The Investigation on Cosine Derived Loss Functions for Classification on MNIST Dataset

Submission Year: 2021

Group: The Curiosity

Abstract

Deep Convolutional Neural Networks is one of the most popular deep learning approaches begin widely used in the various fields such as image recognition, object detection, face recognition tasks and biomedical imaging analysis. There are some trade-offs between the computational workload with the power consumption and applications reliability. CNNs have significant in terms of degrading computational workload. In ArcFace [1], CosFace [2] and SphereFace [3] applications, the different loss configurations based on cosine function are examined. In our work, we focus on the consistency of the described loss functions for the generalization of the classification tasks. With respect to our findings, ArcFace loss function either with combining SphereFace and CosFace or without combining them provides reliable results.

1. Introduction

Deep Convolutional Neural Networks (DCNNs) is one of the deep learning approaches utilized to perform various artificial intelligence and learning tasks. The appropriate design of CNNs in terms of architectural configuration and hyper-parameters is significant to perform these tasks with high performance and lower computational workload. In addition to the beneficial sides of the CNNs, there are also challenging points such as model selection, determination of the activation function for the perceptrons and loss function. Day by day, experimental tasks are closer to the real life aspects and CNNs gets more complex to realize real life patterns. In this manner, many studies are conducted among the CNNs sub-topics such as developing more sophisticated loss function configurations, optimizing hierarchical parameters. In our base paper ArcFace, the softmax that is one of the commonly used loss functions is developed to enhance the discriminative power of features. Face recognition task, is more complicated than the classification task, since there are too much samples to be recognized. Thus, before recognizing the people's faces, the dataset is classified with respect to some metrics. After classification among the faces, they tried to recognize faces for each class. For this task, the softmax function does not provide certain results especially for the boundaries of the classification points as shown in the figure 1.

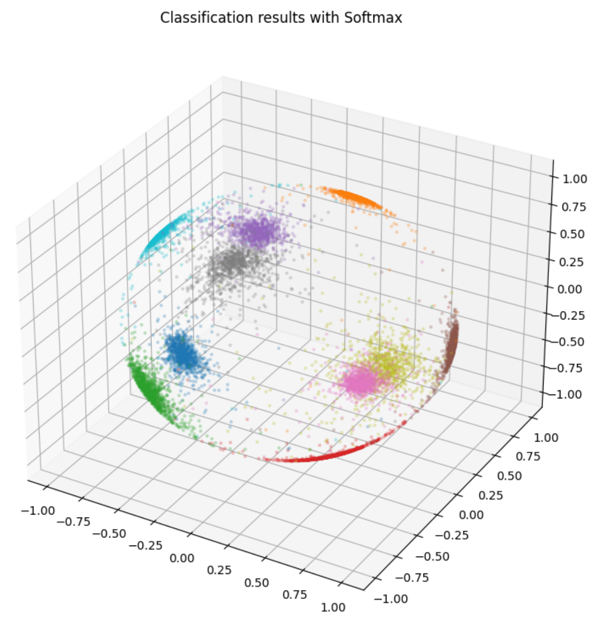

Figure 1: Classification with MNIST dataset and Softmax function projected onto 3D surface

As shown in figure 1, boundaries for classifications observed with softmax function is scattered on to 3D plane. As mentioned previously, boundaries are not certain in terms of determining class of a digit. Similar problem occurs for the classification among the faces. Also, since there are lots of image samples taken from lots of persons for LFW [4], CFP-FP [5] and AgeDB-30 [6] datasets, there are more overlapping boundaries with respect to MNIST dataset. Furthermore, the same problem exists for the intra similarity measure. To solve this unclear approach, ArcFace method which is explained in detailed in the approach section.

