

Optimized Classification for Deep Learning models applied Evolutionary Computation

John Tsiligaridis

*(Math and Computer Science Department, Heritage, University/Toppenish WA, USA

ABSTRACT

Deep Learning (DL) models face challenges in architectural design, which can lead to subpar model performance. Hyperparameter optimization is imperative, especially in the cases where there are many parameters in a given model. Neuro-evaluation using Evolutionary Methods (EMs), such as Genetic Algorithms (GAs), can optimize hyperparameters in many DL models. The purpose of this work is to optimize Convolutional Neural Networks (CNNs) with Variational Encoder (VAE) using GA.

Three different optimization approaches using CNNs are presented. The conventional CNN, the CNN architecture that is encoded into a gene sequence applied only to the selection and crossover operations, and the CNN architecture including the mutation operation. The loss validation across these models is studied with the Fashion-MNIST dataset.

VAE is a probabilistic generative process that uses an encoder/decoder architecture to learn lower dimensional representations of data that can be used to generate new data via sampling. Evolutionary VAE (EVAE) uses the combination of VAE and GA.

Using CNN as the encoder and decoder, we apply the GA operations in the EVAE, to optimize model hyperparameters and obtain satisfactory loss validation. Ultimately, we show that GA-optimized CNNs and EVAEs can successfully be used for classification.

Keywords :— Deep Learning, Variational Autoencoder, Genetic Algorithms, Convolutional Neural Networks.

I. INTRODUCTION

Model options range from activation and optimizer function to the addition of the number and size of layers. Layer depth is often dictated by the number of features a model needs to extract for learning, and activation functions that are typically dictated by the type of problem and form of data.

For any DL model there is the opportunity to continue with numerous of model variations (i.e. adding in more CNN layers or layers like BN, dropout or pooling). Then we can go through and get the various hyperparameters like kernel size, number of neurons, and filters. The whole process takes time and resources. CNN uses convolutional layers to automatically learn hierarchical features from image. The convolutional layers can slide across the input image, extracting features at various positions.

In [1], A comprehensive survey of autoencoders starts with an explanation of the principle of conventional autoencoder and their primary development process. A taxonomy of autoencoders, based on their structures and principles, are thoroughly analysed and the related models are discussed. Applications of autoencoders in various fields and their limitations are presented.

A new model, called a supervised auto-encoder has been analysed, theoretically and empirically. It is a neural network that jointly predicts targets and inputs (reconstruction) in [2].

VAEs are a type of generative model. They learn a compressed, continuous, and probabilistic latent representation of the input data from the latent space. The latent encoding vector uses two learned parameters the mean and the variance. VAE can be used for Dimensionality reduction and anomaly detection by measuring the reconstruction error.

A Nouveau VAE (NVAE), a deep hierarchical VAE built for image generation using depth-wise separable convolutions and batch normalization, is presented. NVAE is equipped with a residual parameterization of Normal distributions and its training is stabilized by spectral regularization in [3]. The application of GAs in optimizing Artificial Neural Networks (ANN) for phoneme recognition is investigated in [4]. This study examines the formalism of GA, their parameters, and operators, and describes the genetic strategy adopted for phoneme recognition using the TIMIT sound database. A new GA that is centred on the natural selection theory and it aims to improve the control of exploitation and exploration is presented. The proposed algorithm is called genetic algorithm based on natural selection theory (GABONST) in [5].

The predictive performance of ML models is challenged when trained on class imbalance real-world construction datasets. To improve the performance of construction prediction models a VAE was integrated into a multi-head graph attention Network (CAT) in [6]. Variational autoencoders (VAEs) are emerging as a powerful paradigm for the unsupervised data analysis, allowing to disentangle the factors of variability and discover optimal parsimonious representation in [7].

