Investigating the Impact of Architectural Variations on the Performance of Siamese Neural Networks

Sharath Kumar Reddy Alijarla
Recent Advancements In Machine Learning
Mechatronics
1588268
Universität Siegen

Abstract

Siamese Neural Networks (SNNs) have gained popularity due to their ability to compare and identify similarity between input data points. In this study, we investigate the impact of architecture details on the performance of Siamese Neural Networks. We explore various parameters that may differ in SNNs, such as the choice of neural network architectures (e.g., CNNs, MLPs), number of layers, neurons per layer, size of the output layer (i.e., final embedding), and hyperparameters (e.g., learning rate, weight initialization scheme). To support our findings, we also look into relevant literature. The results of this study will provide valuable insights into designing efficient and effective Siamese Neural Networks for similarity-based tasks.

1 Introduction:

Siamese Neural Networks (SNNs) are a popular approach for comparing and identifying similarities in data. They can learn complex patterns and excel in similarity-based tasks. In this study, we explore how different architectural details affect SNN performance. We consider factors like neural network architectures (e.g., CNNs, MLPs), layer numbers, neurons per layer, output layer size, learning rate, and weight initialization schemes.

For our investigation, we use the well-known MNIST(3) database of handwritten digits. We randomly select subsets of 100 training images and 50 testing images.

We have two neural network architectures, CNN and MLP, and we assess their performance using the binary cross-entropy loss function. Training is done on 100 images and testing on 50 images, with 10 epochs and a batch size of 1024.

Our performance analysis focuses on several aspects, including the impact of learning rate on accuracy and loss, the relationship between layer numbers and accuracies/losses, and the comparison of weight initialization methods (Glorot Uniform and He Uniform) for optimal results. We also compare MLP and CNN architectures to understand their strengths. Furthermore, we examine training versus validation accuracies to assess model generalization.

In conclusion, this study aims to uncover insights into SNNs and their design choices for similarity-based tasks. By understanding SNN intricacies, we hope to advance machine learning and similarity-based applications.

2 Methodology:

2.1 Selecting Architectural Parameters:

- Different neural network architectures (CNN, MLP) are considered for SNN implementation.

- The number of layers and neurons per layer are varied to create diverse architectural configurations.

- Other relevant hyperparameters, including learning rate and weight initialization, are identified.

2.2 Data Set:

The MNIST(3) dataset is a well-known and widely used dataset in the field of machine learning and computer vision. It was created by modifying a larger dataset originally collected by the National Institute of Standards and Technology (NIST) in the United States.

The MNIST(3) dataset consists of a large collection of grayscale images of handwritten digits from 0 to 9. Each image is a 28x28 pixel square, representing a digit written by various individuals. The dataset is divided into two main sets: a training set with 60,000 images and a testing set with 10,000 images.

The MNIST(3) dataset has become a fundamental benchmark for testing and evaluating image classification algorithms, especially for tasks involving handwritten digit recognition. Its simplicity and easy accessibility have made it a standard dataset for researchers and practitioners to experiment with various machine learning and deep learning models.

Over the years, MNIST(3) has played a crucial role in advancing the field of computer vision and has served as a starting point for many researchers to develop and validate their algorithms. Despite its relatively small size compared to more modern datasets, MNIST continues to be a valuable resource for learning and demonstrating image recognition techniques and has become a cornerstone in the development of new machine learning approaches.

2.3 Neural Network Model:

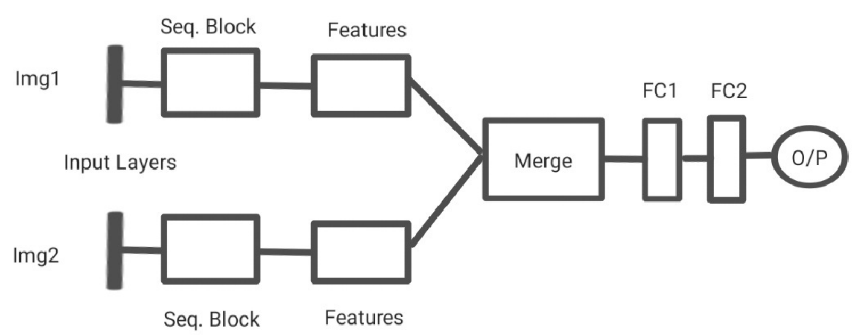

Figure 1: Siamese Neural Network

- Two architectures are used: CNN and MLP.
- The binary cross-entropy loss function is employed.
- Training is performed on 100 images, with testing conducted on 50 images.
- Batch size is set to 1024, and the models are trained for 10 epochs.
- Weight Initialisation: Glorot uniform, He uniform.
- Layers: 2 Input, Sequential Layer, Concatenate Layer, 2 or 3 Fixed Layers, Output Layer
- Metrics: Accuracy, Loss

2.4 Performance Analysis:

- The impact of learning rate on accuracy and loss is evaluated.
- The relationship between the number of layers and accuracies/losses is examined.
- Different weight initialization methods (Glorot Uniform, He Uniform) are compared.
- The performance of MLP and CNN architectures is analyzed based on achieved accuracies and losses.