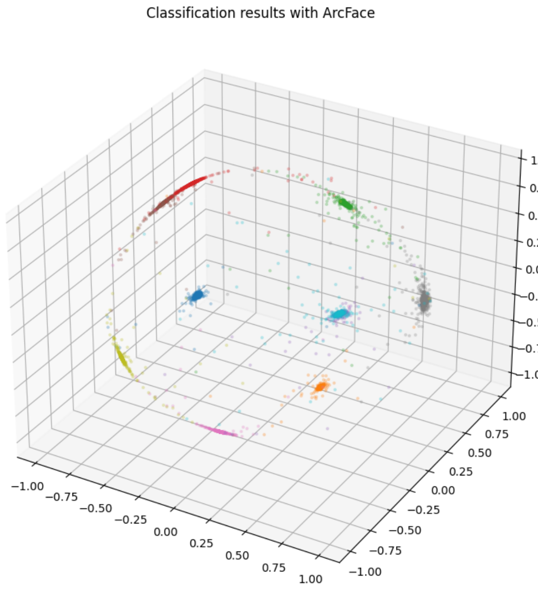

Figure 2: Classification with MNIST dataset and ArcFace method projected onto 3D surface

In figure 2, it can be seen that ArcFace method provides more certain restrictions among the classes which is beneficial for the cases where there are lots of classification clusters. In our work, we try to improve ArcFace results by combining CosFace and SphereFace approaches which will be detailed in the approach part, and generalize both ArcFace and unified approach to another classification task tried with MNIST dataset instead of face recognition. With respect to our findings, combined loss function provides a bit better accuracy compared to the ArcFace for classification of the hand written digits task and, both combined and ArcFace loss function approaches provide more accurate performance for the classification with MNIST dataset with respect to the traditional softmax and cross-entropy loss. Results belongs to our findings are shared in the experiments section.

In recent studies [2, 3], several approaches that utilize margins are proposed to enhance loss functions of DCNNs by maximizing the inter-class distances. In our work, we implement several loss functions by utilizing three different margin types such as multiplicative angular margin, additive angular margin and cosine margin with different values on MNIST dataset [7] to observe the performance of different combinations of these margin types.

2. Related Work

Recently, many studies have been performed to design appropriate loss functions of DCNNs using margin based approaches. For instance, ArcFace [1] proposes a loss function that utilizes additive angular margin to increase the intra-class compactness and inter-class discrepancy and stabilise the training process for face recognition task. In the ArcFace study, it is stated that softmax loss function, most widely used classification loss function, produces ambiguous decision boundaries and does not optimise the features to provide higher similarity for intra-class samples and diversity for inter-class samples. For this reason, a performance gap occurs under large intra-class appearance variations and large-scale test scenarios. ArcFace provides a new approach to enhance traditional softmax loss function. ArcFace calculates the angle between the features and the target weights using arc-cosine function and adds an additive angular margin penalty to the target angle. Thereafter, the target logit is obtained back again using cosine function. Then, all logits are re-scaled by a fixed feature norm. ArcFace enforces more evident gaps than traditional softmax between classes using this method. The results of this approach are presented by performing comprehensive experiments on ten face recognition benchmarks. Besides, several face verification datasets such as LFW, CFP-FP and AgeDB-30 are utilized to check the improvement during training. Furthermore, the CNN architectures ResNet50 and ResNet100 are utilized for experiments. As a result of experiments, it is stated that ArcFace beats the baselines and provides high verification performance. This result shows that the additive angular margin penalty can enhance the discriminative power of features.

CosFace [2] is another study that tries to maximize inter-class variance and minimize intra-class variance by proposing a new approach. CosFace reformulates the softmax loss by normalizing features and weight vectors based on a cosine margin term. CosFace proposes a dubbed Large Margin Cosine Loss (LMCL) algorithm. This algorithm takes the normalized features as input and maximizes the inter-class cosine margin to learn highly discriminative features. The results are presented on popular face datasets such as LFW to demonstrate the effect of proposed algorithm. According to the results, it is stated that CosFace provides high accuracy results for learning highly discriminative features.

Another study SphereFace [3] also focus on enhancing the softmax loss function by proposing angular softmax loss to learn angularly discriminative features. The features learned by proposed loss funtion are angularly distributed. SphereFace loss function generates more seperated decision boundaries by enlarging the inter-class margin and compressing the intra-class angular distribution. Several experiments are performed on various datasets to demonstrate the effect of the angular margin and it is stated that SphereFace loss function provides significant improvement and obtains higher results than traditional softmax loss function.