A Deep Learning approach that utilizes variational autoencoders (VAEs) to improve the energy efficiency of Wireless Sensor Networks (WSNs) by compressing transmission data is developed in [8]. A systematic examination of the impact that different combinations of variational auto-encoder models (measureVAE and adversarialVAE), configurations of latent space in the AI model (from 4 to 256 latent dimensions), and training datasets (Irish folk, Turkish folk, classical, and pop) have on music generation performance in [9]. Many forms of Evolutionary Computation are included in [10].

For classification it starts with training for learning the latent representation and then taking the output (the latent vector) from the encoder and use this as input for the decoder in order to have as final output similar to input. The Evolutionary VAE (EVAE) is incorporated with GA to improve the performance of the EVAE.

The grid search and the random search cannot provide solutions for complex Neural Network since there is a large number of hyperparameters combinations, with a wide range of values. It can be computationally expensive and very time-consuming. Also, there are problems in optimizing the complex network architecture and larger DL systems.

In larger networks there are problems like vanishing gradients, exploding gradients and network memorization problems. The normalization of input data is the answer for the first two problems. To avoid these time-consuming operations and the large network problems the DL architecture is transferred into genes using Genetic Algorithms. The CNN and the VAE are the tools that can cooperate with GAs providing optimization.

The paper organization is as follows. The CNN is included in Section 2. Section 3 deals with Genetic Algorithms, section 4 is referred to Optimization for CNN and GA, AE, VAE, EVAE are developed in section 5 and Simulation is in section 6,

II. CNN

The CNN with the convolutional layer works as a filter that tend to excel at learning short sequences. CNN layers are particularly adept at detecting spatial patterns. Most DL models' network size and layer types are driven by the variance of data and the type of features needed to be learned. For image classification problems, CNN layers are used to extract visual features. CNN are capable to extract relevant visual features from images that are critical to distinguish between different items.

A CNN architecture can be adapted using evolutionary algorithms. To this end GAs are used to encode a sequence of genes defining a CNN model for image classification. CNN components are: a. Convolution layers, b. Pooling layers. c. Dense layer.

The automatic optimization of a CNN model can be achieved by using GAs. The construction process deals with the addition of a new convolutional layer, with BN / Pooling layer.

III. GA

For image classification the appropriate hyperparameter setting needs to be discovered. This is a task that requires a special account. Many approaches have been developed to solve the problem. One of them is to combine metaheuristic algorithms (MA) with CNN. The advantage of MA is the possibility to successfully explore the search space, and it can work well with complex, non-linear problems without being stuck on local optima.

GAs are a subset of a much larger branch of computation known as Evolutionary Computation. GAs are optimization techniques based on evolutionary principles to find solutions to complex problems. They are used to solve complex problems by mimicking the process of evolution to improve a population of potential solutions iteratively.

The GA begins with an initial population of individuals, typically generated randomly. It then goes through a series of iterations, known as generations or epochs, in which the individuals undergo operations such as selection, crossover, and mutation. GA steps: Encoding the solution, initialization, evaluation, selection, crossover, mutation, create the new generation. During the selection phase, individuals from the current population are evaluated based on a fitness function, quantifying how well each solution solves the problem. The individuals with higher fitness values are more likely to be selected for further processing, simulating the survival of the fittest. The GA continues the selection, crossover, and mutation process for a fixed number of generations or until a termination criterion is met.

IV. OPTIMIZATION WITH CNN AND GA

In many cases there is a need of optimizing various aspects of the DL system, from tuning the learning rate to choosing the activation function. If this optimization process is done manually, this can take some substantial effort and time. It can encompass a wide variety of factors. Apart from the usual hyperparameter tuning a look at the network architecture itself is needed. Fortunately, evolutionary DL can provide several potential methods that automatically optimize the size and form of a network for a variety of problems.

Evolutionary methods provide an excellent toolset to optimize search in various problems solutions. For neuroevolutionary optimization of the GA is used instead of backpropagation. Regarding the application of GA to convolutional Autoencoder, the manual optimization process for a DL model may be very time demanding.

