European Proceedings of
Computers and Technology
EpCT

www.europeanproceedings.com

e-ISSN: 2672-8834

DOI: 10.15405/epct.23021.36

HMMOCS 2022
International Workshop "Hybrid methods of modeling and optimization in complex systems"

AN EFFICIENT TRAINING ALGORITHM OF RESTRICTED BOLTZMANN MACHINES

V. V. Matskevich (a)*, V. A. Stasiuk (b)
*Corresponding author

(a) Belarusian State University, 4, Nezavisimosti av., Minsk, Belarus, matskevich1997@gmail.com
(b) Siberian Federal University, 26a, ul. Akademik Kirenskii, Krasnoyarsk, Russia, vstasyuk@sfu-kras.ru

Abstract

The paper deals with an actual applied problem related to the artificial neural networks training. An approach to the solution based on the idea of random search is proposed. An original training algorithm that implements Boltzmann annealing has been developed and its convergence in probability to the global optimum has been proved. It is also shown that the proposed algorithm can be easily modified to train any artificial neural network. Thus, it has a good prospect for solving applied problems using neural network technologies in general. Experimental studies have been carried out, in which, using the example of compressing color raster images problem, the proposed algorithm was compared with the known adaptive moment algorithm - one of the best gradient methods for training neural networks. Image compression was performed using an ensemble of n Gauss-Bernoulli restricted Boltzmann machines. The use of an ensemble of n machines in combination with a specially developed parallelization procedure made it possible to reduce the computational complexity of the training process and increase the speed of the proposed algorithm. As a result of experiments, it was shown that the proposed approach is not inferior to gradient methods in terms of speed. Moreover, the developed training algorithm turned out to be more than twice as effective as the adaptive moment algorithm in terms of the quality of the solution obtained.

Keywords: Annealing method, gradient decent method, training, neural networks, restricted Boltzmann machine

https://doi.org/10.15405/epct.23021.36
Corresponding Author: V. V. Matskevich
Selection and peer-review under responsibility of the Organizing Committee of the conference
eISSN: 2672-8834

1. Introduction

Modern society is moving from the post-industrial to the informational stage of its development. Huge arrays of numerical data are generated daily, which stimulates the development of information and computer technologies (Chen et al., 2022). Recently, neural network technology for data processing has become widespread. The technology is based on a flexible mathematical model - an artificial neural network, with the help of which a wide range of applied problems is solved. To tune a neural network to a specific subject area, it is necessary to train it. Training is a typical optimization problem, where the objective function is given, which describes the solution quality, and the data on which it is necessary to achieve the optimal functional value. At the initial stage of the neural networks development for training, as a rule, the gradient approach was used. This approach has become widespread due to the high convergence rate of the methods that implement it (Nakamura et al., 2021). However, with the development of computer technology, the situation has changed cardinally and this factor has ceased to be a determining factor. This, in turn, made it possible to develop other approaches to training.

The paper considers an alternative approach to training neural networks based on the idea of random search. An algorithm is proposed that implements one of the variants of the annealing method, and its efficiency is studied using the example of solving the color images compression problem.

2. Problem Statement

Consider the problem of color images compression, which can be described as follows.

Let a bitmap color image be given by a set of color pixels (containing red, blue and green colors). In this case, any of the colors is set to a value from 0 to 255, that is, it is an 8-bit number. In this case, can be represented a single pixel as a Cartesian product of three 8-bit sets, and an image as a Cartesian product of 8-bit sets. As a result of image compression in the bit space, a certain vector of minimum dimension k in a certain sense should be constructed.

The problem of color images compression can be formally written as follows:

Let a set of color images X be given. It is necessary to construct an algorithm such that:

$$\begin{cases} A:\{0,...,255\}^{3N} \to \{0,1\}^k; \\ A^{-1}:\{0,1\}^k \to \{0,...,255\}^{3N}; \\ k \to \min; \\ \sum\limits_{x\in X}\sum\limits_{i=1}^{3N}(x_i - y_i)^2 \to \min, \end{cases}$$

where x is the original image; y is an image restored from a compressed image; N is the number of pixels in the image.