3. Approach

The problems mentioned in the introduction part tends to the usage of cosine derived and enhanced loss functions. First of all, traditional softmax function as shown in below equation, includes

$$W_j^T * x_i^n = \|\mathbf{W_j}\| * \|\mathbf{x_i}\| * \cos(\theta_j)$$

term. This term is the dot product of j'th weight vector

$$L_1 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T x_i + b_j}},$$

and i'th sample. For the simplicity, by ignoring the bias term, equation turns into below format where s is the product component and cosine is the angular component. In

$$L_2 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s\cos\theta_{y_i}}}{e^{s\cos\theta_{y_i}} + \sum_{j=1, j\neq y_i}^{n} e^{s\cos\theta_j}}.$$

the base paper, they enhance this angular cosine term by adding some additional methods mentioned in CosFace, SphereFace and ArcFace papers. The updated format for this function is given below where m1 is the SphereFace, m2 is the ArcFace and m3 is the CosFace hyper parameters for combined usage.

$$-\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s(\cos(m_1\theta_{y_i}+m_2)-m_3)}}{e^{s(\cos(m_1\theta_{y_i}+m_2)-m_3)} + \sum_{j=1, j\neq y_i}^{n} e^{s\cos\theta_j}}.$$

In the ArcFace paper, they focused on the m2 parameter (m1=1 and m3=0) and performance metrics are taken for mostly this configuration. We examined two additional approach:

- Questioning that can we generalize ArcFace or combined loss model for classification approaches,
- Examining the combined loss method and ArcFace method classification tasks to find an optimized point from various configurations and comparing performance results of them.

3.1. Generalization of Cosine Derived Loss Functions to Classification Tasks

In this part, we investigate whether either combined loss function or ArcFace loss is used for classification tasks or not. This idea arises from that face recognition tasks include classification or clustering tasks inside. The results are shared in the experiments section.

3.2. Performance Accuracy Comparison for ArcFace and Combined Methods

In this part, we try to build and experiment setup which will detailed in the experiments section to compare the reliability for ArcFace and Combined loss function. Each of these methods disturbs the loss results by adding a margin to these backward losses. In the ArcFace method, there is just one parameter to add as an angular margin. In terms of disturbing, this approach has similar analogy with the weight decay. Combined loss function combines 3 hyper parameters that are coefficient of for angle between input sample and corresponding weight (SphereFace), angular margin (ArcFace) and last disturbing term (CosFace) to control the loss function and observing higher accuracy. An experimental setup is built to try different configurations and hyper parameter values and detailed in the experiments part with corresponding results.

4. Experiments

First of all, to investigate the usability of these cosine derived loss functions on classification tasks, we designed a simple CNN consists of a 4 layer and used the ArcFace loss for the final loss function embedded with the cross entropy. After training for different amounts of epoch, we observe 2 different characteristics. For more number of embedding cases, the configuration that uses ArcFace loss function converges more accurate performance values in a short term duration (i.e 5 and 10 epochs) compared to the traditional softmax approach. For the long run (100 epochs), ArcFace loss reach the accuracy value of 99.08 which is too close to the traditional softmax function's accuracy of 99.11. However, the reason behind this is that the built-in tools of fastai library configures the hyper parameters of the architecture internally for the long run such as adding a momentum to the learning scheme. In contrast, since we implement the ArcFace function manually, the convergence of our modelling is not as successful as the fastai library's traditional softmax for the long run compared. Moreover, building a more sophisticated learning scheme with ArcFace loss by tuning the additional angular margin parameter during the long run is the our curiosity. Nevertheless, we can say that ArcFace loss function can be used for classification tasks as in face recognition according to the our experiment that is trained with MNIST dataset. Secondly, by starting from the obtained in base paper ArcFace, we tried different parameters for both combined loss function and ArcFace loss function. The parameters and results are shared in the below, table 1.