The use of CNN models with a variable number of layers containing a variable number of neurons in each layer, along with the different activation functions for each neuron, optimizer type and others are some of the hyperparameters to be discovered. To solve this problem, a combination of Evolutionary Methods (EM) with a DL model could be beneficial. One potential solution for a good classification performance is to compose a convolutional AE with CNNs for both encoder/decoders. Batch Normalization (BN) layers with dropout are included in CNNs for avoiding the

vanishing/exploding gradient issue and reducing memorization. For tuning the hyperparameters, a variety of time-consuming experiments would be necessary. Another possible solution could be the combination of the Convolutional AE (CAE) with GA by encoding the DL architecture into a gene sequence.

The GA is applied to the CNNs to optimize parameters as an alternative approach to the standard backpropagation (i.e. gradient descent) used in conventional DL problems. The encoding of the architecture into a gene sequence is achieved by using GA.

The construction process, which works with both the encoder and decoder, deals with the addition of a new convolutional layer, with batch normalization (BN)/pooling layer. Each individual represents the CNN architecture. The population is randomly initialized with the predefined population size, using the proposed encoding strategy. Population initialization is realized by the layers of encoder and decoder (i.e. convolutional layers, pooling layers etc.). For the architecture of the model, a function can create the needed layers for the encoder and decoder and creates the gene sequence.

For encoding, there are two phases. First, the encoding of the CNN layer structure (convolutional and max pooling, and Batch Normalization -BN-) into gene sequence of an individual (chromosome). Second, additional hyperparameters are defined for each layer type. The number of model parameters is the number of genes. Encoding a CNN into a variable length gene size sequence is achieved by the rule that all will start with a convolutional layer and finish with a dense layer (architecture encoding). Finally, a build function generates models from the gene sequences using the mating and mutation operators. Another function can create the parameters of the GA model (i.e. probabilities of crossover and mutation, number of genes in the individual, population size, and number of generations) to facilitate a successful search.

V. AE,VAE,EVAE

Autoencoders (AE) introduce unsupervised learning. CAEs are composed of the standard AE architecture and the convolutional layers. VAE is a probabilistic generative process that uses an encoder/decoder architecture to learn lower dimensional representations of data (“latent space”) that can be used to generate new data via sampling.

While for AE the learning is based on compressing and extracting relevant features in VAE the learning is achieved by using the mean and the variance of the input. The construction process involves the creation of gene sequences, the selection, the crossover, the mutation and discovering potential need for additional layers. For accomplishing network optimization, the evolution selects the best individual, with optimizing the fitness function, and providing better loss performance. The solutions are evaluated on the Fashion-MNIST dataset.

In VAE , after the use of the construction process, using a loop that adds CNN layers, the output of encoder is flattened, and the sampling layer is created. The purpose of a sampling layer is to learn how an input is represented with the training of the network to output the mean and the variance of the input. The sampling layer creates the latent encoding vector that is sent to decoder. In turn the decoder using the latent vector proceeds to develop a reconstruction of the original input.

For EVAE, the CNN is used as the encoder and decoder and applying the GA operations to optimize model hyperparameters and to obtain satisfactory loss validation. The loss includes the reconstruction loss and the Kullback-Leibler (KL) divergence [7].[8].

VI. SIMULATION

The optimization of the CA model needs to consider a set of parameters such as the number of convolutional layers, variable number of neurons in each layer, different activation functions for each neuron, etc. The problem of finding the combination of features that provide the best classification result cannot be an easy one and the operation is also time consuming.

The number of layers to be added in the autoencoder (encoder/decoder) should be equal to the length of the individual. The combination of features to be used for classification, using CNN layers architecture, plays the primary role for the GA extension model. Also, with the use of all three operations of GA a better performance can be achieved compared to the case of using only two of them. The EVAE can provide better performance after the correct optimization of a CNN using GA.

A. Experiments

The simulation for optimization is based on four experiments.

1) The conventional CNN model where the loss decreases for both training and validation dataset. A BatchNormalization (BN) layer is added between the pooling layer for better generalization of the model (Fig.1).

