International Journal of Mathematics and Computer Research

ISSN: 2320-7167

Volume 08 Issue 07 July 2020, Page no. - 2091-2095

Index Copernicus ICV: 57.55

DOI: 10.33826/ijmcr/v8i7.04

Evolutionary Methods in the Environmental Sound Classification

Anastasiia Kryvokhata

Department of Software Engineering, Zaporizhzhia National University, Zaporizhzhia, Ukraine

ARTICLE INFO

Published Online:
22 July 2020

Corresponding Author:
Anastasiia Kryvokhata
Zaporizhzhia National University, Zhukovsky str., 66, Zaporizhzhia, 69600, Ukraine
+380972100376

ABSTRACT

Developing neural network models is a significant research area, which involves different optimization problems. For instance, feature set optimization, searching of the best neural network architecture, hyper-parameters and weights optimization. All these problems do not have one solution. The most popular approaches here are Gradient-based methods and Evolutionary methods. Gradient-based methods are quite useful for neural connections weights optimization and for Neural Architecture Search. Evolutionary algorithms (NEAT, HyperNEAT, CoDeepNEAT etc.) are state-of-the-art methods for neural networks topology optimization. There are numerous heuristic methods like Particle Swarm Optimization that also could be used in the neural networks structural tuning.

This paper discusses the neural networks models and evolutionary methods in environmental sound classification systems. We consider Snapshot algorithm also in order to build ensemble of solutions.

KEYWORDS: Sound Classification, Convolutional Neural Network, Ensemble Method, Genetic Algorithm

I. INTRODUCTION

Artificial Neural Networks (ANN) and Deep Neural Networks (DNN) are state-of-the-art methods for various pattern recognition problems such as sound and image classification or detection. However, there are some known issues with applying ANN to real use cases [1]. Firstly, the development of an optimal ANN topology is a non-trivial problem and requires expert level knowledge from the researcher. Secondly, classical neural network types like Multilayer Perceptron, Convolutional Neural Networks need a large dataset at the training stage. Next, different feature set leads to various accuracy, so it is vital stage to define optimal features for particular problem.

These issues could be solved by the following approaches: Meta learning, Neuroevolution, Neural Architecture Search, Metaheuristic Methods. Meta learning methods (Siamese Networks, MAML, Reptile etc.) could be used to decrease size of the train dataset and to implement few-shot learning [2]. Neural Architecture Search and Neuroevolution methods could be used to automate neural networks topology development and to learn optimal feature space. ANN hyperparameter optimization could be provided with Metaheuristic Methods (Particle Swarm Optimization etc.). Thus, these methods increase ability to generalization and adaptability of the neural networks systems and their development is the urgent problem to be solved.

In this paper, we suggest hybrid DNN environmental classification system using CoDeepNEAT [3] and Particle Swarm Optimization (PSO) [4] methods for topology and hyperparameter optimization.

II. THE AIM AND OBJECTIVES OF THE STUDY

The state-of-the-art methods for sound classification are Convolutional Neural Networks (CNN), which are made up from Convolutional, Pooling, Dense, Dropout layers. There are different special layer types like Attention etc.

There are following hyperparameters here: number of filters and units; kernel size; kernel initializer and kernel regularizer; activation function type; pooling matrix size; dropout rate etc.

The topology of the network is defined by layers order and could be described as connectivity graph.

The aim of this paper is to design a proof-of-concept system for the environmental sound classification based on Convolutional Neural Networks with CoDeepNEAT and PSO methods for topology and hyperparameter optimization.

In order to reach the mentioned aim, the following objectives were formulated for the study:

– to review state-of-the-art of the neural architecture search methods;

– to develop the concept of the model for sound classification;

– to implement CoDeepNEAT method;

– to outline a direction for further development such adaptive classification systems.

We going to use UrbanSound8K dataset for training and testing classification model. This dataset contains 8732 labeled sound excerpts (<=4s) of urban sounds from 10 classes [5].

III. LITERATURE REVIEW

There are surveys dedicated to neural networks architecture search methods [6] and evolutionary metaheuristic approaches [4, 7]. These papers describe modern algorithms of automated machine learning systems.

In the papers [8, 9] Genetic Algorithm for CNN is implemented. Specific CNN block structure is proposed. Direct integer and float chromosome coding method is used.

