Selection of neural network architecture and data augmentation procedures for predicting the course of cardiovascular diseases

M G Dorrer¹, S E Golovenkin², S Yu Nikulina², Yu V Orlova², E Yu Pelipeckaya² and T D Vereshchagina²

¹Reshetnev Siberian State University of Science and Technology, 31, Krasnoyarsky Rabochy Ave., Krasnoyarsk, 660037, Russian Federation
²Professor V.F. Voino-Yasenetsky Krasnoyarsk State Medical University, 1, Partizana Zheleznyaka Street, Krasnoyarsk, 660022, Russian Federation

E-mail: mdorrer@mail.ru

Abstract. The article solves the problem of creating models for predicting the course and complications of cardiovascular diseases. Artificial neural networks based on the Keras library are used. The original dataset includes 1700 case histories. In addition, the dataset augmentation procedure was used. As a result, the overall accuracy exceeded 84%. Furthermore, optimizing the network architecture and dataset has increased the overall accuracy by 17% and precision by 7%.

1. Introduction
Diseases of the heart and blood vessels are the leading cause of death in most economically developed countries. Every year about 17 million people die of this pathology [1]. In Russia, for several decades, cardiovascular diseases, as well as in other countries, have been the leading cause of death among the population (50.1%) [2,3,4]. Therefore, early diagnosis of heart disease could significantly reduce unwanted statistics.

2. Related work
In 1990, William Bakst [5] of the University of California, San Diego used a neural network – a multilayer perceptron - to recognize myocardial infarction. In addition, Bakst used only 20 parameters, among which were age, gender, localization of pain, reaction to nitroglycerin, nausea, and vomiting, sweating, fainting, respiratory rate, heart palpitations, previous heart attacks, diabetes, hypertension, distention of the cervical vein, some ECG features and the presence of significant ischemic changes. The network demonstrated 92% accuracy in detecting myocardial infarction and gave only 4% of false alarms, falsely confirming the referral of non-infarcted patients to the cardiology department.

An ideal diagnostic method should have one hundred percent sensitivity and specificity - firstly, do not miss a single sick person and, secondly, do not frighten healthy people. To be insured, you can and should try, first of all, to ensure one hundred percent sensitivity of the method - you must not miss the disease. But this usually results in a low specificity of the technique.
Neural networks are intensively used in various fields of science [6,7]. Their use is advantageous in the following cases:

1. Due to specific features, the problem under consideration does not lend itself to adequate formalization since it contains elements of uncertainty that are not formalized by traditional mathematical methods.
2. When the problem under consideration is formalized, but there is currently no apparatus for its solution.
3. When for the considered, well-formalized problem, there is a corresponding mathematical apparatus, but the implementation of computations with its help based on existing computing systems does not satisfy the requirements for obtaining solutions in time, size, weight, energy consumption, etc., in such a situation it is necessary to either produce simplification of algorithms, which reduces the quality of solutions, or apply an appropriate neural network approach, provided that it provides the required rate of the task.

3. Research methods
As the conclusion to work [7] mentioned, the authors attempted to reproduce its results using the Python language and the Keras library.

Keras functions can work in TensorFlow, CNTK, or Teano environments. Keras was developed with an emphasis on the ability to experiment quickly. Going from idea to result with the lowest possible latency is key to effectively conducting neural network research.

The intensive development of neurocomputer science has led to the emergence of many network architectures that have minimal differences in their structure but at the same time have different spectra of applicability and computational capabilities. The impossibility of a neural network solution to this problem or obtaining less accurate results than when using the conventional mathematical apparatus often indicates an incorrect choice of a neural network model. The neural network approach in solving some mathematical problems of medical diagnostics can significantly simplify the solution process. It allows you to avoid cumbersome mathematical transformations associated with the search for patterns between input and output data, often complicated by the need to recover laws and probabilities distribution functions.

The initial dataset was formed based on the database [8]. Then, a neural network was used to predict diagnoses, consisting of layers Dense, BatchNormalization and DropOut (see figure 1, 2).

![Figure 1. 'build_model' function.](image-url)
The study used the following accuracy metrics based on the Confusion Matrix: overall prediction accuracy, precision, recall.

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)
\]
\[
\text{Precision} = \frac{TP}{TP + FP} \quad (2)
\]
\[
\text{Recall} = \frac{TP}{TP + FN} \quad (3)
\]

Where TP – true positive diagnosis, TN – true negative, FP – false positive, FN – false negative.

4. Database augmentation and balancing

A data augmentation procedure [9] was applied to overcome the lack of data in some experiments.

The data augmentation procedure was applied to solve the small number of each diagnosis in the database, figure 4.
def AugmentRow(row, AugmentedDF, DFValue, PositiveValue):
    if row[-1]==1:
        for i in range(0, round((DFValue-PositiveValue)/PositiveValue)):
            AugmentedDF.loc[-1]=row
            AugmentedDF.index=AugmentedDF.index+1
            AugmentedDF.sort_index(inplace=True)

Figure 4. Lambda function 'AugmentRow' for dataset augmentation.