Inverse mapping A^{-1} may not be unique, but it is required to compress images as much as possible ($k \to \min$), but square error across all images set X should not be greater than some threshold T, which is set manually.

Lower-dimensional representations can improve performance on many tasks, such as image compression, reconstruction and clustering (Liu et al., 2019). Color image compression occurs in many information retrieval problems (Dewi et al., 2021; Khanna et al., 2019; Knop et al., 2016). In such tasks,

https://doi.org/10.15405/epct.23021.36
Corresponding Author: V. V. Matskevich
Selection and peer-review under responsibility of the Organizing Committee of the conference
eISSN: 2672-8834

the restricted Boltzmann machine (RBM) is often used, which allows you to extract informative features that are used later for image classification.

3. Research Questions

In course of the study the following questions were raised:

- Is it possible to build efficient RBM training algorithm implementing annealing method?
- What is the efficiency of the algorithm based on the annealing method in relation to the gradient descent algorithm?

4. Purpose of the Study

The answers to the issues raised above will help achieve the goal and in future to contribute to the development of neural network training algorithms. It should increase a solution quality in a wide range of applied problems.

5. Research Methods

RBM refers to models (architectures) of recurrent neural networks. The parameters that define the properties of the connections between the neurons of the layers are called weights, and the parameters of the neurons are called their characteristics. Characteristics depend on the type of distribution that the network generates. For Gauss-Bernoulli-type RBMs, often used for data compression, neurons have three characteristics. For the hidden and visible layers, the displacements of the neurons *HB* and *B* are set, respectively, as well as the parameters of the dispersion of neurons σ. As a result, the architecture of the RBM *P* can be described by four types of parameters: the sets of weights *W*, the bias of the visible and hidden words *B* and *HB*, and the variances of neurons in the visible layer σ. That is, RBM can be formally described as a parametric family $P=(W, B, HB, \sigma)$. By fixing certain elements in each of the sets, you can set different subtypes of architectures.

Consider a RBM containing n_1 neurons in the input layer and n_2 neurons in the hidden layer. Then arbitrary parameters' values $x=(x_1,x_2,\ldots x_{n1*n2+2n1+n2})$ $x_i \in \Re$, $i=1,\ldots,n_1n_2+2n_1+n_2$ define a specific version of the this type neural network.

It is proposed to use an approach based on the idea of a random search to train the RBM. The optimization problem in this case can be formulated as follows.

Let Ω be the set of feasible solutions. In the case of neural networks of the described above type, it can be represented as a Cartesian product of subsets of admissible parameter values for each group of network parameters D_i (the Gauss-Bernoulli RBM has four groups of parameters). In this case, for each group of parameters, the set of feasible values is determined by the formula

$$D_i=[a_i;b_i]^{k_i},$$

where k_i is the number of parameters in the group; a_i, b_i – upper and lower bounds of values in the group.

https://doi.org/10.15405/epct.23021.36
Corresponding Author: V. V. Matskevich
Selection and peer-review under responsibility of the Organizing Committee of the conference
eISSN: 2672-8834

Suppose that on the set of feasible solutions Ω the objective function F is defined, and for each element $x \in \Omega$ there is a set of neighboring elements $N(x) \subset \Omega$. Then the optimization problem can be formally specified as a triple (Ω, F, N). A set of neighboring elements determines the optimization algorithm. Almost all random search theory was build on several restrictions on this set.

Let us consider the possibility of solving the problem described above using Boltzmann annealing.

For this variant of the annealing method, the sequence T_0, T_1, T_2,… is specified, the elements of which are interconnected by the relation:

$$T_k = \frac{T_0}{\ln(k+2)}, k>0 \tag{1}$$

Using these values, the transition probability from the current solution x to the new solution y is determined. The probability is determined by the formula:

$$P(y|x)=\min\left(1,\exp\left(\frac{F(x)-F(y)}{T_k}\right)\right) \tag{2}$$

An algorithm that implements Boltzmann annealing is proposed (Kirkpatrick et al., 1983).

