GOLOVKO V.A., MIKHNO E.V., BRICH A.L., MIKHNYAEV A.L., VOYTSEKHOVICH L.Yu., MATYUSHKOV A.L. The reduced svertochny neural network for exact recognition of hand-written figures

The convolutional neural network for accurate handwritten digit recognition is considered. In this work we have shown, that high accuracy can be achieved using reduced shallow convolutional neural network without adding distortions for digits. The main contribution of this paper is to point out how using simplified convolutional neural network is to obtain test error rate 0.71% on the MNIST handwritten digit benchmark. It permits to reduce computational resources in order to model convolutional neural network.

УДК 004.89

Головко В.А., Крощенко А.А., Хацкевич М.В.

ТЕОРИЯ ГЛУБОКОГО ОБУЧЕНИЯ: КОНВЕНЦИАЛЬНЫЙ И НОВЫЙ ПОДХОД

Введение. ***Глубокие нейронные сети*** (deep neural networks) представляют собой нейронные сети с множеством слоев нейронных элементов. Существуют следующие глубокие нейронные сети (DNN):

- нейронные сети глубокого доверия (deep belief neural networks);
- глубокий персептрон (deep perceptron);
- глубокая сверточная нейронная сеть (deep convolutional neural networks);
- глубокая рекуррентная нейронная сеть (deep recurrent neural networks);
- глубокий автоэнкодер (deep autoencoder);
- глубокая рекуррентная-сверточная нейронная сеть (deep R-CNN).

Исторически, первыми появились нейронные сети глубокого доверия и глубокий персептрон, которые в общем случае представляют собой многослойный персептрон с более чем двумя скрытыми слоями [4]. Основным отличием нейронной сети глубокого доверия от глубокого персептрона является то, что нейронная сеть глубокого доверия не является в общем случае сетью с прямым распространением сигнала (feed forward neural network). До 2006 года в научной среде была приоритетной парадигма, что многослойный персептрон с одним, максимум двумя скрытыми слоями является более эффективным для нелинейного преобразования входного пространства образов в выходное по сравнению с персептроном с большим количеством скрытых слоев. Считалось, что персептрон с более чем двумя скрытыми слоями не имеет смысла применять. Данная парадигма базировалась на теореме, что персептрон с одним скрытым слоем является универсальным аппроксиматором. Другой аспект этой проблемы заключается в том, что все попытки использовать алгоритм обратного распространения ошибки (backpropagation algorithm) для обучения персептрона с тремя и более скрытыми слоями не приводили к улучшению решения различных задач. Это связано с тем, что алгоритм обратного распространения ошибки является неэффективным для обучения персептронов с тремя и более скрытыми слоями при использовании сигмоидальной функции активации. Это происходит из-за проблемы ***исчезающего градиента*** (vanishing gradient problem). Так, например, максимальное значение производной сигмоидной функции активации $F(S_j)$ равно 0,25. Поэтому использование обобщенного дельта-правила для обучения персептрона с большим количеством скрытых слоев приводит к затуханию градиента при распространении сигнала от последнего слоя к первому. В 2006 Хинтон (Hinton) предложил «жадный» алгоритм послойного обучения (greedy layer-wise algorithm) [1], который стал эффективным средством обучения глубоких нейронных сетей. Было показано, что глубокая нейронная сеть имеет большую эффективность нелинейного преобразования и представления данных по сравнению с традиционным персептроном. Такая сеть осуществляет глубокое иерархическое преобразование входного пространства образов. В результате первый скрытый слой выделяет низкоуровневое пространство признаков входных данных, второй слой детектирует пространство признаков более высокого уровня абстракции и т. д. [2].

2. Архитектура глубокой нейронной сети. Как уже отмечалось, глубокая нейронная сеть содержит множество скрытых слоев нейронных элементов (рис. 1) и осуществляет глубокое иерархическое преобразование входного пространства образов.

