	82.85±0.22	84.10±0.42	**85.57**±0.25	84.38±0.28 84.59±0.21†
	VRF (neg=1)	98.73±0.03	99.15±0.03	99.32±0.03	**99.50**±0.02	98.72±0.04
	VRF (neg=71)	73.04±0.13	83.13±0.15	87.42±0.18	**88.63**±0.14	78.44±0.33

Table 1: Mean recognition and verification accuracy with standard error on the test set of Car-333 and Flower-610 datasets. The recognition accuracy of all models are evaluated using kNN classifier; for models with softmax classifier, we also evaluate recognition accuracy using softmax classifier (†). The verification accuracy (VRF) is evaluated at different numbers of negative examples.

Experimental Results
: Distance metric learning for unseen object recognition

Online product							
	triplet	triplet-nm	triplet-lifted structure [21]	60-pair-ovo	60-pair-ovo -nm	60-pair-mc	60-pair-mc -nm
F1	19.59	24.27	25.6	23.13	25.31	26.53	**28.19**
NMI	86.11	87.23	87.5	86.98	87.45	87.77	**88.10**
K=1	53.32	62.39	61.8	60.71	63.85	65.25	**67.73**
K=10	72.75	79.69	79.9	78.74	81.22	82.15	**83.76**
K=100	87.66	91.10	91.1	91.03	91.89	92.60	**92.98**
K=1000	96.43	97.25	97.3	97.50	97.51	97.92	**97.81**

	Car-196				CUB-200			
	triplet	triplet-nm	60-pair-ovo	60-pair-mc	triplet	triplet-nm	60-pair-ovo	60-pair-mc
F1	24.73	27.86	**33.52**	**33.55**	21.88	24.37	25.21	**27.24**
NMI	58.25	59.94	**63.87**	**63.95**	55.83	57.87	58.55	**60.39**
K=1	53.84	61.62	69.52	**71.12**	43.30	46.47	48.73	**50.96**
K=2	66.02	73.48	78.76	**79.74**	55.84	58.58	60.48	**63.34**
K=4	75.91	81.88	85.80	**86.48**	67.30	71.03	72.08	**74.29**
K=8	84.18	87.81	90.94	**91.60**	77.48	80.17	81.62	**83.22**

Table 2. F1, NMI, and recall@K scores on the test set of online product [21], Car-196 [12], and CUB-200 [25] datasets. F1 and NMI scores are averaged over 10 different random seeds for kmeans clustering but standard errors are omitted due to space limit. The best performing model and those with overlapping standard errors are bold-faced.

Experimental Results
: Face verification and identification

	triplet	triplet-nm	192-pair-ovo	192-pair-mc	320-pair-mc
VRF	95.88±0.30	96.68±0.30	96.92±0.24	**98.27**±0.19	**98.33**±0.17
Rank-1	55.14	60.93	66.21	88.58	**90.17**
DIR@FIR=1%	25.96	34.60	34.14	66.51	**71.76**

Table 3: Mean verification accuracy (VRF) with standard error, rank-1 accuracy of closed set identification and DIR@FAR=1% rate of open-set identification [1] on LFW dataset. The number of examples per batch is fixed to 384 for all models except for 320-pair-mc model.