PAPER • OPEN ACCESS

Fuzzy U-Net Neural Network Architecture Optimization for Image Segmentation

To cite this article: M M Kirichev *et al* 2021 *IOP Conf. Ser.: Mater. Sci. Eng.* **1031** 012077

IOP Conf. Series: Materials Science and Engineering 1031 (2021) 012077 doi:10.1088/1757-899X/1031/1/012077

Fuzzy U-Net Neural Network Architecture Optimization for Image Segmentation

M M Kirichev[1], T S Slavov[1] and G D Momcheva[1]
[1] Varna Free University "Chernorizets Hrabar", Department of Computer Science, Address: Yanko Slavchev str 84, Chayka Resort, 9007, Varna, Bulgaria

e-mail: galina.momcheva@vfu.bg

Abstract. In this article, the optimization of the modified U-Net neural network model extended with fuzzy layers has been studied with the usage of Grid search and Keras tuner. The article is a continuation of previous work where the model is suggested and explored. From one point of view, the research is focused on the optimization of Fuzzy Layers embedded in the U-Net model in order to find the better neural network architecture for nuclei segmentation in the research work in BioMed Varna R&D ecosystem for the segmentation of cellular nuclei. At the same time from a global perspective, this experiment is a part of the bigger one for the searching of new neural network architecture design techniques.

1. Introduction

Machine learning algorithms are used to solve many visual recognition complex problems nowadays. For example, issues such as ultrasound noise are handled through different machine learning techniques, to reduce that noise and see clearer visual results as well as get a better segmentation of different regions of interest [1]. Fuzzy computations and computational models with fuzzy elements give prospective results for biomedical purposes [2, 3].

The neural networks, which are used for image segmentation, are most commonly optimized through the training of a larger dataset or constructing a new model. However, as the results from our research show, this is not sufficient because every model depends somewhat on its set of hyperparameters and this can greatly change the final accuracy of the model. This can be proved by trying out models with different dropout values and through this experiment, we have found out that the accuracy of the same architecture could drop from 96.2% to 74.3%. Many of the current neural network architecture optimization techniques do not cover the hyperparametrization, which can lead to a much larger loss in terms of accuracy of the model.

Now many of the models used for biomedical image segmentation are different implementations of the convolutional neural network (CNN) since it is designed to produce image data from a certain input. One of the mostly used architecture is the U-Net, which allows for certain layers to be given an equal level of importance as opposed to the previous sequential models, where every layer is between 2 others, which produces a straight order of the data flow. Those main ideas for architectures can widely vary in their implementation, because CNNs can use a different number of layers, different number of neurons per layer, different arrangement of the layers inside, etc.

Here, in this article, experiments are provided, using another approach to design and optimise Fuzzy Neural Networks, i.e. how would the addition of Fuzzy Layers to an already existing Neural Network Architecture – namely, the U-Net – to evaluate the effectiveness of such design approach [2].

2. Related Work

The process of development of new machine learning tools is very extensive because it is important not only for research but also for business. There are some AutoML systems such as Auto-WEKA, Hyperopt-Sklearn, Auto-sklearn, Auto-Net. In them specific optimization techniques are embedded: searches over different classification and regression methods, their hyperparameter settings, and data preprocessing methods meta-learning for warm starting the optimization and automatic ensembling system for automated deep learning that selects both the architecture and the hyperparameters of deep neural networks. An early version of Auto-Net produced the first automatically tuned neural network that won against human experts in a competition setting Neural Architecture Search [5].

Another aspect of optimization is determining the optimal dropout values which is a key part in reducing overfitting. Randomly dropping out nodes during training offers a very computationally cheap and very efficient regularisation method in improving generalisation errors in deep neural networks [6].

At the same time optimization could be in the sense of searching for the most appropriate architecture. In Neural Architecture Search (NAS), the process of automating architecture engineering, is thus a logical next step in automating machine learning. NAS can be seen as a subfield of AutoML and has significant overlap with hyperparameter optimization and meta-learning. Some categorized methods for NAS are: search space, search strategy, and performance estimation strategy: search space, search strategy, performance estimation strategy [7].