At the same time, it was evident that you should not use the extended sample directly because it will contain a significant number of examples that coincide with the test sample. The procedure for solving this problem is shown in figure 5. The Python code that implements this process is shown in figure 6.

Figure 5. The training procedure (BPMN).

def Teach(workrow, dataset, model1, model2, model3, W_init1, W_init2, W_init3, N_out):
    TempDF = dataset.drop(dataset[dataset.ID == workrow.ID].index)
    TempX=TempDF.iloc[:,:len(TempDF.columns)-N_out]
    TempY=TempDF.iloc[:,len(TempDF.columns)-N_out:len(TempDF.columns)]
    subrow=workrow[0:len(TempDF.columns)-N_out]
    testdata=pd.DataFrame([subrow])
    model1.set_weights(W_init1)
    model1.fit(TempX,TempY,epochs=200, batch_size=len(TempDF),verbose=0, shuffle=True)
    predictions1=model1.predict(testdata)
    OutY.append(predictions1)
    print("Records ",len(OutY))

Figure 6. Teach lambda function.

The Teach and AugmentRow function in a sliding window loop to predict all diagnoses in the database [8] is done in the Python code snippet shown in figure 7.

Figure 7. Solving the prediction problem.

5. Applied architectures
A set of experiments is shown in table 1.
Table 1. Description of Computational Experiments.

| # | Dataset augmentation | Architecture (hidden layers) | Epochs | Consilium | Experiment |
|---|----------------------|-----------------------------|--------|-----------|------------|
| 1. | No                   | Dense (64)                  | 100    | No        | Dense_64_1_100ep |
| 2. | Yes                  | Dense (64)                  | 100    | No        | Dense_64_1_augment_100ep |
| 3. | Yes                  | Dense (64) + Dense (32)     | 100    | No        | Dense_64_32_1_augment_100ep |
| 4. | Yes                  | Dense (100) + Dense (50) + Dense (25) + Dense (12) | 200    | No        | Dense_100_50_25_12_DO_BN_augment_200ep |
| 5. | Yes                  | Dense (64) + Dense (32) + Dense (16) | 200    | No        | Dense_64_32_16_DO_BN_augment_200ep |
| 6. | Yes                  | Dense (64) + Dense (32) + Dense (16) | 100    | Yes       | Consilium_Dense_64_32_16_DO_BN_augment_100ep |
| 7. | Yes                  | Dense (64) + Dense (32) + Dense (16) | 100    | No        | Contrast_Dense_64_32_16_DO_BN_augment_100ep |
| 8. | Yes                  | Dense (64) + Dense (32) + Dense (16) | 100    | Yes       | Contrast_Consilium_Dense_64_DO_BN_augment_100ep |

6. Results
Table 2 shows the diagnostic quality metrics for the respective experiments. In addition, the plan of experiments for selecting the optimal architecture is included in the comparative assessment of the combinations of the studied architectures and the sets of training and test samples shown in table 2.

Table 2. Comparative indicators of diagnostic quality metrics for experiments.

| # | Experiment                                      | Metric          |
|---|-------------------------------------------------|-----------------|
|   | 1. Dense_64_1_100ep                             | Accuracy        |
|   |                                                | 64.64%          |
|   | 2. Dense_64_1_augment_100ep                     | Precision       |
|   |                                                | 4.35%           |
|   | 3. Dense_64_32_1_augment_100ep                  | Recall          |
|   |                                                | 0.06%           |
|   | 4. Dense_64_32_16_DO_BN_augment_100ep           | Accuracy        |
|   |                                                | 65.31%          |
|   | 5. Dense_64_32_16_DO_BN_augment_200ep           | Precision       |
|   |                                                | 6.13%           |
|   | 6. Dense_100_50_25_12_DO_BN_augment_200ep       | Recall          |
|   |                                                | 38.77%          |
|   | 7. Consilium_Dense_64_32_16_DO_BN_augment_100ep | Accuracy        |
|   |                                                | 66.21%          |
|   | 8. Contrast_Dense_64_32_16_DO_BN_augment_100ep  | Precision       |
|   |                                                | 5.59%           |
|   | 9. Contrast_Consilium_Dense_64_DO_BN_augment_100ep | Recall         |
|   |                                                | 33.89%          |
|   |                                                | 72.14%          |
|   |                                                | 5.86%           |
|   |                                                | 24.69%          |
|   |                                                | 74.09%          |
|   |                                                | 5.71%           |
|   |                                                | 25.12%          |
|   |                                                | 76.65%          |
|   |                                                | 5.69%           |
|   |                                                | 21.91%          |
|   |                                                | 76.98%          |
|   |                                                | 6.54%           |
|   |                                                | 25.19%          |
|   |                                                | 83.94%          |
|   |                                                | 5.44%           |
|   |                                                | 12.41%          |

On the example of one of the experiments, we will consider the detailing of the metrics of the quality of diagnostics in the context of diagnoses of the database.
Table 3. The quality of diagnostics is based on the example of the experiment Dense_64_32_16_DA_BN_augment_100ep.