Table 1. Accuracy results for ArcFace loss and Combined loss functions on MNIST dataset

Accuracy Results			
ArcFace Method		Combined Method ArcFace+SphereFace+ CosFace m2+(m1=1)+m3	
m = 0.4	98.84	**m2 = 0.4** **m3 = 0.2**	**98.97**

m = 0.41	98.75	m2 = 0.41
		m3 = 0.2
			95.93

m = 0.42	98.66	m2 = 0.42
		m3 = 0.2
			98.80

m = 0.43	98.77	m2 = 0.43
		m3 = 0.2
			98.89

m = 0.44	98.77	m2 = 0.44
		m3 = 0.2
			98.82

m = 0.45	98.70	m2 = 0.4
		m3 = 0.25
			98.78

m = 0.46	98.85	m2 = 0.41
		m3 = 0.25
			98.69

m = 0.47	98.71	m2 = 0.42
		m3 = 0.25
			98.83

m = 0.48	98.79	m2 = 0.43
		m3 = 0.25
			98.80

m = 0.49	98.66	m2 = 0.44
		m3 = 0.25
			98.88

m = 0.50	**98.93**	m2 = 0.4
		m3 = 0.3
			98.84

m = 0.51	98.84	m2 = 0.41
		m3 = 0.3
			97.73

m = 0.52	98.65	m2 = 0.42
		m3 = 0.3
			98.77

m = 0.53	98.70	m2 = 0.43
		m3 = 0.3
			98.69

m = 0.54	98.87	m2 = 0.44
		m3 = 0.3
			98.81

m = 0.55	98.76	m2 = 0.4
		m3 = 0.35
			98.73

m = 0.56	97.58	m2 = 0.41
		m3 = 0.35
			98.71

m = 0.57	98.86	m2 = 0.42
		m3 = 0.35
			98.71

m = 0.58	98.83	**m2 = 0.43** **m3 = 0.35**	**98.95**

m = 0.59	96.01	m2 = 0.44
		m3 = 0.35
			96.83

For ArcFace configuration, m is tried with 0.01 increments for 20 different iterations with 20 epoch training amount. In this experiment, the maximum accuracy is observed when m = 0.5 different from the ArcFace work that has reach it's maximum accuracy for LWF dataset at 0.4. With this results, we can say that for each task, angular margin selection is tuned with respect to the application since disturbing amount results in different scenarios depending on the applications features and characteristics. Additionally, our main focus is to investigate that whether ArcFace is more successful compared to the combined loss function for other domains. To investigate this, second experiment is built with the configuration parameters given in the right side of the table 1. We observed that combined method result in bit accurate than the ArcFace method. The reason behind this, instead of constantly adding an amount of margin, combined method captures more meaningful resultant cosine loss with the configuration of CosFace and ArcFace parameters. SphereFace parameter which is the coefficient of the angle is hold as 1 during the second experiment. As a result, a user can train its dataset with different configurations with the hyper parameters of the combined loss, and choose the optimized accurate parameters with regression method. To do so, there has to be the similarity between the datasets for user's dataset and our parameters. In a more generalized way, it can be collected that lots hyper parameter data for different training dataset and this can be used for the users to determine optimal hyper parameters used with combined loss with respect to the user's dataset.

5. Conclusion

In this work, we present an approach to investigate the effects of ArcFace loss function and combined loss function for classification tasks such as identifying hand written digits on MNIST dataset as we did. We perform several experiments to compare the performance of the traditional softmax loss and the cosine derived loss functions (both ArcFace and Combined loss functions) and observe the effect of three different margin types that are proposed in ArcFace, SphereFace and CosFace studies. We obtained that these Cosine loss derived methods can result in different performances with respect to the application domain and dataset. We observed this as described in the experiment section and shown table 2. Moreover, further experiments can be performed to observe the performance results of ArcFace loss function and combined loss function on more exhaustive datasets for different tasks. In this way, a repository that includes different combined loss hyper parameters can be created for various datasets and varying deep learning tasks, and users can prefer their hyper parameters with a simple regression methods from these repository.

References

[1] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. pages 4685–4694, 2019.

[2] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu. Cosface: Large margin cosine loss for deep face recognition. pages 5265–5274, 2018.

[3] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. pages 6738–6746, 2017.

[4] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[5] C.D. Castillo V.M. Patel R. Chellappa D.W. Jacobs S. Sengupta, J.C. Cheng. Frontal to profile face verification in the wild. In *IEEE Conference on Applications of Computer Vision*, February 2016.

[6] Stylianos Moschoglou, Athanasios Papaioannou, Christos Sagonas, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Agedb: The first manually collected, in-the-wild age database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[7] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.