Deep Learning Instead of Mathematical Decision Making Statistics for Change Point Detection

Elena N. Benderskaya[1], Arseny M. Sazanov[2]
Peter the Great St. Petersburg Polytechnic University
St. Petersburg, Russia
[1]helen.bend@gmail.com, [2]arseny.sazanov@gmail.com

***Abstract*— The paper compares classical algorithms based on mathematical statistics for making decisions about the presence or absence of changes in the properties of a random process and a bioinspired approach based on neural networks of deep learning. The advantages of neural networks and the new approach difficulties are analyzed in the paper.**

Keywords*— *deep learning; neural network; changing the properties of random processes; statistics for decision-making

I. Introduction

The task of processing time series arises in many applications. For functional diagnostics, the processing of the results of work over time in the information-measuring system is transformed into the search for the moments of changing the properties of the time series-switching from one model related to the normal functioning of the intelligent measuring systems (IMS) to another model associated with the appearance of the defect [1].

The peculiarity of this formulation of the problem is the need to determine the switching moment as soon as possible. Formalization in this case leads to change detection problem – the fastest detection of changes in the properties of a stochastic process. Successive methods of solving such a problem are used in the classical approach. These methods are usually based on statistics for decision making.[2, 3, 4] The need a number of solutions based on the neural network approach can be proposed to solve the problem in real time and to detect even small deviations from the basic model of normal functioning.[1, 5].

II. Feedforward Neural Networks and Changes Detection Problem

The first attempts to apply neural networks of direct propagation led to parallelization of the general decision algorithm. It did not significantly improve the quality of detection: a significant reduction in the average number of steps of delay in the change detection at a fixed level of false detection probability could not be achieved.

Various variants of the neural network structure formation were considered and implemented. Different variants of setting the problem of intelligent measuring systems (IMS) functional diagnostics [1] were taken into account.

Some results of neural network structures simulation and the corresponding characteristic of detection quality that depend on the magnitude of the disruption are presented in Fig. 1.

In the latter case (a neural network of direct propagation with several hidden layers, trained on samples of normal functioning and functioning with a minimally detectable defect such as a shift in the mathematical expectation corresponding to technical requirements), one can speak of the implementation of an algorithm with generalized statistics for decision making and a multi-threshold decision-making system.

As it can be seen from the characteristic detection curves, the direct propagation network can be tuned to the required probability of false detection and, at the same time, provide a better delay time in detecting small disparities in both expectation and dispersion.

III. Deep Learning Networks and Changes Detection Problem

Successful use of deep learning networks to solve problems of static images recognition naturally leads to the desire to apply this approach in order to the changes detection problem. In this case, for training and testing, as well as for the direct distribution networks, it is necessary to create sets of training and test sequences from the time series by shifting the observation windows.

In this paper, the basic model of CNN [6] for the 10th class image classifier of the CIFAR-10 dataset is taken as a basis, see Fig. 2.

Structure of the basic CNN:

- input layer;
- convolution layers;
- subsampling layers;
- fully-connected hidden layer;
- output classification layer.

The rectified linear activation function ("Relu") is used in the convolution layer and hidden layer as activation tool. "Softmax" activation function is used for the output layer.