Preliminary stage. Parameters initialization.

Step 0. Setting initial values for problem parameters $x=(x_1,x_2,\ldots x_{n1n2+2n1+n2})$ and temperature T_0.

General k-th iteration.

Step 1. Four random variables are generated n_1, n_2, n_3, n_4 according to the formula

$$n_i = \lfloor R[0;m_i] \rfloor, i=\overline{1,4}$$

where R[a;b] is a realization of a uniformly distributed random variable on the segment $[a;b]$ $a,b \in \Re$; m_i is number of parameters in each group. Values n_1, n_2, n_3, n_4 are amount of parameters to change.

Step 2. Random permutations are generated Q_1, Q_2, Q_3, Q_4 m_1, m_2, m_3, m_4 length respectively. First n_1, n_2, n_3, n_4 elements specify the indexes of the parameters to be changed in each of the parameter groups.

Let, for example, $J_1=\{x_{q11},x_{q12},\ldots,x_{q1n1}\}$, $J_2=\{x_{q21},x_{q22},\ldots,x_{q2n2}\}$, $J_3=\{x_{q31},x_{q32},\ldots,x_{q3n3}\}$, $J_4=\{x_{q41},x_{q42},\ldots,x_{q4n4}\}$, $J_1,J_2,J_3,J_4 \subseteq \Omega$, – sets of changing parameters.

Step 3. New solution $y=(y_1,y_2,\ldots y_{n1*n2+2n1+n2})$ is generating according to the formula:

$$y_k=\begin{cases} y_k, y_k \notin (J_1 \cup J_2 \cup J_3 \cup J_4) \\ y_k + a_{ik}, y_k \in J_i, i=\overline{1,4} \\ a_{ik}=R[-l_i;l_i], i=\overline{1,4}, \\ k=\overline{1,n_1n_2+2n_1+n_2} \end{cases}$$

where l_1, l_2, l_3, l_4 – algorithm parameters. These parameters determine the set of neighboring elements size. The set of neighboring elements size is critical. It influences on convergence speed and final solution quality.

Step 4. Calculating of the objective function value $F(y)$.

Step 5. A new solution is chosen according to the probability value (2).

https://doi.org/10.15405/epct.23021.36
Corresponding Author: V. V. Matskevich
Selection and peer-review under responsibility of the Organizing Committee of the conference
eISSN: 2672-8834

Step 6. Checking the optimality criterion. If the time for training has expired, then the algorithm terminates, otherwise the value of k is increased by one and go to Step 1.

It is easy to show that the developed training algorithm is correct, in the sense that the algorithm does not make transitions to invalid solutions.

An important characteristic of this type of algorithm is the convergence property. For this annealing method variant, the convergence theorem is known.

Theorem 1 (Hajek, 1988).

For any non-local minimum x

$$\lim_{k\to+\infty} P(x_k = x) = 0$$

2. If B is the set of local minima of depth d, then for any x from B

$$\lim_{k\to+\infty} P(x_k \in B) = 0$$

if and only if when

$$\sum_{k=1}^{+\infty} \exp(-d / T_k) = +\infty$$

3. Let Ω* be the set of global minima and d* be the maximum from the depths of local minima that do not match with any of the global

$$\lim_{k\to+\infty} P(x_k \in \Omega^k) = 1$$

if and only if when

$$\sum_{k=1}^{+\infty} \exp(-d^* / T_k) = +\infty \quad (3)$$

According to (Hajek, 1988) theorem, the algorithm must satisfy the following conditions:

- a new solution must be chosen with probability (2);
- problem (Ω, F, N) must be irreducible and have the weak reversibility property;
- temperature sequence (1) must be decreasing and converging to zero;
- the generator of new solutions must have the reversibility property.

The convergence conditions were formulated in the form of statements and proved (Krasnoproshin & Matskevich, 2022).