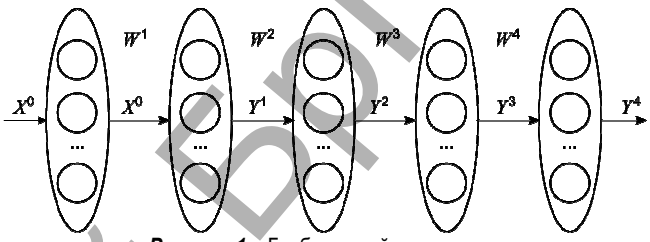


Рисунок 1 – Глубокая нейронная сеть

Выходное значение *j*-го нейрона *k*-го слоя определяется следующим образом:

$$y_j^k = F\left(S_j^k\right), \tag{1}$$

$$S_j^k = \sum_{i=1} w_{ij}^k y_i^{k-1} + T_j^k, \tag{2}$$

где F – функция активации нейронного элемента, S_j^k – взвешенная сумма *j*-го нейрона *k*-слоя, w_{ij}^k – весовой коэффициент между *i*-м нейроном (*k*-1)-го слоя и *j*-м нейроном *k*-го слоя, T_j^k – пороговое значение *j*-го нейрона *k*-го слоя.

Для первого (распределительного) слоя

$$y_i^0 = x_i. \tag{3}$$

В матричном виде выходной вектор *k*-го слоя

$$Y^k = F\left(S^k\right) = F\left(W^k Y^{k-1} + T^k\right), \tag{4}$$

где W – матрица весовых коэффициентов, Y^{k-1} – выходной вектор (*k*-1)-го слоя, T^k – вектор пороговых значений нейронов *k*-го слоя.

Если глубокая нейронная сеть используется для классификации образов, то выходные значения сети часто определяются на основе функции активации softmax:

$$y_j^F = soft\max\left(S_j\right) = \frac{e^{S_j}}{\sum_l e^{S_l}}.$$

Несмотря на архитектурные различия глубоких нейронных сетей, ***принципы их обучения являются идентичными***. Поэтому рассмотрим основные концепции обучения таких сетей на примере глубокого персептрона.

3. Конвенциальные методы обучения глубоких нейронных сетей. Рассмотрим обучение глубоких нейронных сетей. Существуют два основных метода обучения:

1. ***Метод с предварительным обучением***, который состоит из двух этапов:

Крощенко Александр Александрович, *старший преподаватель кафедры интеллектуальных информационных технологий Брестского государственного технического университета.*

Хацкевич М.В., *ст. преподаватель кафедры интеллектуальных информационных технологий Брестского государственного технического университета.*

Беларусь, БрГТУ, 224017, г. Брест, ул. Московская, 267.

- предобучение нейронной сети методом послойного обучения, начиная с первого слоя (pre-training). Данное обучение осуществляется без учителя и базируется на ограниченной машине Больцмана (RBM);
- настройка синаптических связей всей сети (fine-tuning) при помощи алгоритма обратного распространения ошибки или алгоритма «бодрствования и сна» (wake-sleep algorithm).

2. ***Метод стохастического градиента (SGD***) с ректификационной функцией активации (ReLU) нейронных элементов.

В настоящее время принята следующая парадигма для обучения глубоких нейронных сетей. Если обучающая выборка является большой, то есть размерность обучающей выборки намного больше, чем количество настраиваемых параметров сети, то используется метод стохастического градиента (SGD) с функцией активации ReLU нейронных элементов. Если размерность обучающей выборки сравнима с количеством настраиваемых параметров сети, то применяется предварительное обучение нейронной сети на основе RBM и алгоритм обратного распространения ошибки для точной настройки синаптических связей сети (fine-tuning).

Важным этапом обучения глубоких нейронных сетей является предобучение слоев нейронной сети. Существует два основных подхода к предварительному обучению слоев глубоких нейронных сетей (рис. 2).

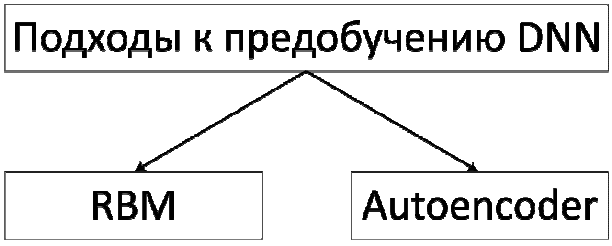


Рисунок 2 – Методы предварительного обучения глубоких нейронных сетей

Первый подход называется автоэнкодерным и базируется на представлении каждого слоя в виде автоассоциативной нейронной сети. Второй подход базируется на представлении каждого слоя нейронной сети в виде ограниченной машины Больцмана (RBM).