However, designing a network with excellent performance is time-consuming and labor-intensive, which limits the progress of biological research. In [8] is proposed a neural architecture search (NAS) based solution for cell segmentation in microscopy images. In it the search space is restricted by exploring sophisticated network blocks. In this way, both expert knowledge and NAS are considered to facilitate the network searching. The authors attempt NAS with two prevailing backbone networks of U-net and Unet++. The experimental results on seven cell tracking challenge (CTC) microscopy cell data sets demonstrate that the searched networks achieve better performances with much fewer parameters than the baseline method. Thanks to its simplicity and transportability, the proposed method is applicable to many deep learning based cell segmentation methods [8].

The U-Net is developed especially for biomedical image segmentation and is widely used in visual recognition. The U-Net architecture consists of a contracting path capturing context and a symmetric expanding path that enables precise localization [4]. This Deep Learning Model encompasses both the well-known Fully CNN and the functional structure, giving better results to Biological Image Segmentation (e.g. segmentation of cell nuclei). There are many attempts to create fuzzy neural network architectures, in one of which multi-layer feedforward neural networks are used that can be fuzzified by using fuzzy numbers for inputs, targets and connection weights [9]. In our previous work, authors suggest the design of the fuzzy layers U-Net model [2]. The Fuzzy Layer U-Net model is constructed similarly to the original U-Net model - through the concatenation of Fuzzy Layers between the convolutional and deconvolutional layers. The idea behind this is that the U-Net has proven to be extremely useful because of the functional model architecture.

The process was divided into two main parts: training of a deep learning neural network model; and creation of a bitmask as the outcome. In the first step, the Fuzzy U-Net model is constructed by concatenating one or more Fuzzy Layers between the contracting and expanding path of a standard U-Net model. In the second step, several thresholding techniques are used and evaluated. This serves for the segmentation of nuclei cells.

To ensure that the placement of the Fuzzy Layers in the U-Net model would yield the best possible configuration of the model, a combinatorial search was implemented, covering all possible arrangements - 96 in total. The synthesized results of the Exhaustive Search can be found in Table 1. After finding the best combinations of fuzzy layers in terms of accuracy of the models, we used Keras Tuner's AutoML library to find the most reliable hyperparameters of the model. The parameters, for which we wanted to pin down the best values, were activation functions and dropout values. After considering the following algorithms - Exhaustive Search, Random Search, and Bayesian Search - we concluded that the best approach was the one that gave the best mix of speed and precision - the third method. The first would take too much time computing all possible combinations while the second would be much faster but would rarely yield desirable results. The Bayesian Search methods is the application of Bayesian

IOP Conf. Series: Materials Science and Engineering 1031 (2021) 012077 doi:10.1088/1757-899X/1031/1/012077

statistics to find as good hyperparameters as achievable in as little time as possible. In Table 2. are published the results of the hyperparameter optimisation using AutoML's Bayesian Search.

3. Results and Discussion

3.1. Training the model, data sets and Data Augmentation

For the training of the model are used: the dataset from the Kaggle 2018 microscopic nuclei images [10] and annotated images from public library of Medical University Varna, and augmented data based on real one examples. The solution was to use data augmentation to turn the provided images into a fully usable data set. After performing a list of transformations on each picture – namely rotation in the range -45 degrees to 45 degrees, flipping the image horizontally and vertically, stretching the image in both directions by a factor in the range of 0.5 and 1.5, mirroring the image on the outside bounds and others – we were able to turn no more than 40 images into almost 800. After the preparation of the preprocessing and finding the hyperparameters for the model, the model summary has shown that the trainable parameters are 1 947 616.

3.2. Acquiring the best possible FL configurations using Grid Search

Through inspection, we were able to deduce that not all combinations of Fuzzy Layers yield satisfactory results. Firstly, switching between Fuzzy Layers by hand but we quickly realized this would be inefficient in the context of time. The solution for finding the best combinations of Fuzzy Layers in the neural architecture with the least amount of human supervision was the following: After the construction of a certain model, consisting of a particular set of Fuzzy Layers, a Grid Search Algorithm looping through all possible variations of layers, marks all the best results according to the accuracy obtained.

After the successful implementation of the Grid Search combinatorial algorithm, the results for every possible set of configurations of the Fuzzy U-Net Architecture - with 1, 2, 3 and 4 Fuzzy Layers were recorded in a file. The sorting of the accuracy values has been used to acquire the best possible models.

In the table below the Roman numerals – I, II, III and IV – point to the Fuzzy Layers in the architecture, which are concatenated symmetrically around the convolution and deconvolution layers, with the first one being connected to the input and output layers, the second one - to the first MaxPooling 2x2 and the last Convolution2DTranspose (Deconvolution) layer, the third one to the second MaxPooling 2x2 and the second last Conv2DTranspose layer, etc. In general, besides the first FL, the i-th FL is concatenated to the i-th MaxPooling 2x2 and the (n-i)-the Conv2DTranspose layer, where i is the number of the current Fuzzy Layers and n – the total number of all Fuzzy Layers.