Statement 1. Problem (Ω, F, N) is irreducible and has the weak reversibility property.

Statement 2. The temperature sequence (1) monotonic decreasing to zero and satisfies the constraints (3).

Note. The proposed training algorithm can be easily modified for neural networks of any architecture. To do this, you need to determine the number of network parameter groups and set the parameters of the algorithm from step 3.

https://doi.org/10.15405/epct.23021.36
Corresponding Author: V. V. Matskevich
Selection and peer-review under responsibility of the Organizing Committee of the conference
eISSN: 2672-8834

It should be noted that this algorithm allows the use of parallelization procedures, including those developed earlier to speed up the training of neural networks.

The theoretical guarantee of convergence makes it possible to obtain a global optimum, but when solving an applied problem, it is important to know how fast the algorithm converges. Therefore, to study the problem and study the effectiveness of the proposed approach, experimental studies were carried out using the example of solving the color images compression problem.

The well-known dataset CIFAR-10 (2020), which contains 60000 color raster images with a resolution of 32x32 pixels, was chosen as experimental data. Each image contains exactly an object belonging to one of the ten classes. Images, in addition to the object, contain an additional background, which greatly complicates compression.

For experimental studies, 8-fold, 16-fold and 32-fold compression ratios were chosen. Higher compression ratios lead to excessive losses, and lower compression ratios have no practical application in neural networks.

An ensemble of n RBMs was used as a compression tool. For 8-fold compression, 256 Gauss-Bernoulli RBMs were used. 128 machines were used for 16x compression, and 64 machines for 32x compression.

One of the best modifications of the gradient method, the adaptive moment algorithm (Hamis et al., 2019), was used as a comparison algorithm. To estimate the gradient, there are many modifications: CD-N (Li et al., 2021), PCD (Oswin et al., 2018), N-PT-M (Brugge et al., 2013). To speed up training by the gradient method and achieve maximum quality (by mean square error), the CD-1 algorithm was chosen.

Training and validation were carried out on 4000 images. The remaining 52,000 images were used as a test set to check the quality of the resulting solution. To measure the quality, the functionals MSE, PSNR, PSNR_HVS, SSIM (Temel & AlRegib, 2019) were used.

The experiments were carried out on a computer with the operating system Lubuntu 20.04, with 4 core CPU intel i7-4770k with 16 GB 1600 MHz RAM and GPU nvidia rtx 3070 with 5888 cores. The training time was measured using the gettimeofday function.

The results are presented in the Table 1.

Table 1. Training results for a restricted Boltzmann machine

Training algorithm	Compress ratio, bit/pixel	Training time, h	MSE	PSNR	PSNR_HVS	SSIM
adaptive moment algorithm	3	1.5	2345	14.7	14.9	0.401
	1.5	1	2531	14.3	14.6	0.330
	0.75	0.5	2992	13.5	13.8	0.179
proposed algorithm	3	1	376	22.5	22.6	0.794
	1.5	1	518	21.1	21.3	0.737
	0.75	1	722	19.7	19.9	0.650

6. Findings

Based on the experimental results, a conclusion can be drawn. Thanks to the use of a special parallelization procedure, it was possible to achieve parity in performance with gradient methods. In terms

of the MSE functionality (the lower the value, the better), the proposed algorithm more than doubled the quality of the opponent, in terms of the PSNR and PSNR_HVS functionals (the higher the value, the better) by about 50%, and in terms of the SSIM functionality (more is better), more than twice.

7. Conclusion

The paper proposes an approach to training neural networks based on random search and develops an original algorithm that implements Boltzmann annealing. The algorithm convergence in probability to the global optimum is proved, and it is shown that the proposed algorithm can be easily modified to train any artificial neural network.

As part of experimental studies, using the example of the problem of compressing color bitmap images, the proposed algorithm was compared with the adaptive moment algorithm.

As a result of experiments, it was shown that the proposed approach is not inferior to gradient methods in terms of speed. Moreover, according to various metrics, the developed algorithm turned out to be more efficient than the adaptive moment algorithm in terms of the quality of the solution obtained.