3.1 Автоэнкодерный метод обучения. Данный подход базируется на представлении каждого слоя в виде ***автоассоциативной нейронной сети.*** В этом случае, вначале обучается первый слой как автоассоциативная нейронная сеть с целью минимизации суммарной квадратичной ошибки реконструкции информации, затем второй и так далее. Для обучения каждого слоя можно использовать алгоритм обратного распространения ошибки. После этого осуществляется точная настройка синаптических связей всей сети (fine tuning), используя алгоритм обратного распространения ошибки.

Рассмотрим персептрон с тремя скрытыми слоями (рис. 3). Тогда в соответствии с автоэнкодерным методом, прежде всего, берутся первые два слоя нейронной сети (1 и 2) и на базе их конструируется автоассоциативная (PCA сеть) нейронная сеть (1-2-1), то есть добавляется восстанавливающий слой (рис. 4). Затем происходит обучение, например, при помощи алгоритма обратного распространения ошибки такой сети с целью минимизации квадратичной ошибки реконструкции информации. Продолжительность обучения обычно составляет не больше чем 100 эпох.

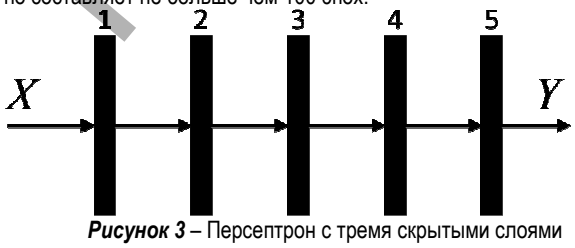


Рисунок 3 – Персептрон с тремя скрытыми слоями

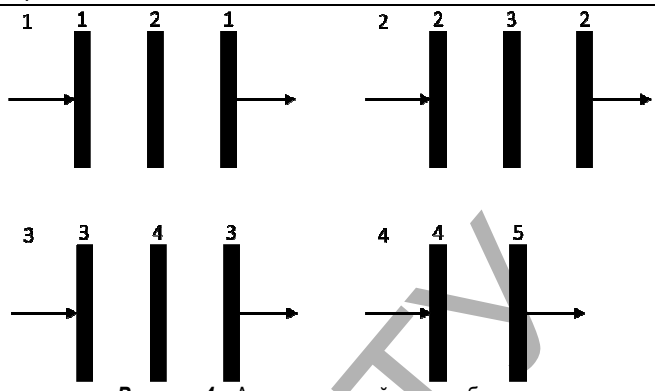


Рисунок 4 – Автоэнкодерный метод обучения

После этого отбрасывается восстанавливающий слой (последний слой), фиксируются веса скрытого слоя, и конструируется автоассоциативная сеть из следующих двух слоев нейронной сети (2-3-2), которая обучается на основе данных, поступающих с предыдущего (2-го слоя). Процесс продолжается до последнего или предпоследнего слоя, как это схематично изображено на рис. 4. В результате послойного обучения получается предварительно обученная нейронная сеть. Далее осуществляется точная настройка (fine tuning) посредством, например, алгоритма обратного распространения ошибки с учителем.

Данный процесс можно представить в виде следующего алгоритма:

1. Конструируется автоассоциативная сеть с входным слоем X, скрытым Y и выходным слоем X.
2. Обучается автоассоциативная сеть, например, при помощи алгоритма обратного распространения ошибки (как правило, не более 100 эпох) и фиксируются синаптические связи первого слоя W^1.
3. Берется следующий слой и формируется автоассоциативная сеть аналогичным образом.
4. Используя настроенные синаптические связи предыдущего слоя W^1, подаем входные данные на вторую автоассоциативную сеть и обучаем ее аналогичным образом. В результате получаются весовые коэффициенты второго слоя W^2.
5. Процесс продолжается до последнего слоя нейронной сети.
6. Обучается вся сеть для точной настройки параметров при помощи алгоритма обратного распространения ошибки.