Innovative approach of neural network morphism is used in the publications [10, 11]. This method allows to simplify the network topology without losing accuracy.

There are exists different variations of the evolution methods basic idea. For instance, in [12] authors suggest so-called coevolutionary approach. This method uses two different populations and their collaboration in order to produce optimal DNN topology.

Neural networks architecture search problem could be solved using both metaheuristics methods and continuous algorithms.

In the papers [13] suggested method that is based on continuous gradient descent optimization. Recurrent neural network (RNN) is used [14] as controller of time series of strings that describe particular DNN.

The alternative approach is to predict performance of different neural topologies using simple regression method [15].

Meta learning methods also could lead to more adaptive classification systems. For instance, such approach to one-shot topology learning is proposed in the paper [16].

System for automatically discover different machine learning algorithms is described in [17].

Genetic methods and numeric optimization approaches are combined in the articles [18, 19].

Metaheuristic technics for directional search are implemented in [20, 21].

Papers [22, 23] describe some variant of the term 'novelty' as the desire of the population to create new members.

Thus, there are following three directions in recent neural networks architecture search researches: implementing of the neuroevolution methods, developing of the hybrid metaheuristic and continuous algorithms, applying of the meta learning approaches.

IV. METHODOLOGY

In this paper, we suggest to use the CoDeepNEAT method for network topology optimization and the Snapshot ensemble method for training final best architecture [24]. To avoid time and memory leak at the CoDeepNEAT stage we apply undersampling approach to the train dataset. So, CoDeepNEAT learns on the relatively small dumps of original dataset.

The workflow of automated environmental sound classification system has shown in the figure 1.

The CoDeepNEAT method is an extension of NEAT algorithm [25]. The basic terms in the CoDeepNEAT are as follows: blueprint for graph relation structure, module for basic ANN building blocks. So, this algorithm is appropriate for CNN, because of its ability to discover different module types. The algorithm begins from very simple ANN structure as NEAT does. Then, after genetic crossover and mutation operators we obtain some population with relevant fitness scores. We consider standard multiclass classification problem, so categorical crossentropy function is the best candidate for fitness value.

In this implementation of the genetic algorithm, we use direct integer coding for chromosomes. So, other ANN hyperparameters are represented as dictionaries. Main dictionaries are following:

batch_size = {0: 8, 1: 16, 2: 32, 3: 64, 4: 128, 5: 256, 6: 512};

learning_rate = {0: 0.1, 1: 0.01, 2: 0.001, 3: 0.0001, 4: 0.00001};

epochs = {0: 10, 1: 300};

optimizer = {0: 'Adam', 1: 'Adadelta', 2: 'Adagrad', 3: 'Adamax', 4: 'Nadam'};

layer_type = {0: 'Conv2D', 1: 'Dense', 2: 'MaxPolling2D', 3: 'AveragePooling2D', 4: 'Dropout', 5: 'Attention3D'}

units_filters = {0: 8, 1: 1024};

stride = {0: 2, 1: 4, 2: 6};

kernel_size = {0: 2, 1: 4, 2: 6};

init = {0: 'RandomNormal', 1: 'RandomUniform', 2: 'Zeros', 3: 'Ones', 4: 'GlorotNormal', 5: 'GlorotUniform', 6: 'he_normal', 7: 'he_uniform', 8: 'lecun_normal', 9: 'lecun_uniform'};

regular = {0: 'l1', 1: 'l2', 2: 'l1_l2', 3: None};

activation = {0: 'relu', 1: 'sigmoid', 2: 'softmax', 3: 'softplus', 4: 'softsign', 5: 'tanh', 6: 'selu', 7: 'elu'}.