Table 1: Results from Grid Search.

Model	Accuracy of the model
I, II FL; 64 neurons	0.9626574
I, II FL; 32 neurons	0.96221095
I, II, III, IV FL; 32 neurons	0.9613269

3.3. Summarizing the results from the Grid Search

After analyzing the results some patterns were easily recognized - namely the last Fuzzy Layer never makes an appearance in the top results table. This is due to the fact that having a Fuzzy Layer so early in the model fuzzyfies the data too much without doing enough in return to be useful. All in all, the conclusion is that the best accuracies were achieved with a smaller number of Fuzzy Layers but not zero layers. We can also conclude that the best results include the first and the second Fuzzy Layers.

3.4. Hyperparameter optimisation using Keras Tuner's AutoML.

After acquiring the results from Table 1, a different optimization technique was needed to ensure that the best possible model architectures were found. The three models were trained and have had the exact same preprocessing in order to reduce ambiguity. However, that would mean that the models could have a better set of hyperparameters, which would result in a higher overall accuracy. That's why a new tool was used to determine those hyperparameters.

A simple trial and error approach wouldn't suffice in finding the most optimal configuration. The solution was to use Keras Tuner's AutoML methods. Several algorithms were tried out including Bayesian Search, Extensive Search and Random Search. Although the second would often find the best set of hyperparameters arrangement it proved to take way a very large amount of time, which does not allow for it to be considered useful (in comparison - a combinatorial grid search of the hyperparameters proved to be 1.5 times faster - i.e. 5.2 hours for the Extensive Search compared to 3.6 hours for a normal looping through all possibilities. The next approach – the Random Search algorithm – had a run time that was drastically lower, however, it would yield varying results depending on the seed of randomness, which were in a large range of accuracy values – between 0.82 and 0.96 approximately. Finally, the results showed that the third algorithm – Bayesian Search – proved to yield the best results in terms of both time and efficiency. With its quick run time and the fact that it yielded "visually satisfactory" results, it proved itself to be the best for the purpose of nuclei segmentation.

Each of the 3 models, shown in Table 1, has been put through Bayesian Search analysis from the Keras Tuner tool to produce the best set of hyperparameters. This was conducted through the hp.Choice() and hp.Float() for the activation function and the Dropout values. The set of the activation functions included the following options: 1) Rectifier (ReLu); 2) Sigmoid; 3) LeakyReLu; 4) Hyperbolic tangent – tanh. The Dropout ranges were between 7.5×10^{-1} and 1.25×10^{-2}. The default value for the dropout in the original U-Net architecture was 1×10^{-1}, and the default activation function – ReLu. After the Bayesian search on every model from the aforementioned, we received the following data results for the best set of hyperparameters:

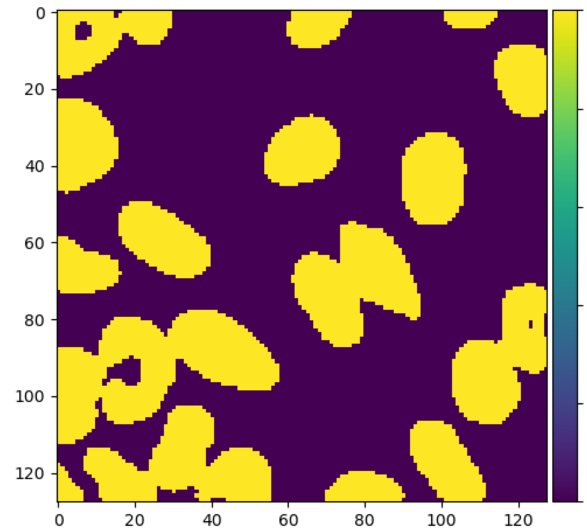
Table 2. Hyperparameters for the models from Table 1.