3.2 Обучение глубокой нейронной сети на основе RBM. Как уже отмечалось, данный подход базируется на представлении каждого слоя нейронной сети в виде ***ограниченной машины Больцмана*** (RBM). Ограниченная машина Больцмана состоит из двух слоев стохастических бинарных нейронных элементов, которые соединены между собой двунаправленными симметричными связями (рис. 5). Входной слой нейронных элементов называется видимым (слой X), а второй слой называется скрытым (слой Y). Глубокую нейронную сеть можно представить как совокупность ограниченных машин Больцмана. Ограниченная машина Больцмана может аппроксимировать (генерировать) любое дискретное распределение, если используется достаточное количество нейронов скрытого слоя [9].

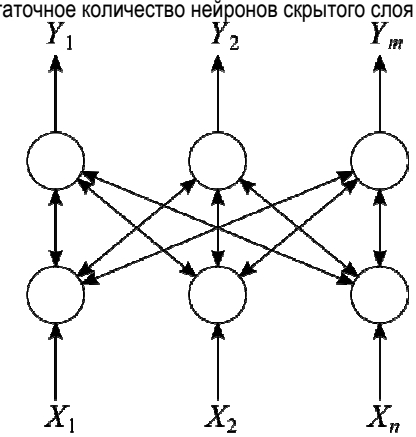


Рисунок 5 – Ограниченная машина Больцмана

Данная сеть является ***стохастической нейронной сетью***, в которой состояния видимых и скрытых нейронов меняются в соответствии с вероятностной версией сигмоидной функции активации:

$$p(y_i|x)=\frac{1}{1+e^{-S_j}},\quad S_j=\sum_{i=1}^{n}w_{ij}x_i+T_j, \tag{5}$$

$$p(x_i|x)=\frac{1}{1+e^{-S_i}},\quad S_i=\sum_{j=1}^{m}w_{ij}y_j+T_i. \tag{6}$$

Состояния видимых и скрытых нейронных элементов принимаются независимыми:

$$P(x|y)=\prod_{i=1}^{n}P(x_i|y);$$

$$P(y|x)=\prod_{j=1}^{m}P(y_i|x).$$

Таким образом, состояния всех нейронных элементов ограниченной машины Больцмана определяются через распределение вероятностей. В RBM нейроны скрытого слоя являются детекторами признаков, которые выделяют закономерности входных данных. Основная задача обучения состоит в воспроизведении распределения входных данных на основе состояний нейронов скрытого слоя как можно точнее. Это эквивалентно максимизации функции правдоподобия путем модификации синаптических связей нейронной сети. Рассмотрим это подробнее.

Вероятность нахождения видимого и скрытого нейрона в состоянии (x, y) определяется на основе распределения Гиббса:

$$P(x|y)=\frac{e^{-E(x,y)}}{Z},$$

где $E(x, y)$ – энергия системы в состоянии (x, y), Z – параметр, который определяет условие нормализации вероятностей, то есть, чтобы сумма вероятностей равнялась единице. Данный параметр определяется следующим образом:

$$Z=\sum_{x,y}e^{-E(x,y)}.$$

Вероятность нахождения видимых нейронов в определенном состоянии равняется сумме вероятностей конфигураций $P(x, y)$ по состояниям скрытых нейронов:

$$P(x)=\sum_{y}P(x,y)=\sum_{y}\frac{e^{-E(x,y)}}{Z}=\frac{\sum_{y}e^{-E(x,y)}}{\sum_{x,y}e^{-E(x,y)}}.$$

Для нахождения правила модификации синаптических связей необходимо максимизировать вероятность воспроизведения состояний видимых нейронов $P(x)$ ограниченной машиной Больцмана. Для того, чтобы определить максимум функции правдоподобия распределения данных $P(x)$, будем использовать метод градиентного спуска в пространстве весовых коэффициентов и пороговых значений сети, где в качестве градиента применим функцию логарифмического правдоподобия:

$$\ln P(x)=\ln\sum_{y}e^{-E(x,y)}-\ln\sum_{x,y}e^{-E(x,y)},$$

тогда градиент равен

$$\frac{\partial\ln(P(x))}{\partial\omega_{ij}}=\frac{\partial}{\partial\omega_{ij}}\left(\ln\sum_{y}e^{-E(x,y)}\right)-\frac{\partial}{\partial\omega_{ij}}\left(\ln\sum_{x,y}e^{-E(x,y)}\right).$$