| Diagnosis       | TP    | TN    | FP    | FN    | Accuracy | Precision | Recall |
|-----------------|-------|-------|-------|-------|----------|-----------|--------|
| FIBR_PRED       | 52    | 1157  | 373   | 118   | 71.12%   | 12.24%    | 30.59% |
| PRED_TAH        | 9     | 1242  | 438   | 11    | 73.59%   | 2.01%     | 45.00% |
| JELUD_TAH       | 10    | 1249  | 409   | 32    | 74.06%   | 2.39%     | 23.81% |
| FIBR_JELUD      | 13    | 1176  | 453   | 58    | 69.94%   | 2.79%     | 18.31% |
| A_V_BLOK        | 14    | 1243  | 400   | 43    | 73.94%   | 3.38%     | 24.56% |
| OTEK_LANC       | 55    | 1167  | 374   | 104   | 71.88%   | 12.82%    | 34.59% |
| RAZRIV          | 17    | 1233  | 413   | 37    | 73.53%   | 3.95%     | 31.48% |
| DRESSLER        | 20    | 1186  | 439   | 55    | 70.94%   | 4.36%     | 26.67% |
| ZSN             | 126   | 975   | 331   | 268   | 64.76%   | 27.57%    | 31.98% |
| REC_IM          | 41    | 1127  | 414   | 118   | 68.71%   | 9.01%     | 25.79% |
| P_IM_STEN       | 40    | 1169  | 383   | 108   | 71.12%   | 9.46%     | 27.03% |
| LET_IS_1        | 38    | 1196  | 394   | 72    | 72.59%   | 8.80%     | 34.55% |
| LET_IS_2        | 6     | 1271  | 411   | 12    | 75.12%   | 1.44%     | 33.33% |
| LET_IS_3        | 15    | 1246  | 400   | 39    | 74.18%   | 3.61%     | 27.78% |
| LET_IS_4        | 7     | 1238  | 439   | 16    | 73.24%   | 1.57%     | 30.43% |
| LET_IS_5        | 3     | 1249  | 439   | 9     | 73.65%   | 0.68%     | 25.00% |
| LET_IS_6        | 7     | 1249  | 424   | 20    | 73.88%   | 1.62%     | 25.93% |
| LET_IS_7        | 6     | 1223  | 450   | 21    | 72.29%   | 1.32%     | 22.22% |
| Total:          | 479   | 21596 | 7384  | 1141  | 72.14%   | 6.09%     | 29.57% |

7. Discussion and conclusions
As can be seen in table 1, the accuracy of solving the problem riches 84%. The data augmentation procedure provided a significant (almost one and a half times). The use of the auxiliary layers increased both accuracy and precision by nearly 6-7%.

The overall improvement in the forecast as a result of the entire set of activities is 19%. Improving the sensitivity metrics can be realized by enhancing the dataset and using more flexible augmentation methods.

References
[1] Townsend N, Wilson L, Bhatnagar P, Wickramasinghe K, Rayner M and Nichols M 2016 Cardiovascular disease in Europe: epidemiological update Eur. Heart J. 37 3232-45
[2] Anon 2015 The Demographic Yearbook of Russia. 2015: Statistical Handbook (Moscow: Rosstat)
[3] Boytsov S, Samorodskaya I, Starinskaya M, Semenov V and Kakorina E 2016 analysis of structure and dynamics of indicators of mortality from cardiovascular diseases in Russia in 2006 and 2014 Russ. J. Prev. Med. 19(2) 11-2
[4] Boytsov S, Deev A and Shalnova S 2017 Mortality and risk factors for non-communicable diseases in Russia: Specific features, trends, and prognosis Ter. Arkh. 89 5-13
[5] Baxt W G 1990 Use of an Artificial Neural Network for Data Analysis in Clinical Decision-Making: The Diagnosis of Acute Coronary Occlusion Neural Comput. 2 480-9
[6] Barrett L A, Payrovznizri S N, Bian J and He Z 2019 Building Computational Models to Predict One-Year Mortality in ICU Patients with Acute Myocardial Infarction and Post Myocardial Infarction Syndrome. AMIA Jt Summits Transl Sci Proc. 2019 May 6;2019:407-416. PMID: 31258994; PMCID: PMC6568079.
[7] Dorrer M G, Golovenkin S E, Nikulina S Y, Orlova Y V and Pelipeckaya E Y 2020 Using artificial intelligence to predict the human body’s response to cardiovascular disease J. Phys. Conf. Ser. 1679 42012

[8] Golovenkin, S.E.; Gorban, Alexander; Mirkes, Evgeny; Shulman, V.A.; Rossiev, D.A.; Shesternya, P.A, Nikulina S Y, Orlova Y V and Dorrer M G. 2020: Myocardial infarction complications Database. University of Leicester. Dataset. https://doi.org/10.25392/leicester.data.12045261.v3

[9] Bai Y, Yang K, Mei T, Ma W-Y and Zhao T 2018 Automatic Dataset Augmentation ACM Transactions on Multimedia Computing, Communications, and Applications Volume 14 Issue 3 August 2018 Article No.: 69pp 1–20https://doi.org/10.1145/3204941