978-1-5386-5824-6/18/$31.00 ©2018 IEEE

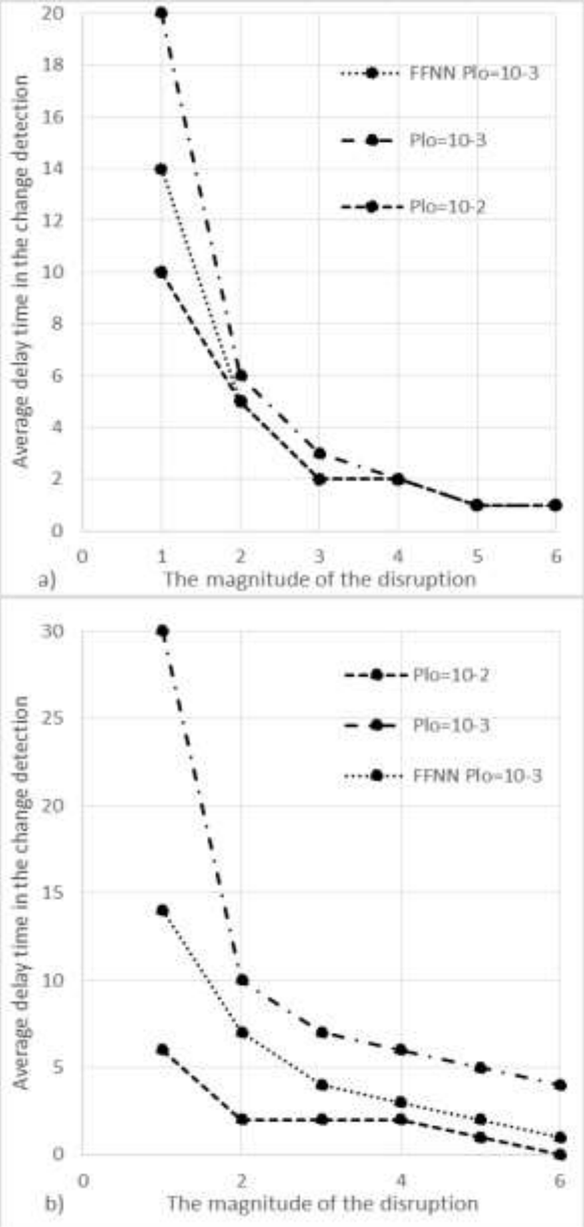

Fig. 1. First method of synergetic computer building

A number of changes in architecture were introduced according to the task:

- the number of classes were reduced to two (classification was made on the defect presence basis);
- one-dimensional convolution and subsampling layers (Conv1D and MaxPooling1D) are used, because the input image size is a one-dimensional sequence (vector);
- as CNN has a tendency to overfit [7] the Dropout layers were immediately added to prevent overfitting (their task was to exclude the random number of neurons from the learning process).

The structure of the network is shown in Fig. 3

A. CNN software implementation tools

To implement this task, the following software was selected:

- Python programming language;
- open Keras neural network library [8] written in Python, which is a high-level add-on for machine learning frameworks: TenzorFlow, Theano, and others;
- TenzorFlow [9] was selected as a backend for Keras;
- Pycharm IDE (Free Community version).

B. Learning and testing

The following parameters were selected to learn the model. The volume of training and test samples amounted to 200,000 sequences. The length of one sequence was T = 20 counts. Each sequence was divided into N windows with M = 10 samples, which were fed to the HC input. Accordingly, N = T-M = 10, the total volume of the test sample is 2 million windows. Change point - a random value in the range t = (M: T].

The ratio of the sequences with the defect to the sequences without defects was chosen 1 to 1. The ratio of the windows with the defect to the windows without a defect was also chosen 1 to 1, due to the training quality metric - the accuracy of classification by classes. The metric for the error is "categorical cross entropy". The optimizer: "SGD", lr = 0.001. The number of learning epochs is 100. Early stopping on training is 25. The training of the model was carried out on the CPU (PC Intel Core I-7 2600K, RAM 32 Gb.). One epoch lasted about 100 seconds.

C. The results of the experiments

The following results were obtained during the experiments. Probability of the change point detection on the defect M = 1 (shift of the mathematical expectation) - 62% (accuracy).

After post processing the windows for the sequences:

- detection of a defect without delays (at the time of the change point) 6.7%;
- defect detection with delays: 27.4%;
- false defect detection 44.8%;
- the correct no defect detection is 21.1%.

The network showed good results, the average detection time was 1.37 (variance 3.75). However, this result spoils the high probability of false detection.

The network has a tendency to over fit. The network demonstrated more than 95% accuracy at the train samples, nerveless at the test samples could not reach 55% of accuracy. The network is sensitive to the initial data and to the classes representation in the learning sequences. It should be noted that no special preprocessing was carried out on the initial data, except for their subdivision into smaller sequences ("windows"). In addition, the network was practically not optimized for a specific task – the basic architecture was used.