Figure 1: Sound classification system workflow

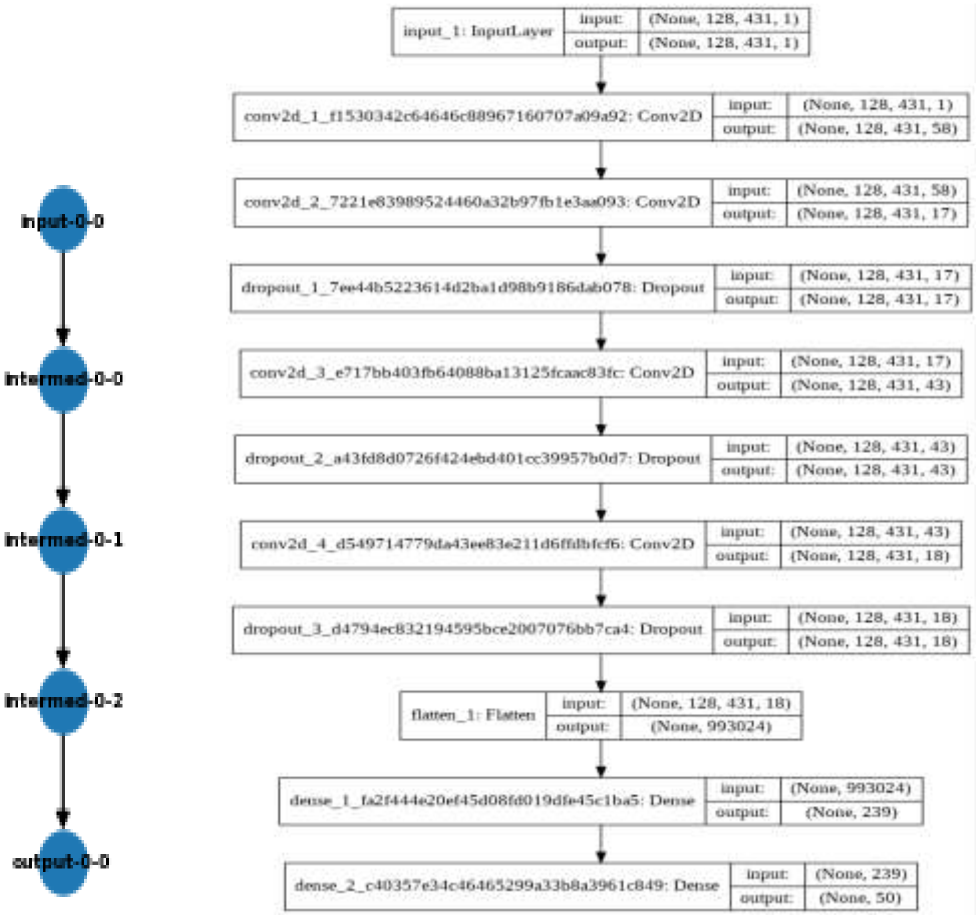

Figure 2: Connection graph and related neural network topology, zero generation

We consider in the CoDeepNEAT different levels of the solution. These levels are: module and blueprint. The representation of the possible solutions on the zero generation has shown in the figure 2.

Keras library is used for neural network implementation as it allows to use tensor processing unit (TPU).

Different non trivial topologies were generated already from the first generation as it has shown in the figure 3.

We use original CoDeepNEAT implementation (https://github.com/sbcblab/Keras-CoDeepNEAT) but have added data structures for TPU learning process.