2) The CNN with GA along with the selection and crossover operations. After the 1st epoch the validation does not follow the training curve (Fig.2).

3) The CNN and GA with all the operations (selection, crossover and mutation). It is shown that there is an improvement of the validation loss (Fig.3).

4) Finally, the EVAE results are quite good at an average time. The EVAE model considers the reconstruction loss and the KL loss (Fig.4).

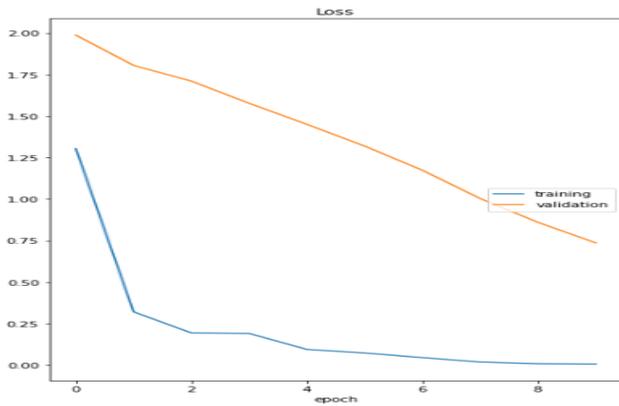


Fig.1 Loss, for CNN

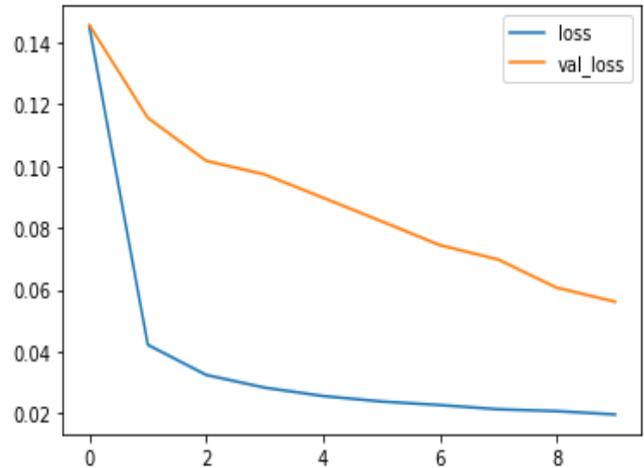


Fig.4 Loss with EAVE

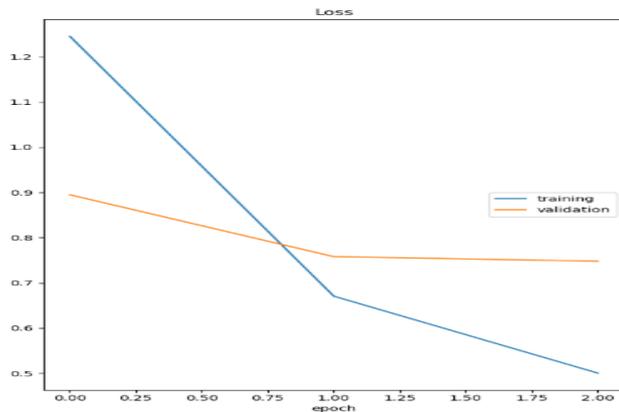


Fig.2 Loss for CNN, GA with selection and crossover

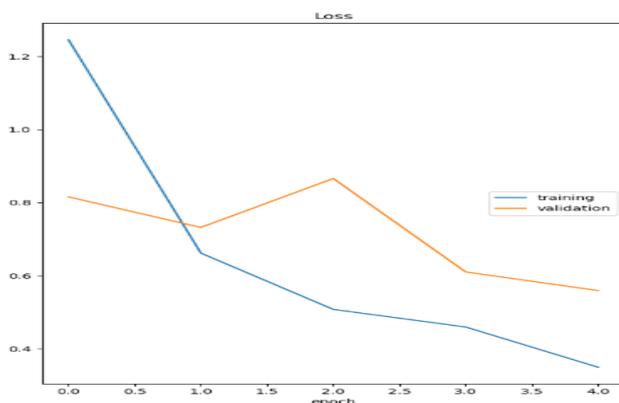


Fig.3 Loss, for CNN, GA with all the operations

VII. CONCLUSIONS

The primary purpose of VAE is to learn input image data, to compress it into lower dimensional representation and then reconstruct the output similar to input via sampling applied probabilistic generative process. Many aspects of VAEs, considering the latent space structure with the loss function on the digit MNIST dataset, were developed.

