Comparison of Loss Functions on Face Embedding

Yu Sun
ys3225@nyu.edu

Hetian Bai
hb1500@nyu.edu

Jieyu Wang
jw4937@nyu.edu

1 Introduction

Improving architecture and loss function are main approaches to enhance the performance of a face embedding learning model. There is a rich line of works using state-of-art DNN architectures to build deep face embedding representation layers, so it is hard to improve performance on a well-defined model or create a new one by limited time. While we realize that loss functions plays the same weighted roles in face embedding. Therefore, the goal of this project is to compare model performances with different loss functions on face verification and recognition tasks. We conducted five experiments by applying Center loss as baseline model, softmax loss, triplet loss, margin-based loss, and our innovative loss function-Hard Margin Based Loss. In this study, we focus on compare loss functions on face recognition task rather than verification, because there are sufficient research papers are evaluating model's verification rate but not recognition, so there is a lot space to explore.

2 Literature Review

Transforming images into rich and semantic representations lies at the heart of face recognition. An uniform system for face learning task is composed of two stages[4]: learning an Euclidean embedding per image using DNN and choosing appropriate models corresponding to face verification, recognition, and clustering. Deep metric learning has made big progress in recent years due to two main reasons: the representation architecture and the loss function.

2.1 Architecture

Deep Face Networks[3] proposed to use deep CNN followed by three locally-connected layers and two fully-connected layers in the field of face recognition. Later in 2015's CVPR, Facenet[4] was introduced and became the state-of-art model in Face embedding learning. Followed by this model, researchers applied new architectures in most recent publications – ResNet[2] and Inception[6].

Resnet[2] is an architecture that could significantly deepen the DNN layers to enhance the model performance without triggering "degradation", which is an issue in "plain" deep models when model gets deeper, the train loss goes up unexpectedly[2]. With Resnet, a DNN architecture can go deeper safely. Therefore, we chose Resnet as the base of our DNN model in this study.

2.2 Loss Function

As far as loss function is concerned, Softmax loss, given by equation 1, is a straightforward method in face recognition as a N-ways classification problem[3] to separate image embedding by classes.

$$L^{Softmax} = -\sum_{i=1}^{m} log \frac{e^{W_{y_i}^T u_i} + b_{y_i}}{\sum_{j=1}^{n} e^{w_j^T u_i + b_j}}\text{[1]} \tag{1}$$

[1] where u_i is the output of the classification layer. W and b is the weight and bias

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Differ from softmax loss, the center loss 2 minimizes the intra-class variations and keep the embedding of different classes separately [7].

$$L_{center} = \frac{1}{2}\sum_{i=1}^{m} ||x_i - c_{y_i}||_2^2 \quad (2)$$ [2]

Another idea for deep distance metric learning is to pull images embedding of the same identity closer while pushing the embeddings of different identities apart. The contrastive loss 3 directly optimizes the distance by encouraging all positive distances to approach zero, while keeping negative distances above a certain threshold[8].

$$L_{ij}^{contrastive} = y_{ij}D_{ij}^2 + (1 - y_{ij})[\alpha - D_{ij}]_+^2 \quad (3)$$ [3]

Compared with contrastive loss, triplet loss [4]4 ensures that negative pairs have larger embedding distance compared with positive pairs, which outperforms many prior loss functions in face verification tasks. However, triplet loss merely try to keep all positives closer to each negatives in each example, which shows in 4.

$$L^{triplet} = [D_{ap}^2 - D_{an}^2 + \alpha] \quad (4)$$ [4]

However, both triplet loss and contrastive loss have drawbacks on slow convergence and they often require expensive data sampling method to provide nontrivial pairs or triplets to accelerate training process[5], such as hard negative mining and semi-hard negative mining[4]. Based on this, [8] provides a margin-based loss 5 which could enjoy the flexibility of triplet loss with more efficient complexity and outperforms other loss for face verification.

$$L_{i,j}^{margin} := [\alpha + y_{ij}(D_{ij} - \beta)]_+ \quad (5)$$ [5]

This paper also claimed that sampling methods are more important than loss functions to improve performance and proposed a distance-weighted sampling method which corrects the sample bias while controlling variance. On the other hand, [5] mentioned a N-pairs loss to address the slow convergence problem of triplet loss. With an efficient mini-batch construction method, the N-pairs loss could update N-1 negative examples in each triplet. In this project, we explored into distance-weighted margin-based loss to solve face recognition problem, compared with softmax loss, center loss, and triplet loss.