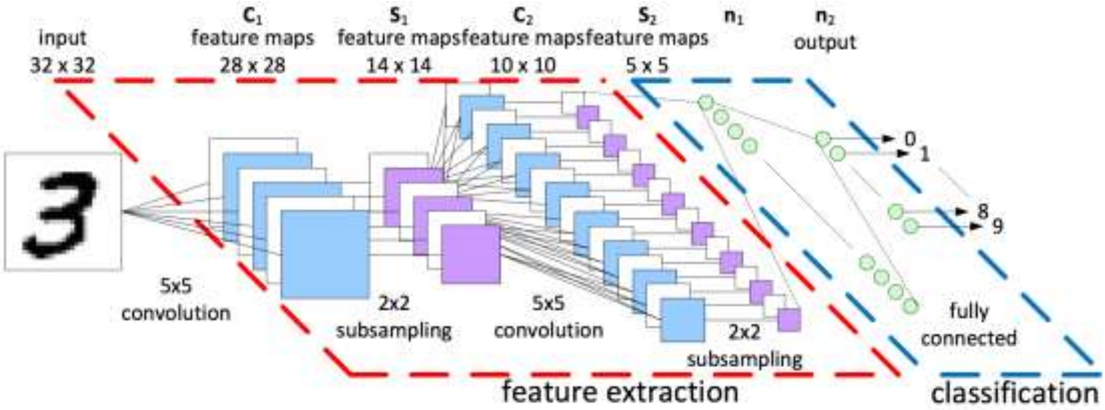

Fig. 2. Basic model of CNN for solving the problem of classification of images for ten classes [6]

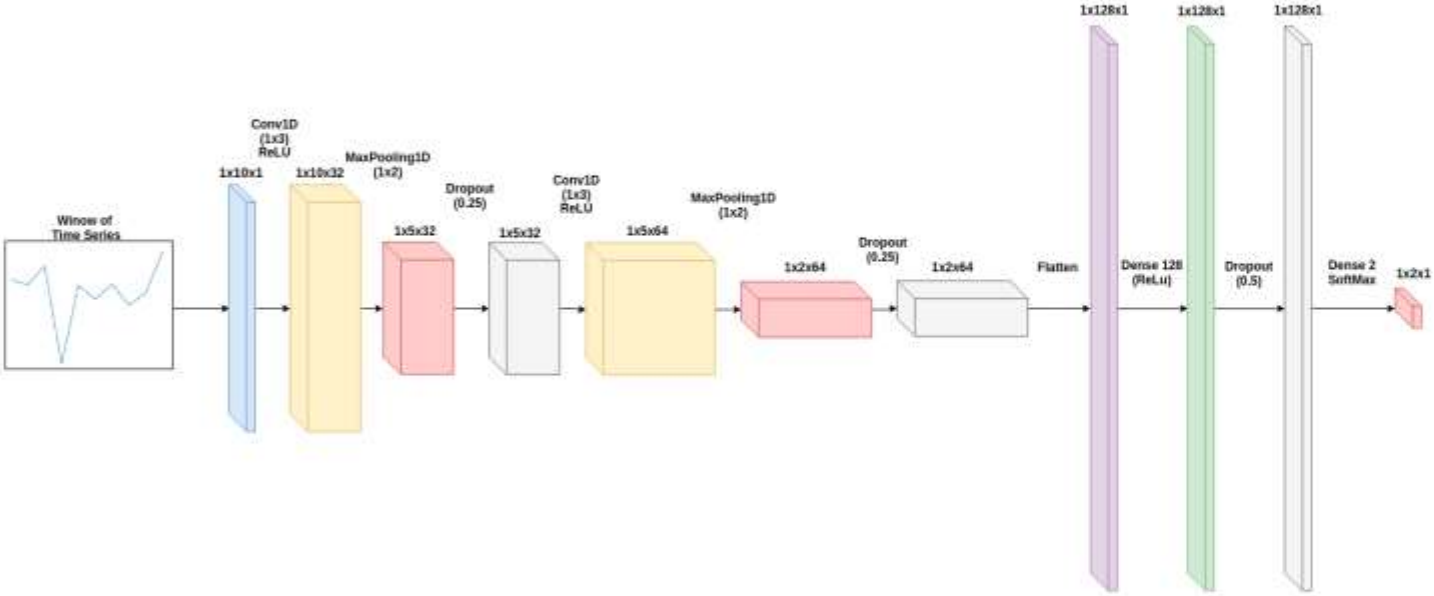

Fig. 3. The architecture of the investigated convolutional neural network (CNN) for the changes detection problem

