УДК 004.048

V. Kondratiuk

ALGORITHM OF THE SOFTWARE MODULE FOR PRE-DIAGNOSIS OF PATIENTS BASED ON THE KOHONEN NEURAL NETWORK

Vinnytsia National Technical University

Анотація

У статті розглянуто алгоритм до медичного діагностування пацієнтів, що заснований на використанні показників загального аналізу крові. Досліdженно, що даний підхід реалізується за допомогою кластеризації даних нейронною мережею Кохонена.

Ключові слова: нейронна мережа Кохонена, алгоритм, нейрони, кластери.

Abstract

The article considers an algorithm for medical diagnosis of patients based on the use of indicators of general blood tests. It is investigated that this approach is implemented by clustering data with the Kohonen neural network.

Keywords: Kohonen network, algorithm, neurons, clusters.

Introduction

To implement this technique, it is appropriate to use data clustering, which will be able to ensure high accuracy of their processing, while having a simple software structure.

Method

The structure of the developed software module for pre-diagnosis of patients based on the Kohonen neural network consists of two layers of neurons. The Kohonen layer consists of a number of *n* parallel linear elements. They all have the same number of inputs *m* and will receive the same vector of input signals $x=(x_1 \ldots x_m)$. At the output of the j-th linear element we obtain a signal calculated by the formula:

$$y_j = w_{j0}\sum_{i=1}^{m} w_{ji}x_i$$

where w_{ji} - the weighting factor of the *i*-th input of the *j*-th neuron, w_{j0} - the threshold coefficient.

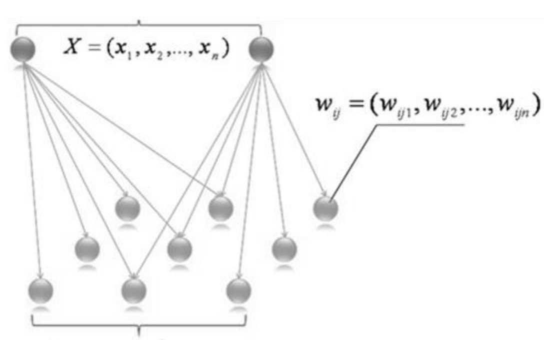

Figure 1. The structure of the Kohonen network of the developed software module

After passing the layer of linear elements, the signals are submitted for processing according to the rule "the winner takes everything": the maximum is searched among the output signals y_i; his number $j_{\max} = \arg\max_j \{y_j\}$. Finally, the output signal with the number $j_{\max}$ is one, and all the others - zero. If the maximum is reached simultaneously for several $j_{\max}$, then receive all the corresponding signals equal to one.

The training of the developed network is as follows. At the beginning of the work the number of clusters and their centers is determined. This information is predetermined: the number of clusters is 30, the initial centers of the clusters correspond to the sets of disease indicators for certain groups of diseases. After that, some input vector from the training set is selected and installed at the input of the neural network. At this stage, the differences between the input vector and all vectors are calculated by the formula:

$$D_{ij} = \left| X^l - W_{ij} \right| = \sqrt{(x_1 - w_{ij1})^2 + \ldots + (x_n - w_{ijn})^2}$$

where *i* and *j* - indicators of neurons in the source layer. After that, the neural network selects the winning neuron from a list of defined centers of clusters, ie such that its weight vector is similar to the input by the formula:

$$D(k_1, k_2) = \min_{i,j} D_{i,j}$$

where k_1 and k_2 are the indicators of the winning neuron. After that, the weight vectors of the winner and neighboring neurons are corrected. Close neurons to the winner are determined by the topological function of the neighborhood "Mexican Hat", which is calculated by the formula:

$$h(p,t) = \exp(-\frac{p^2}{\sigma^2(t)})(1 - \frac{2}{\sigma^2(t)} p^2)$$

where *p* - the distance to the winner's neuron, which is according to the formula:

$$p = \sqrt{(k_1 - i)^2 + (k_2 - j)^2}$$

where σ - the function that determines the radius of the neighborhood. At the beginning of the operation of the software module, it includes the entire space of the sensor field (grid), but over time its value decreases.