Thus, the proposed training approach, based on the idea of random search, has good prospects for solving applied problems using neural network technologies in general.

Acknowledgments

This work was supported by the Ministry of Science and Higher Education of the Russian Federation (Grant № 075-15-2022-1121).

References

Brugge, K., Fischer, A., & Igel, C. (2013). The flip-the-state transition operator for restricted Boltzmann machines. *Machine Learning*, *93*(1), 53-69. https://doi.org/10.1007/s10994-013-5390-3

Chen, H., He, X., Yang, H., Qing, L., & Teng, Q. (2022). A Feature-Enriched Deep Convolutional Neural Network for JPEG Image Compression Artifacts Reduction and its Applications. *IEEE Transactions on Neural Networks and Learning Systems, 33*(1), 430-444. https://doi.org/10.1109/tnnls.2021.3124370

CIFAR-10 dataset. (2020). Retrieved on 04 March 2022, from https://www.cs.toronto.edu/~kriz/cifar.html

Dewi, C., Chen, R.-C., Hendry, & Hung, H.-T. (2021). Experiment Improvement of Restricted Boltzmann Machine Methods for Image Classification. *Vietnam Journal of Computer Science, 08*(03), 417-432. https://doi.org/10.1142/s2196888821500184

Temel, D., & AlRegib, G. (2019). Perceptual image quality assessment through spectral analysis of error representations. *Signal Processing: Image Communication, 70*, 37-46. https://doi.org/10.1016/j.image.2018.09.005

Hajek, B. (1988). Cooling Schedules for Optimal Annealing. *Mathematics of Operations Research*, *13*(2), 311-329. https://doi.org/10.1287/moor.13.2.311

Hamis, S., Zaharia, T., & Rousseau, O. (2019, June). Image compression at very low bitrate based on deep learned super-resolution. In *2019 IEEE 23rd International Symposium on Consumer Technologies (ISCT)* (pp. 128-133). IEEE. https://doi.org/10.1109/isce.2019.8901038

Khanna, M. T., Ralekar, C., Goel, A., Chaudhury, S., & Lall, B. (2019). Memorability-based image compression. *IET Image Processing*, *13*, 1490-1501. https://doi.org/10.1049/iet-ipr.2018.6097

Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science, 220*(4598), 671-680. https://doi.org/10.1126/science.220.4598.671

Knop, M., Kapurscirnski, T., Mleczko, W. K., & Angryk, R. (2016). *Neural Video Compression Based on RBM Scene Change Detection Algorithm*. Artificial Intelligence and Soft Computing, Springer, 660-669. https://doi.org/10.1007/978-3-319-39384-1_58

Krasnoproshin, V. V., & Matskevich, V. V. (2022). Random search in neural networks training. *Proceedings of the 13-th International Conference "Computer Data Analysis and Modeling" – CDAM'2022*, Minsk, 96-99.

Li, X., Gao, X., & Wang, C. (2021). A Novel Restricted Boltzmann Machine Training Algorithm With Dynamic Tempering Chains. *IEEE ACCESS*, *9*, 21939-21950. https://doi.org/10.1109/access.2020.3043599

Liu, W., Meng, F. Y., Liang, Y. S., Yang, H. X., & Wang, C. W. (2019). Loss Function Optimization Based on Adversarial Networks. *In Fuzzy Systems and Data Mining V* (pp. 619-634). IOS Press. https://doi.org/10.3233/FAIA190230

Nakamura, K., Derbel, B., Won, K.-J., & Hong, B.-W. (2021). Learning-Rate Annealing Methods for Deep Neural Networks. *Electronics*, *10*(16), 2029. MDPI AG. https://doi.org/10.3390/electronics10162029

Oswin, K., Fischer, A., & Igel, C. (2018). Population-Contrastive-Divergence: Does consistency help with RBM training? *Pattern Recognition Letters*, *102*, 1-7. https://doi.org/10.48550/arXiv.1510.01624