The application of GA on CNN, presented that the encoder/decoder components could successfully be optimized providing improved model performance over the fashion MNIST dataset.

Optimizing a network model often becomes the primary interest, and if done manually, this can take some substantial effort. Evolutionary algorithms can provide optimization solutions. The construction process for individuals includes the creation of the gene sequence and can check for additional layers. For optimization of large DL networks, apart from standard DL methods, GAs and EAVE can be used as alternative approaches to solve a multitude of problems after appropriate changes to the construction process are made.

Future work could include work with other image datasets, and various Evolutionary Computation techniques (i.e. Swarm Intelligence). The nature-inspired approaches can provide success due to their abilities for adaptability, global search capacity, scalability and parallelism.

REFERENCES

- [1] K. Berahmand, F. Daneshfarm, E. Salehi, Y. Li, Y. Xu, "Autoencoders and their applications in machine learning: a survey", Springer Nature, Vol. 57, No. 28, 2024 <https://link.springer.com/article/10.1007/s10462-023-10662-6>
- [2] L. Le, A. Patterson, A. Whit, "Supervised autoencoders: Improving generalization performance with unsupervised

- regularizers”, In Proceedings of Neural Information Processing Systems (NeurIPS). Montreal, 2-8 Dec. 2018
- [3] A.Vahdat, J. Kautz , “NVAE: A Deep Hierarchical Variational Autoencoder”, in Proceedings of Neural Information Processing Systems ,(NeurIPS),2020, <https://proceedings.neurips.cc/paper/2020>
- [4] M. Salhi, A. Alenezi, ” Voice recognition enhancement by genetic algorithm”,ScienceDirect, Heliyon, Vol.10, Issue:23,2024
- [5] M. Albadr, S. Tiun, M. Ayab, ” Genetic Algorithm Based on Natural,Selection Theory for Optimization Problems”, Symmetry, MDPI, 12(11), 1758,2020 <https://doi.org/10.3390/sym12111758>
- [6] F. Mastofi, O. Tokdemir, V. Togan,”, Generating synthetic data with variational autoencoder to address class imbalance of graph attention network prediction model for construction management”, Advanced Engineering Informatics, Elsevier, Vol. 62, Part A, Oct 2024,102606
- [7] M. Valletti, M. Ziatdinov, Y. Liu, S. Kalinin, “Physics and chemistry from parsimonious representations: image analysis via invariant variational autoencoders”, Nature, Aug. 2024, <https://www.nature.com/articles/s41524-024-01250-5>
- [8] B. C. Sengodan, P. M. Stanislaus, S. S. Arumugam, D. K. Sah, D. Dhabliya, P. Chenniappan, J. D. Hezekiah and R. Maheswar, ”Variational Autoencoders for Network Lifetime Enhancement in Wireless Sensors”, Sensors, MDPI, 23(17) 5630, Aug. 2024
- [9] N. Bryan-Kinns, B. Jhang, S. Zhao, B.Banar, “Exploring Variational Auto-encoder Architectures, Configurations, and Datasets for Generative Music Explainable AI”, Machine Intelligence Research, Springer, Vol.21, pp 29-45,2024, <https://link.springer.com/article/10.1007/s11633-023-1457-1>
- [10] V. Cicirello, “Evolutionary Computation: Theories, Techniques and Applications”, Applied Sciences, Vol.14, Issue: 6, 2024