Преобразуя последнее выражение, получим

$$\frac{\partial\ln P(x)}{\partial\omega_{ij}}=-\frac{1}{\sum_{y}e^{-E(x,y)}}\sum_{y}e^{-E(x,y)}\frac{\partial E(x,y)}{\partial\omega_{ij}}+$$

$$+\frac{1}{\sum_{x,y}e^{-E(x,y)}}\sum_{x,y}e^{-E(x,y)}\frac{\partial E(x,y)}{\partial\omega_{ij}}.$$

Так как

$$P(x,y)=P(y|x)P(x),$$

то

$$P(y|x)=\frac{P(x,y)}{P(x)}=\frac{\frac{1}{Z}e^{-E(x,y)}}{\frac{1}{Z}\sum_{y}e^{-E(x,y)}}=\frac{e^{-E(x,y)}}{\sum_{y}e^{-E(x,y)}}.$$

В результате можно получить следующее выражение:

$$\frac{\partial\ln P(x)}{\partial\omega_{ij}}=-\sum_{y}P(y|x)\frac{\partial E(x,y)}{\partial\omega_{ij}}+\sum_{x,y}P(x,y)\frac{\partial E(x,y)}{\partial\omega_{ij}}.$$

В данном выражении первое слагаемое определяет позитивную фазу работы машины Больцмана, когда сеть работает на основе образов из обучающей выборки. Второе слагаемое характеризует негативную фазу функционирования, когда сеть работает в свободном режиме независимо от окружающей среды.

Рассмотрим энергию сети RBM. С точки зрения энергии сети задача обучения состоит в том, чтобы на основе входных данных найти конфигурацию выходных переменных с минимальной энергией. В результате, на обучающем множестве сеть будет иметь меньшую энергию по сравнению с другими состояниями. Функция энергии бинарного состояния (x, y) определяется аналогично сети Хопфилда:

$$E(x,y)=-\sum_{i}x_iT_i-\sum_{j}y_jT_j-\sum_{i,j}x_iy_i\omega_{ij}. \tag{7}$$

В этом случае

$$\frac{\partial E(x,y)}{\partial\omega_{ij}}=-x_iy_j,$$

$$\frac{\partial\ln P(x)}{\partial\omega_{ij}}=\sum_{y}P(y|x)x_iy_i-\sum_{x,y}P(x,y)x_iy_i.$$

Так как математическое ожидание равняется:

$$E(x)=\sum_{i}x_iP_i,$$

то

$$\frac{\partial\ln P(x)}{\partial\omega_{ij}}=E[x_iy_i]_{data}-E[x_iy_i]_{model}.$$

Аналогичным образом можно получить градиенты для пороговых значений:

$$\frac{\partial\ln P(x)}{\partial T_i}=E[x_i]_{data}-E[x_i]_{model},$$

$$\frac{\partial\ln P(x)}{\partial T_j}=E[y_j]_{data}-E[y_j]_{model}.$$

Как следует из последних выражений, первое слагаемое характеризует работу сети на основе данных из обучающей выборки, а второе слагаемое характеризует работу сети на основе данных модели (данные генерируемые сетью), то есть в свободном режиме независимо от окружающей среды.

Так как вычисление математического ожидания на основе RBM сети является очень сложным, Хинтон предложил использовать аппроксимацию данных слагаемых, которую он назвал контрастным расхождением (contrastive divergence (CD)) [1].

Такая аппроксимация основывается на ***сэмплировании Гиббса*** (Gibbs sampling). В этом случае первые слагаемые в выражениях для градиента характеризуют распределение данных в момент времени t=0, а вторые слагаемые характеризуют реконструированные или генерируемые моделью состояния в момент времени t=k. Исходя из этого, CD-k процедура может быть представлена следующим образом:

$$x(0)\to y(0)\to x(1)\to y(1)\to\ldots.\to x(k)\to y(k). \tag{8}$$

В результате, можно получить следующие правила для обучения RBM сети. В случае применения CD-1, k=1 и учитывая, что в соответствии с методом градиентного спуска

$$\omega_{ij}(t+1)=\omega_{ij}(t)+\alpha\frac{\partial \ln P(x)}{\partial \omega_{ij}(t)}.$$

Можно получить для последовательного обучения, что

$$\begin{aligned}\omega_{ij}(t+1)&=\omega_{ij}(t)+\alpha\left(x_i(0)y_j(0)-x_i(1)y_j(1)\right),\\ T_i(t+1)&=T_i(t)+\alpha\left(x_i(0)-x_i(1)\right),\\ T_j(t+1)&=T_j(t)+\alpha\left(y_j(0)-y_j(1)\right).\end{aligned}\quad(9)$$