	FL I, II 64 neurons	FL I, II 32 neurons	FL I, II, III, IV 32 neurons
Activation function	ReLu	LeakyReLu	Tanh
Dropout Values	0.085	0.1	0.09

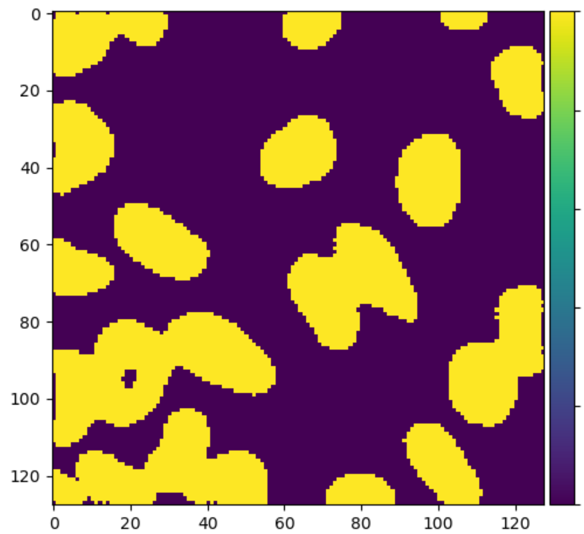
3.5. Thresholding

In order to extract the best possible bitmasks of the nuclei, we did an image normalization. After normalizing the data we had to turn the probability arrays which were spit out of the U-Net into bitmasks. This process is called thresholding. Different thresholding techniques were used, some proving better than others. The Manual thresholding expectedly performed the worst. Many of the nuclei were fused together, and some smaller ones were not annotated at all. Secondly, we tried adaptive thresholding. While the conventional (or manual) thresholding uses a constant or global value for the threshold of each pixel, the value for each pixel changes dynamically throughout the image with Adaptive thresholding. This more sophisticated technique can reflect changes in light e. g. strong illumination or shadows. Finally, an upgraded and customized Otsu thresholding [11] technique was implemented, which exploited the power of histograms, ensuring a sharper annotation. Otsu's algorithm exhaustively searches for the value that maximizes the inter-class variance, given the histogram representing the image in probability of each pixel having a certain intensity. The algorithm returns a constant value which is then used for thresholding the image and turning it into a bitmask. The comparison between all those techniques [2] proved efficient mostly for the visual representation of the data. The experiments have also shown that through Otsu's thresholding sharper edges of nuclei could be differentiated better, as well as cells which are very close to one another, but are parts of different ROIs.

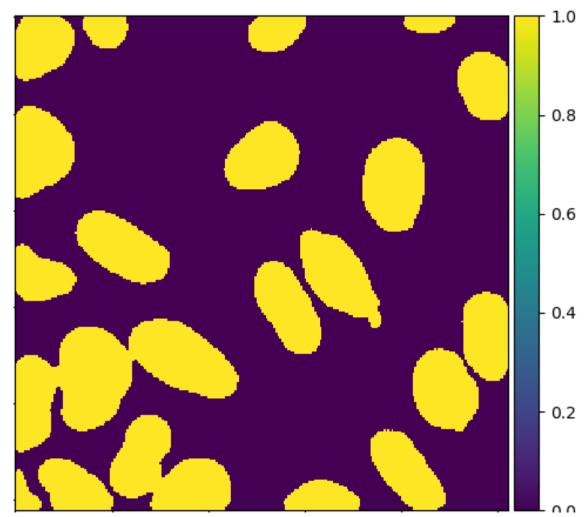
IOP Conf. Series: Materials Science and Engineering 1031 (2021) 012077 doi:10.1088/1757-899X/1031/1/012077



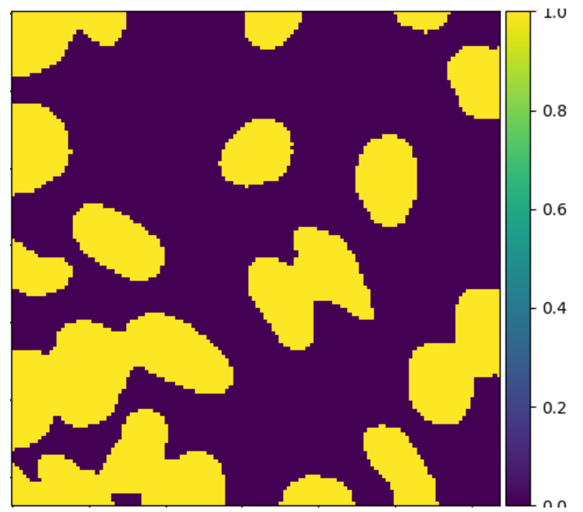
a. Original U-Net Model



b. FL I, II, 32 neurons Model



c. FL I, II, 64 neurons Model



d. FL I, II, III, IV, 32 neurons Model

Figure 1. Visual representation of results.