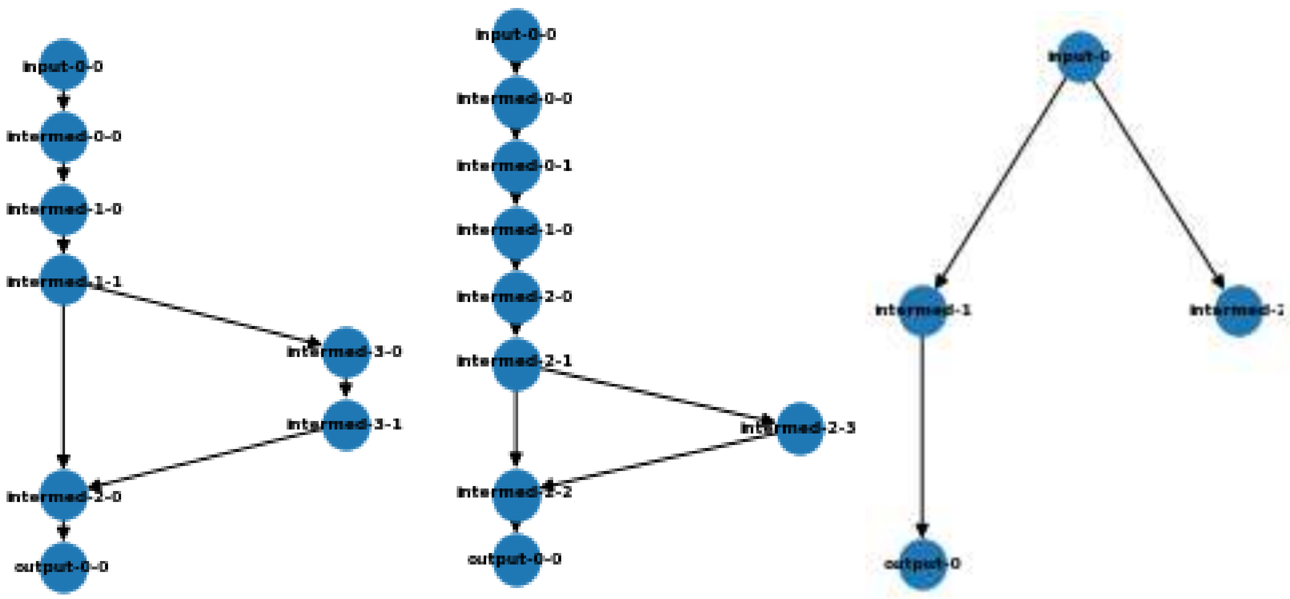

Figure 3: Neural network topologies at the higher generations

V. CONCLUSION

The convolutional neural networks in combination with CoDeepNEAT method for best topology searching and Snapshot ensemble method could be quite effective models for classification problems. Non-trivial architectures could be generated automatically while a few better solutions could be used for an ensemble.

The prospect of further research is related to the extension of the considered approach with meta-learning methods, for instance MAML or Reptile. This will help solve the problem of training data lack.

REFERENCES

1. Molnar C., König G., Herbinger J., Freiesleben T., Dandl S., Scholbeck C.A., Casalicchio G., Grosse-Wentrup M. & Bischl B. (2020). Pitfalls to Avoid when Interpreting Machine Learning Models. Preprint arXiv.org, 10 p, URL: https://arxiv.org/abs/2007.04131
2. Ravichandiran S. (2018). Hands-On Meta Learning with Python. Birmingham: Packt Publishing.
3. Bohrer J.S, Grisci B.I. & Dorn M. (2020). Neuroevolution of Neural Network Architectures Using CoDeepNEAT and Keras. Preprint arXiv.org, 29 p., URL: https://arxiv.org/abs/2002.04634
4. Sarmah D.K. (2020) A Survey on the Latest Development of Machine Learning in Genetic Algorithm and Particle Swarm Optimization. In: Kulkarni A., Satapathy S. & etc. Optimization in Machine Learning and Applications. Algorithms for Intelligent Systems. Springer, Singapore. DOI: https://doi.org/10.1007/978-981-15-0994-0_6.
5. Salamon J., Jacoby C. & Bello J.P. (2014). A Dataset and Taxonomy for Urban Sound Research. 2014 ACM Multimedia Conference (MM '14): Proceedings of the 22nd ACM international conference on Multimedia. (Orlando, 3-7 November 2014). Orlando, Florida, pp. 1041-1044, DOI: https://doi.org/10.1145/2647868.2655045
6. Elsken T., Metzen J.H. & Hutter F. (2019). Neural Architecture Search: A Survey. Journal of Machine Learning Research Vol. 20., pp. 1-21.
7. Hospedales T., Antoniou A., Micaelli P. & Storkey A. (2020). Meta-Learning in Neural Networks: A Survey. Preprint arXiv.org, 23 p, URL: https://arxiv.org/abs/2004.05439
8. Sun Y., Xue B., Zhang M., Yen G.G. & Lv J. (2020). Automatically Designing CNN Architectures Using the Genetic Algorithm for Image Classification. Preprint arXiv.org, 14 p, URL: https://arxiv.org/abs/1808.03818
9. Bakhshi A., Noman N., Chen Z., Zamani M. & Chalup S. (2019). Fast Automatic Optimisation of CNN Architectures for Image Classification Using Genetic Algorithm. Congress on Evolutionary Computation (CEC 2019): Proceedings of the 2019 IEEE Congress on Evolutionary Computation. (Wellington, 10-13 June 2019). Wellington, New Zealand, pp. 1283-1290, DOI: https://doi.org/10.1109/CEC.2019.8790197
10. Kwasigroch A., Grochowski M. & Mikołajczyk M. (2019). Deep neural network architecture search using network morphism. Methods and Models in Automation and Robotics (MMAR): Proceedings of the 2019 24th International Conference. Międzyzdroje, Poland, pp. 30-35, DOI:https://doi.org/10.1109/MMAR.2019.8864624
11. Radiuk P. & Kutucu H. (2020). Heuristic Architecture Search Using Network Morphism for Chest X-Ray Classification. Proceedings of the 1st International Workshop. (Khmelnitskyi, 10-12 June 2020). Khmelnitskyi, Ukraine, pp. 107-121. URL: http://ceur-ws.org/Vol-2623/paper11.pdf.
12. Tirumala S.S. (2020). Evolving deep neural networks using coevolutionary algorithms with multi-population strategy. Neural Computing and Applications. DOI: https://doi.org/10.1007/s00521-020-04749-2
13. Liu H., Simonyan K. & Yang Y. (2019). DARTS: Differentiable Architecture Search. Preprint arXiv.org, 13 p, URL: https://arxiv.org/abs/1806.09055/
14. Zoph B. & Le Q.V. (2017). Neural Architecture Search With Reinforcement Learning. Preprint arXiv.org, 16 p, URL: https://arxiv.org/abs/1611.01578.
15. Baker B., Gupta O., Raskar R., Naik N. (2017) Accelerating Neural Architecture Search using Performance Prediction. Preprint arXiv.org, 14 p, URL: https://arxiv.org/abs/1705.10823.
16. Bender G., Kindermans P.-J., Zoph B., Vasudevan V. & Le Q. (2018). Understanding and Simplifying One-Shot Architecture Search. PMLR 80: Proceedings of the 35th International Conference on Machine Learning. (Stockholm, 10-15 July 2018). Stockholm, Sweden, pp. 550-559.
17. Real E., Liang C., So D.R. & Le Q.V. (2020). AutoML-Zero: Evolving Machine Learning Algorithms From Scratch. Preprint arXiv.org, 23 p, URL: https://arxiv.org/abs/2003.03384
18. Gottapu R.D. & Dagli C.H. (2020). Efficient Architecture Search for Deep Neural Networks. Procedia Computer Science, Vol. 168 (2020), pp. 19-25, DOI: https://doi.org/10.1016/j.procs.2020.02.246
19. Lopes V. & Fazendeiro P. (2020). A Hybrid Method for Training Convolutional Neural Networks. Preprint arXiv.org, 6 p, URL: https://arxiv.org/abs/2005.04153