- Training vs. validation accuracies are compared to understand model generalization.

3 Results and Observations:

All the outcomes achieved in various runs, involving different hyperparameters, architectures, and weight configurations, are recorded and stored in an Excel spreadsheet.

	Siamese Neural Network Using CNN							
No. of Layers	No. of neurons per layer	Size of Output Layer	Learning Rate	Weight Initialisation	Accuracy	loss	val accuracy	val Loss
6	1	1	0.001	Glorot uniform	0.896	0.3106	0.8776	0.3404
	1							
	64							
	128							
	64							
	1							
6	1	1	0.01	Glorot uniform	0.8938	0.3249	0.8832	0.3335
	1							
	64							
	128							
	64							
	1							
6	1	1	0.0001	Glorot uniform	0.8848	0.3218	0.8832	0.4364
	1							
	64							
	128							
	64							
	1							
7	1	1	0.001	Glorot uniform	0.9023	0.2306	0.8896	0
	1							
	64							
	128							
	64							
	128							
	1							

Figure 2: Sample image of results collected in an Excel spreadsheet.

Learning Rate: The ideal learning rate for achieving optimal accuracy is 0.001. Any deviations from this value result in decreased performance. Specifically, at a learning rate of 0.001, the average validation accuracy reaches its peak at 89.70 percent, while the validation loss reaches its lowest of 0.2953. In comparison, using learning rates of 0.01 and 0.0001 yields inferior results

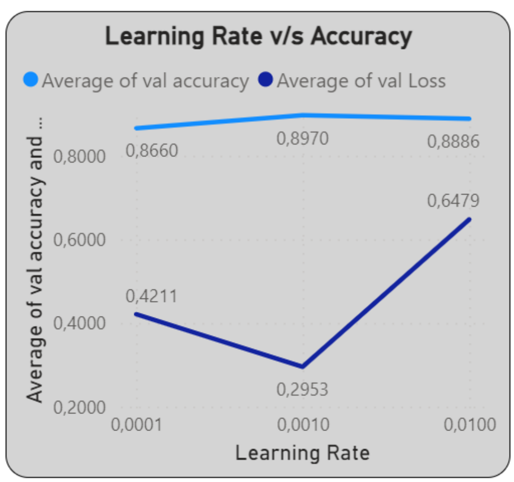

Figure 3: Learning Rate v/s Accuracy

Number of Layers: In my observations, the impact of the number of layers on neural network performance did not exhibit a consistent pattern. The decision regarding the appropriate number of layers to use depends on the complexity of the task being performed and the characteristics of the dataset being used. According to the obtained results, the performance of CNN remained relatively similar when using 6 and 7 layers. MLP, on the other hand, performed slightly better with 7 layers than with 6 layers.

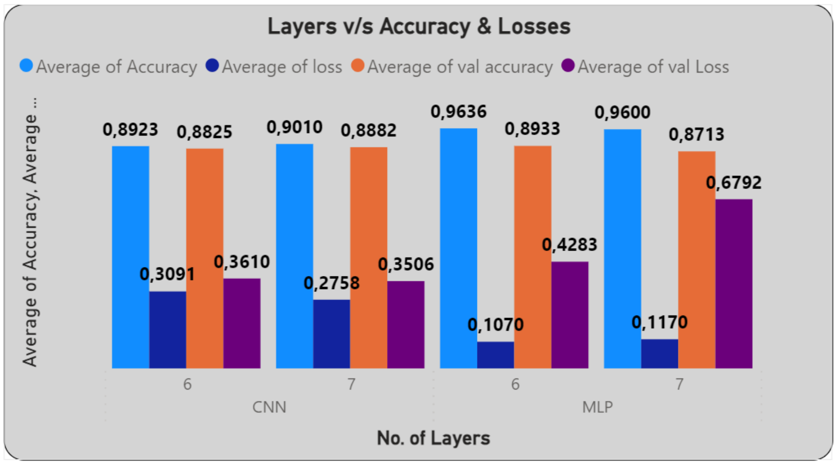

Figure 4: Layers vs Accuracy and Losses

Weight Initialization: HeUniform initialization outperforms GlorotUniform initialization in terms of both accuracy and loss.