3.6. Comparing the results

Here are presented the results from after the segmentation. As it can be seen, the results from the segmentation of the 64 neurons with Fuzzy Layers on positions I and II yielded not only better results in terms of accuracy, but also in terms of visual representation of the desired results. In order for our work to be tested with the best results, the Keras AutoML Tool was used on the top 3 models – described in table 1 – in order for the architectures to be equipped with the best set of hyperparameters prior training. Also, the images from both sources were resized to fit a frame with a constant number of pixels on each side – 128. This was needed, because results have shown that a larger number of pixels corresponds to a better accuracy of the particular model tested. Thus, to eliminate this factor, we have used a constant one. Moreover, all of the models were subjected to the Otsu thresholding, the Neural Network Architectures have used the same seed, and they were trained with the same preprocessing in order to reduce ambiguity to a minimum. The results clearly show the differences between the original model and the 3 models, described in Table 1. In the original model some of the cells are not well segmented and the nuclei have areas where 2 ROIs are connected, but the original image does not show that. All 4 images are a result from the same input data. Looking closer, 2 major differences can be seen on the output data. Firstly, the Original U-Net Model (a) tolerates high-intensity pixels the worst from

IOP Conf. Series: Materials Science and Engineering 1031 (2021) 012077 doi:10.1088/1757-899X/1031/1/012077

all the proposed models. The upper left cell (from 0 to 18 pixels vertically and from 0 to 16 pixels horizontally) has a hole inside it, unlike the Fuzzy Layer Models. Moreover, the cell, found between 90 and 98 pixels vertically and between 11 and 20 pixels horizontally, misses the most number of pixels from all 4 samples. Secondly, the cells are better differentiated from one another and less cells stick together in sample (c) than in the other three samples.

4. Conclusion

In this article, we focused on the optimization of the modified U-Net neural network model extended with fuzzy layers. From one point of view, the research is focused on the optimization of Fuzzy Layers embedded in the U-Net model in order to find the better neural network architecture for nuclei segmentation in the research work in BioMed Varna R&D ecosystem for the segmentation of cellular nuclei. At the same time from a global perspective, this experiment is a part from the bigger one for searching for the neural network architecture design techniques.

References

[1] Wang X, Yuan J, Cheng Q, Wang X, Carson P 2019 Medical breast ultrasound image segmentation by machine learning - *Ultrasonics*, Volume 91, p. 1-9

[2] Kirichev M, Slavov T, Momcheva G 2020 Fuzzy U-Net Neural Network Design for Image Segmentation, *International Symposium on Bioinformatics and Biomedicine (accepted)*

[3] Zaychenko Y, Aghaei A, Nafas G 2019 Medical Images Classification and Diagnostics Using Fuzzy Neural Networks, *American Journal of Neural Networks and Applications* 5(2): 45-50

[4] Ronneberger O, Fischer P, Brox T 2015 U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image *Computing and Computer-Assisted Intervention, Springer*, LNCS, Vol.9351: 234--241, available at arXiv:1505.04597

[5] Hutter F, Kotthoff L, Vanschoren J 2019 Automated Machine Learning Methods, Systems, *Challenges The Springer Series on Challenges in Machine Learning*

[6] Poernomo A, Kang D 2018 Biased Dropout and Crossmap Dropout: Learning towards effective Dropout regularization in convolutional neural network, *Neural Networks, Volume 104*, p. 60-67, ISSN 0893-6080

[7] Elsken T, Metzen J, Hutter F 2019 Neural Architecture Search: A Survey, *Journal of Machine Learning Research* 20 p. 1-21

[8] Zhu Y, Meijering E 2020 Neural Architecture Search for Microscopy Cell Segmentation, International Workshop on Machine Learning in Medical Imaging, *MLMI 2020: Machine Learning in Medical Imaging Springer* p. 542-551

[9] Ishibuchi H 1996 Development of Fuzzy Neural Networks. *International Series in Intelligent Technologies, vol 7. Springer*, Boston, MA

[10] Kaggle Data Science Bowl 2018 https://www.kaggle.com/c/data-science-bowl-2018/data

[11] Zhu N, Yang W, Dai W 2009 A Fast 2D Otsu Thresholding Algorithm Based on Improved Histogram *Chinese Conference on Pattern Recognition*, Nanjing pp. 1-5, doi: 10.1109/CCPR.2009.5344078