And the training sample had a relatively small amount applied to this task (due to the time-consuming calculation of the CNN on the CPU). In the future, it is planned to use the GPU to train the CNN, and increase the sample size and complexity of the model.

The article demonstrates the principal possibility of solving the problem of abrupt changes using CNN, with further optimization and complexity of the network, you can achieve significantly better results.

Options for improving the quality of the results obtained are:

- To combine several models, each of which will be configured separately.
- To complicate the current model: to increase the window size, to increase the depth of the model, to change the parameters of convolution kernels and subsampling layers.
- To change the input format, the vector to an array of window vectors, offset in time.
- To convert data to another form, (for example, a synthetic image and its preprocessing) or presenting data in another form (for example, changing the coordinate system to a polar coordinate system).

IV. Conclusion

The advantage of the approach based on deep learning networks is a large variety of architectures that, when properly selected, allow solving the problems of functional diagnostics in different settings-like the detection of different types of disruption (mathematical expectation and variance, and, in the long term, the autoregression coefficient), and the localization of the disorder and recoverable changes detection.

The features of the detection method based on the network of deep learning – the probability of a defect detecting is practically independent of the magnitude of the defect and the detection rate of a defect is practically independent of the magnitude of the defect. This features can be used later to configure the small changes detection in model's.

One of the main drawbacks of deep learning networks is the absence of structure universality and the lack of scientific sound recommendations for the selection of the structure. In addition, there are almost no self-learning structures, only training with the teacher. This does not allow to achieve good adaptability when changing the situation and the functionality expansion needs.

REFERENCES

[1] Benderskaya E.N., Kolesnikov D.N., Pahomova V.I. Functional diagnostics of control systems. Saint Petersburg. SPbGTU. 2000. 144 p. (in Russian)

[2] Qi J.-P., Qi J., Zhang Q. A Fast Framework for Abrupt Change Detection Based on Binary Search Trees and Kolmogorov Statistic. Computational Intelligence and Neuroscience. Vol. 2016. Article ID 8343187. 16 pages. 2016. Available at: https://doi.org/10.1155/2016/8343187.

[3] Alippi C., Boracchi G., Roveri M. Hierarchical Change-Detection Tests. IEEE Transactions on Neural Networks and Learning Systems. Vol. 28. No. 2. Pp. 246-258. 2017.

[4] Volkhonskiy D., Burnaev E., Nouretdinov I., Gammerman A., Vovk. V. Inductive conformal martingales for change-point detection. Proceedings of Machine Learning Research 60. Pp. 132-153. 2017.

[5] Benderskaya E.N. Nonlinear Trends in Modern Artificial Intelligence: A New Perspective. Beyond AI: Interdisciplinary Aspects of Artificial Intelligence. Topics in Intelligent Engineering and Informatics. Springer. Vol. 4. Pp. 113–124. 2013.

[6] Peemen M., Mesman B., Corporaal H. Efficiency optimization of trainable feature extractors for a consumer platform. Proceedings of the 13th International Conference on Advanced Concepts for Intelligent Vision Systems. Springer. Pp. 293–304. 2011.

[7] Wan Li, Zeiler Matthew, Zhang Sixin, Le Cun Yann, Fergus Rob. Regularization of Neural Networks using DropConnect. Proceedings of the 30th International Conference on Machine Learning. PMLR 28(3). Pp.1058-1066. 2013.

[8] Keras. The Python Deep Learning library. Available at: https://www.keras.io/ (accessed 21.05.18).

[9] About TensorFlow. Available at: https://www.tensorflow.org/ (accessed 21.05.18).