Exploring DINO Weights and Triplet Loss for Writer Verification

Manan Shah*
Indian Institute of Technology, Mandi
mt-shah@hotmail.com

Abstract

The task of writer verification from handwritten text is a step ahead from the offline writer independent signature verification task. Given two handwritten texts, this task involves predicting whether the texts were written by the same person. We try to model the problem as a metric learning task and experiment with a ResNet50[1] pretrained with DINO[2] self supervision on ImageNet[3] as the backbone. The code for the experiments can be found at https://github.com/cs-mshah/NCVPRIPG2023_Writer_Verification

1 Introduction

The task of offline handwritten signature verification has been long studied in [4], [5], [6], [7], [8]. The task of writer verification from two handwritten texts is more challenging than signature verification as it can involve texts written in different languages, having special characters (digits, punctuations), different strokes/angles and varying sizes. We try to show results on the dataset from the **NCVPRIPG-2023** challenge on writer verification[2]. This dataset consists of training, validation and test sets. The training set consists of 1352 folders with images in one folder written by the same person. The validation and test set consists of images coming from 92 and 360 different writers respectively. We are required to output confidence scores for pairs of images in the `val.csv` and `test.csv` files. The AUC score is the metric used for evaluation. Our best performing method is able to obtain an AUC of **0.9775** on the validation set.

2 Method

The task of predicting whether two texts are written by the same person is that of binary classification. All images are resized to 224×224 for feeding them to a standard ResNet50[1] architecture. The ResNet50 backbone is chosen to be a DINO[2] pretrained backbone on ImageNet[3], as it is a good nearest neighbour classifier and would be a good candidate for transfer learning and modeling metric learning tasks.

2.1 Model architecture

The final layer of the original ResNet50 model is a *fully connected (fc) layer* with 1000 out-features (for 1000 classes). We fine-tune the ResNet50 model by replacing this *fc layer* with a linear layer having an output dimension of 64. These 64 dimensional features are coupled with a loss function to train the model 1.

We use a **triplet margin loss** to train the network. Given a batch of images, the triplet margin loss is defined as,

*https://cs-mshah.github.io/
[2]https://vl2g.github.io/challenges/wv2023/

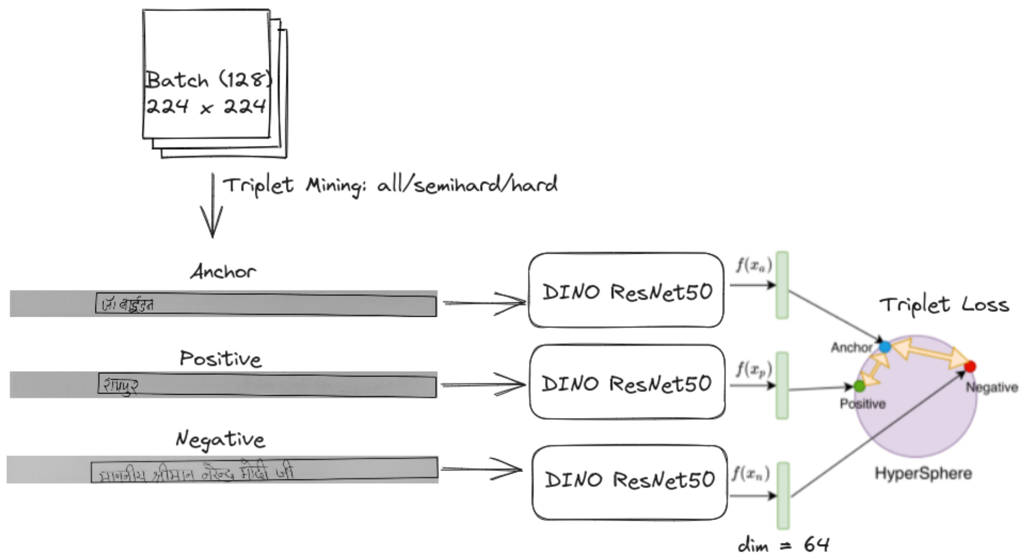

Figure 1: Architecture diagram

$$L(a,p,n) = max\{d(a_i,p_i) - d(a_i,n_i) + margin, 0\} \tag{1}$$

where a denotes the anchor, p denotes the positive (image from the same writer class as anchor) and n denotes the negative (image from a different writer than the anchor). Triplets are mined by a `Miner` function depending on specific conditions [3]. Here d denotes the distance metric, which in our case is chosen to be the cosine similarity between two vectors. The margin is chosen to be 0.2 in all experiments. After mining triplets from the batch, the final loss for that batch is computed by averaging the loss of individual triplets.

2.2 Training strategy and conclusions

Various hyperparameters were varied to see improvements in the AUC score on the validation set and training time till convergence of the AUC. A learning rate of 0.01 was used for the head (fc layer), 0.001 was used for the backbone, along with adam[9] as the optimizer and an exponentially decaying lr scheduler. The lr decays every epoch according to,

$$lr = lr_0(1 + 0.001lr)^{-0.75} \tag{2}$$

The batch sizes were varied in {32, 128} and the batch size of 128 significantly outperformed on the AUC as compared to 32. The models were trained till 80 epochs.

For easy implementation of losses, miners and samplers, the pytorch-metric-learning[10] package was used. It was found that using an `MPerClassSampler` instead of the standard `RandomSampler` for sampling in the `Dataloader` provided extremely fast convergence in about 10 to 15 epochs only. This sampler samples 4 images of every class in a batch of images, ensuring more number of triplets, and thus a faster convergence.

The `TripletMarginMiner`, was used for mining triplets with `all`, `hard`, `semihard` strategies, of which the AUC scores were in the order: all > semihard > hard. In case of hard mining, the loss collapsed (stagnated) in a few epochs leading to no learning.

All the experiments were monitored using weights and biases [4]. The best AUC achieved on the `val` set is **0.9775**, which is obtained with a batch size of 128, `all` mining strategy, `MPerClassSampler`, 0.2 margin, 64 dimensional output vector and training till 10 epochs.

[3] TripletMarginMiner
[4] wandb experiment logs

References

[1] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Computer Vision And Pattern Recognition*, 2015.

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[4] Siladittya Manna, Soumitri Chattopadhyay, Saumik Bhattacharya, and Umapada Pal. Swis: Self-supervised representation learning for writer independent offline signature verification. *arXiv preprint arXiv: 2202.13078*, 2022.

[5] Soumitri Chattopadhyay, Siladittya Manna, Saumik Bhattacharya, and Umapada Pal. Surds: Self-supervised attention-guided reconstruction and dual triplet loss for writer independent offline signature verification. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 1600–1606. IEEE, 2022.

[6] Saeeda Naz, Kiran Bibi, and Riaz Ahmad. DeepSignature: fine-tuned transfer learning based signature verification system. *Multimed. Tools Appl.*, 81(26):38113–38122, November 2022.

[7] Li Liu, Linlin Huang, Fei Yin, and Youbin Chen. Offline signature verification using a region based deep metric learning network. *Pattern Recognit.*, 118:108009, October 2021.

[8] Sounak Dey, Anjan Dutta, J Ignacio Toledo, Suman K Ghosh, Josep Lladós, and Umapada Pal. Signet: Convolutional siamese network for writer independent offline signature verification. *arXiv preprint arXiv:1707.02131*, 2017.

[9] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv: 1412.6980*, 2014.

[10] Kevin Musgrave, Serge J. Belongie, and Ser-Nam Lim. Pytorch metric learning. *ArXiv*, abs/2008.09164, 2020.