3 Framework

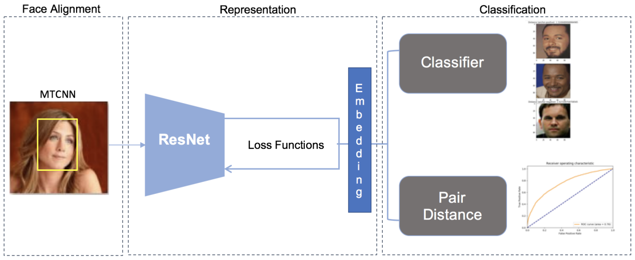

Figure 1: Pipeline

3.1 Alignment Layer with MTCNN

For images in training and testing set, we conducted face alignment to detect and crop face by applying the pre-trained CNN model named MTCNN[9]. This framework leverages a cascaded architecture with three stages of deep convolutional networks to predict face in a coarse-to-fine manner.

[2] $C_{y_i} \in R^d$: the y_ith class center of deep features

[3] $y_{ij} \in \{0, 1\}$, D_{ij} is distance

[4] a: anchor; p: positive example; n: negative example; α: threshold

[5] α: margin; β: is distance boundary

3.2 Representation Layer

Images data were transformed before entering into the deep network. First, all images were resized into 96*96 in order to have a unified size. Next, images were horizontal-flipped to increase data robustness. Then images were turned into tensor and normalized. The DNN architecture is derived from Resnet 18, where the *avepool* layer was dropped and followed by two fully-connected layers.

In this layer, we utilized 5 different loss functions to train the embedding:
1. *Center Loss (baseline model)*[7]
2. *Softmax*[3]
3. *Triplet Loss*[4] We use l_2 norm triplet loss here and random sampling method to generate triplets
4. *Margin-based Loss* [8] shows that margin-based loss outperforms triplet loss since it makes improvements in sampling methods. Hard/semi-hard sampling focus on the negative samples closed to anchors and would generate high variance, while random-sampling would select negative samples according to its distribution[8], which are often too far apart to reduce loss. The Distance-Weighted Sampling uniformly samples data according to distance, so it combine the advantages of above two methods and cancel out the disadvantages.
5. *Hard-Margin-Based Loss (Innovative Loss Function)* Based on margin-based loss, we proposed a new margin-based loss which gives different weight of penalty corresponding to different distances. As 2 shows, in margin loss, the relationship between distance and loss is linear. However, it's easy to misidentify the closed negative pairs, which would generate False Positive and False Negative items. At the meantime, when a pair of images is apart and misidentified, it should be given a high loss but not incredibly high, which may compromised the power of the weights of closed pairs in this batch, since we sum them up in one batch. In this way, the new margin loss6 we proposed, as showed in 3, gives higher penalty to misidentified closed pairs and lower penalty to pairs that are far away than margin, which may contribute to separate face images better.

$$L_{i,j}^{hard\_margin} := log([\alpha + y_{ij}(D_{ij} - \beta)]_{+} + 1)^{6} \tag{6}$$

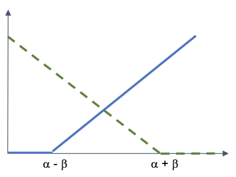

Figure 2: Margin Loss: Green dashed line for negative pairs, blue for positive pairs

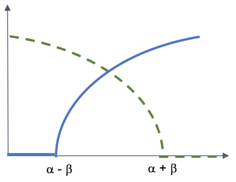

Figure 3: New Margin Loss: Green dashed line for negative pairs, blue for positive pairs

3.3 Classification Layer

Having the embedding, we conducted two tasks–face recognition and verification:
Face recognition is a muti-classification task. We forwarded embedding into a linear classifier to build the mapping from embeddings to classes. Then a layer of Softmax was applied to obtain predictions.
Face verification is a binary classification task. By calculating l_2 distance of embeddings of a pair of exemplars, we set a threshold to make the identity judgment. When the distance is less than the threshold, the pair was classified as the same identity, otherwise, prediction will be two different identities.

4 Experiment

4.1 Datasets

In this study, we obtained training and testing set from the 140 GB cropped face images dataset from MS-Celeb-1M developed by Microsoft [1]. Due to the time constraints, we chose to use a

[6] α: margin; β: is distance boundary

subset of the cropped face images dataset instead of the whole set to reduce training time. There are 6400 identities for both training set and face recognition testing set. In training set, each identity contains 68 images, and 2 images per person in face recognition testing set. To test face verification performance, we have LFW dataset, which is commonly used in relative research projects.