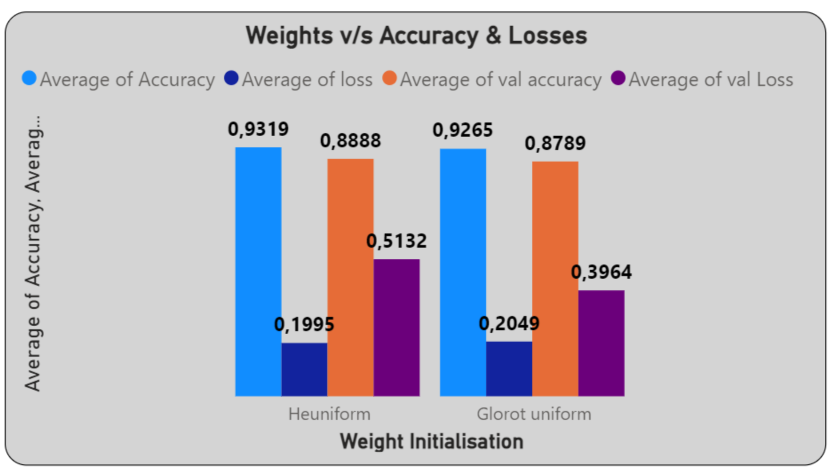

Figure 5: Enter Weights v/s Accuracy and Losses

Architecture: The MLP architecture demonstrates superior accuracy and lower loss compared to CNN on the MNIST dataset. Normally, CNN is expected to perform better for image data, but the unexpected performance of MLP might be because of the small size of the MNIST images, which are only 28x28 pixels.

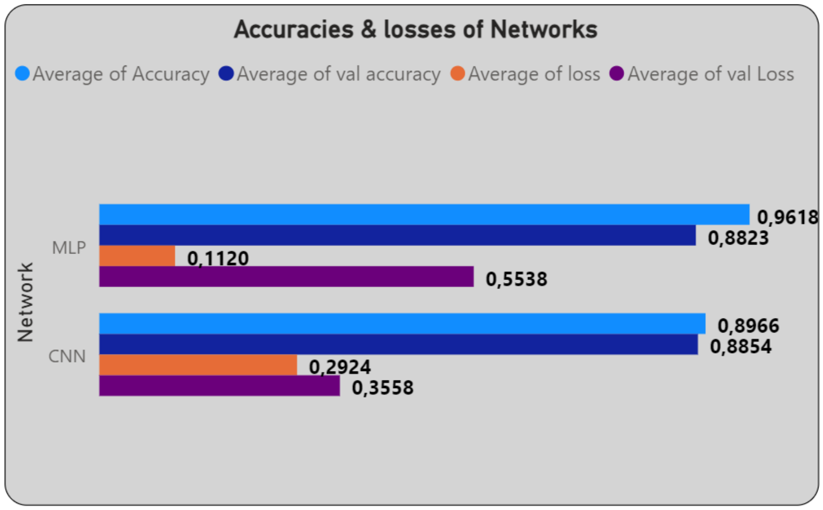

Figure 6: Architecture v/s Accuracies and Losses

Training vs. Validation Accuracies: Training accuracies are consistently higher than validation accuracies.

4 Related Work:

- Spectrogram Classification Using Dissimilarity Space(1)
- Features for Multi-Target Multi-Camera Tracking and Re-Identification (2)

5 Code and Acknowledgement

Website (Code is available at): https://github.com/SharathKumarReddyAlijarla

This project was conducted under the guidance of Prof. Dr.-Ing. Joeran Beel as part of the academic course titled "Recent Advancements in Machine Learning."

6 Conclusion:

Our findings reveal that the ideal learning rate for achieving optimal accuracy is 0.001, and deviations from this value result in decreased performance. Additionally, we observed that the number of layers did not follow a consistent trend in affecting the network's performance, emphasizing the importance of considering task complexity and dataset characteristics when choosing the number of layers.

Moreover, we discovered that HeUniform initialization outperformed GlorotUniform initialization in terms of both accuracy and loss. Surprisingly, the MLP architecture exhibited superior accuracy and lower loss than CNN on the MNIST dataset, which might be attributed to the small size of the images in the dataset.

Through a comprehensive analysis of our experiments and relevant literature, we have gained valuable insights into designing efficient and effective Siamese Neural Networks for similarity-based tasks. Our results contribute to advancing the field of machine learning and provide a foundation for future research in this domain.

References

[1] Loris Nanni, Andrea Rigo, Alessandra Lumini, and Sheryl Brahnam. *Spectrogram Classification Using Dissimilarity Space*. Applied Sciences, 10(12), 4176, 2020.

[2] E. Ristani and C. Tomasi. *Features for Multi-Target Multi-Camera Tracking and Re-Identification*. arXiv preprint arXiv:1803.10859, 2018.

[3] Yann, LeCun and Corinna, Cortes. *MNIST handwritten digit database*. `http://yann.lecun.com/exdb/mnist/`. Accessed on January, 2023.

[4] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. `https://www.tensorflow.org/`. Software available from tensorflow.org, 2015.

[5] DVL, Technical University of Munich. *Siamese Neural Networks - Lecture Slides*. Retrieved from `https://dvl.in.tum.de/slides/adl4cv-ss20/2.Siamese.pdf`.