After calculating the topological function, the weights of all neurons are recalculated according to the formula:

$$W_{ij}(t+1) = W_{ij}(t) + \alpha(t) h(p,t)(X^l(t) - W_{ij}(t))$$

where $\alpha(t)$ - learning speed function, which also changes over time.

If the neuron is a winner or adjacent to it, its weight vector is updated or remains unchanged otherwise. At each step, the neural network identifies the neuron whose weight vector is most similar to the input vector and adjusts its weights and the weights of the neighbors to bring them closer to the input vector. (Figure 2)

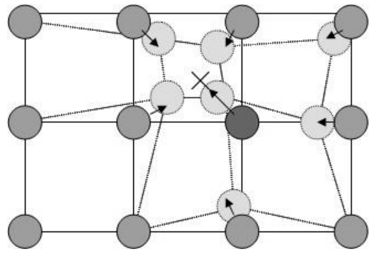

Figure 2. Update the winning neuron and its neighbors and "push" towards the input vector, which is marked "×" in the figure.

Each input vector from the training sample is presented to the neural network and training lasts either a fixed number of cycles, or until the difference between the input and weight vectors reaches a given value of ε. The difference between neighboring neurons decreases over time, and therefore they are organized into groups (clusters) that correspond to one of the classes in the learning set.

For optimal calculations of the proposed network used in the development of the software module, the procedure of pre-processing of input data was also introduced, in which the values of the features that form the input vector are reduced to a given range.

To implement the proposed approach to diagnosis, the normalization of input data was used according to the formula:

$$y = \frac{(x - x_{\min})(d_2 - d_1)}{x_{\max} - x_{\min}} + d_1$$

where: x - the value to be normalized;

$[d_1, d_2]$ - range of values x;

$[x_{\min}, x_{\max}]$ - the interval to which the value will be reduced x.

Conclusions

Thus, an algorithm was developed, according to which the software module of pre-diagnosis of patients based on the Kohonen neural network will work in the future.

REFERENCES

1. Dostal P. Cluster analysis and neural network / P. Dostal, P.Pokorny. 10 p. URL: http://dsp.vscht.cz/konference_matlab/MATLAB08/prispevky/025_dostal.pdf (accessed 03.02.2022).
2. Kohonen T. Self-Organizing Maps. Berlin, Heidelberg : Springer Berlin Heidelberg, 2001. URL: https://doi.org/10.1007/978-3-642-56927-2 (date of access: 01.02.2022).
3. Neural Networks. Berlin/Heidelberg : Springer-Verlag, 2005. URL: https://doi.org/10.1007/3-540-28847-3 (date of access: 03.02.2022).
4. Kohonen T. Self-organizing neural projections. *Neural Networks*. 2006. Vol. 19, no. 6-7. P. 723–733. URL: https://doi.org/10.1016/j.neunet.2006.05.001 (date of access: 05.02.2022).
5. Clusters / ed. by M. T. Nguyen, B. Kiran. Cham : Springer International Publishing, 2017. URL: https://doi.org/10.1007/978-3-319-48918-6 (date of access: 05.02.2022).

Науковий керівник: Мельник Олеся Дмитрівна — кандидат філологічних наук, доцент кафедри іноземних мов, Вінницький національний технічний університет, Вінниця.

Кондратюк Володимир Олексійович **-** 2КІ-21м Факультет Інформаційних технологій та комп'ютерної інженерії.

Supervisor: Melnyk Olesya Dmytrivna — candidate of Philological Sciences, Associate Professor at the Department of Foreign Languages, Vinnytsia National Technical University, Vinnytsia.

Kondratiuk Volodymyr - Faculty of Information Technology and Computer Engineering.