4.2 Evaluation

We chose to use accuracy and AUC to evaluate models with different loss functions. For face recognition task, we forward the output embeddings to a multi-class classifier and then calculate the accuracy by comparing the output with true labels. For the face verification task, we computed the distance between a pair of images, and compared it to a threshold to determine whether this pair is from the same person. For different threshold, we can get a pair of False Positive Rate and True Positive Rate. By this, we calculated AUC metric and use it to evaluate the face verification task comprehensively. Inspired by this, we chose the best accuracy among all thresholds as the metric.

4.3 Results and Analysis

4.3.1 Baseline

Center loss generates centers according to classes, then minimizes the distance of images to centers, which is a straightforward approach to separate image embeddings. We chose it as the baseline and got the verification accuracy of 61% and recognition accuracy of 35%. The poor performance might result from the fact that it only consider the distance between images to center, but not the distances among centers, which not fit the situation of classification from a very large number of classes. Therefore, we attempted to use other loss functions to improve the center loss model.

4.3.2 Experiments on Convergence Speed

As showed in Figure 4, we could find softmax loss converges very fast and gets the lowest loss. Although triplet loss converges faster than baseline, it fluctuated wildly because we use random sampling and the induced loss depends on the triplets in each batch.

We didn't show the loss of margin-based loss and hard margin-based loss in the figure because we don't have enough time to tune the parameter and wait for the model to converge. We spend a lot time on implementing the distance-weighted sampling from [8], where we constructed a batch with 1600 identities and 4 images per identity consecutively. In each batch, since we know that consecutive 4 images are from the same class, we assigned weights of the same class with the anchor as zero and give weights of images in other classes by their distances to the anchor. Additionally, since we changed dataset and set different batch construction parameters from original code, so the optimal learning rate doesn't fit for the current margin-based model and hard Margin-based model. In future work, we would continue to tune the parameter and get results.

4.3.3 Experiments on Verification Task

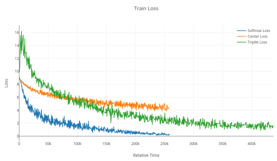

Figure 4: Train Loss

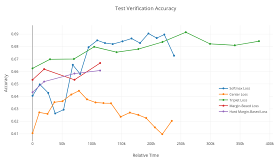

Figure 5: Test Verification Accuracy

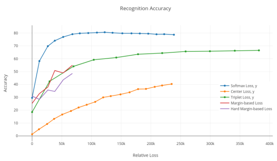

Figure 6: Test Recognition Accuracy

The accuracy rates of our models could not match with the state-of-art model because of the small datasets and limited training time we applied. According to Figure 5, we find that for verification task, triplet loss has a relatively better performance than others. Even compared with softmax loss, triplet loss is more stable because it ensures negative pairs has a larger embedding distance than positive pairs, which fits verification tasks. What's more, even though we do not obtained perfect parameters for margin-based loss, we could find both margin-based loss and hard-margin-based loss outperform baseline, we may expect to have good performances after parameter-tuning.

4.3.4 Experiments on Recognition Task

According to Figure 6, we find that softmax loss has the best performance over all in terms of accuracy and speed. Triplet loss gives a relatively good result on verification and a poor performance on recognition task. This is because it only take the distances into account, which could merely separate images embedding, however, it could not contribute to identify one image from 6400 classes. On the other hand, in softmax loss, we forwarded the embedding to a multi-classifier, which is fairly efficient for face recognition. Additionally, we find that imperfect margin-based loss and new margin-based loss yield close performance with triplet loss. Triplet loss ensures that negative pairs has a larger embedding distance compared with positive pairs, which outperforms many prior loss functions in face verification tasks. However, in practice, the convergence speed of triplet loss is very low due to the inefficient sampling method in hard-negative/positive pairs mining. In addition, embedding collapses is a big issue, which means images has the same embedding which because the hard sampling method of triplet loss[8]. To prevent embedding collapse, we randomly sampled data in the process of triplets.

5 Conclusion

From our experiments and analysis, we conclude that softmax loss yields good results on both face verification and recognition tasks. Even though triplet loss takes more time to converge on loss, it gives good performance on verification task. Constrained by time, we did not complete training and tuning two margin-based function models, however, based on their initial performances and trends, we believe it could outperform other loss functions models after parameter tuning.

References

[1] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision*, pages 87–102. Springer, 2016.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[3] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. *Procedings of the British Machine Vision Conference 2015*, 2015.

[4] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[5] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems*, pages 1857–1865, 2016.

[6] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, volume 4, page 12, 2017.

[7] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515, 2016.

[8] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krähenbühl. Sampling matters in deep embedding learning. *arXiv preprint arXiv:1706.07567*, 2017.

[9] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.