20. Lyu Z., Karns J., ElSaid A. & Desell T. (2020). Improving Neuroevolution using Island Extinction and Repopulation. Preprint arXiv.org, 11 p, URL: https://arxiv.org/abs/2005.07376
21. Gülcü Y. & Kuş Z. (2020). Hyper-Parameter Selection in Convolutional Neural Networks Using Microcanonical Optimization Algorithm. IEEE Access, Vol. 8, pp. 52528-52540. DOI: https://doi.org/10.1109/ACCESS.2020.2981141
22. Doncieux S., Paolo G., Laflaquière A. & Coninx A. (2020). Novelty Search makes Evolvability Inevitable. Preprint arXiv.org, 9 p, URL: https://arxiv.org/abs/2005.06224
23. Alet F., Schneider M.F., Lozano-Pérez T. & Kaelbling L.P. (2020). Meta-learning curiosity algorithms. Preprint arXiv.org, 22 p., URL: https://arxiv.org/abs/2003.05325.
24. Huang G., Li Y., Pleiss G., Liu Z., Hopcroft J.E. & Weinberger K.Q. (2017). Snapshot Ensembles: Train 1, Get M for Free. International Conference on Learning Representations (ICLR 2017): Proceedings of the 5th international conference. (Toulon, 24-26 April 2017). Toulon, France, pp. 1-14.
    URL : https://openreview.net/pdf?id=BJYwwY9ll
25. Kenneth O. Stanley, Risto Miikkulainen (2002). Evolving Neural Networks through Augmenting Topologies. Evolutionary Computation. Vol. 10(2), pp. 99-127. URL: http://nn.cs.utexas.edu/downloads/papers/stanley.ec02.pdf