Аналогичным образом, для алгоритма CD-*k*

$$\begin{aligned}\omega_{ij}(t+1)&=\omega_{ij}(t)+\alpha\left(x_i(0)y_j(0)-x_i(k)y_j(k)\right),\\ T_i(t+1)&=T_i(t)+\alpha\left(x_i(0)-x_i(k)\right),\\ T_j(t+1)&=T_j(t)+\alpha\left(y_j(0)-y_j(k)\right).\end{aligned}\quad(10)$$

В случае группового обучения и CD-*k*

$$\begin{aligned}\omega_{ij}(t+1)&=\omega_{ij}(t)+\alpha\sum_{l=1}^{L}\left(x_i^l(0)y_j^l(0)-x_i^l(k)y_j^l(k)\right),\\ T_j(t+1)&=T_j(t)+\alpha\sum_{l=1}^{L}\left(y_j^l(0)-y_j^l(k)\right),\\ T_i(t+1)&=T_i(t)+\alpha\sum_{l=1}^{L}\left(x_i^l(0)-x_i^l(k)\right).\end{aligned}\quad(11)$$

Из последних выражений видно, что правила обучения ограниченной машины Больцмана минимизируют разницу между оригинальными данными и данными, генерируемыми моделью. Генерируемые моделью данные получаются при помощи сэмплирования Гиббса.

Обучение нейронной сети глубокого доверия происходит на основе «жадного» алгоритма послойного обучения (greedy layer-wise algorithm). В соответствии с ним вначале обучается первый слой сети как RBM машина. Для этого входные данные поступают на видимый слой нейронных элементов и, используя CD-*k* процедуру, вычисляются состояния скрытых $p(y|x)$ и видимых нейронов $p(x|y)$. В процессе выполнения данной процедуры (не более 100 эпох) изменяются весовые коэффициенты и пороговые значения RBM сети, которые затем фиксируются. Затем берется второй слой нейронной сети и конструируется RBM машина. Входными данными для нее являются данные с предыдущего слоя. Происходит обучение, и процесс продолжается для всех слоев нейронной сети, как показано на рис. 6 [11]. В результате такого обучения без учителя можно получить подходящую начальную инициализацию настраиваемых параметров глубокой нейронной сети. На заключительном этапе осуществляется точная настройка параметров всей сети при помощи алгоритма обратного распространения ошибки или алгоритма «бодрствования и сна» (wake-sleep algorithm).

3.3 Метод стохастического градиента с использованием ReLU. Ректификационная функция активации (ReLU) применяется, как правило, во всех слоях глубокой нейронной сети за исключением последнего слоя. В случае использования данной функции активации (рис. 7) для обучения глубокой нейронной сети необязательно использовать предобучение слоев нейронных элементов. В этом случае можно использовать стандартный алгоритм обратного распространения ошибки для обучения сети. Метод градиентного спуска для последовательного или группового обучения, если используются группы образов небольшого размера (minibatch), и происходит случайный выбор образов из обучающей выборки, называется методом ***стохастического градиентного спуска*** (stochastic gradient descent, SGD).

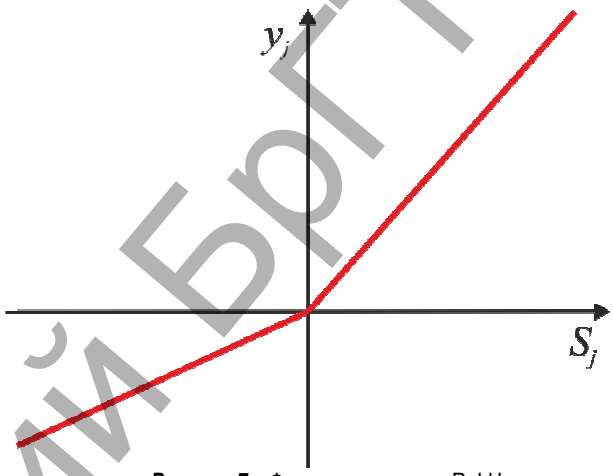


Рисунок 7 – Функция активации ReLU

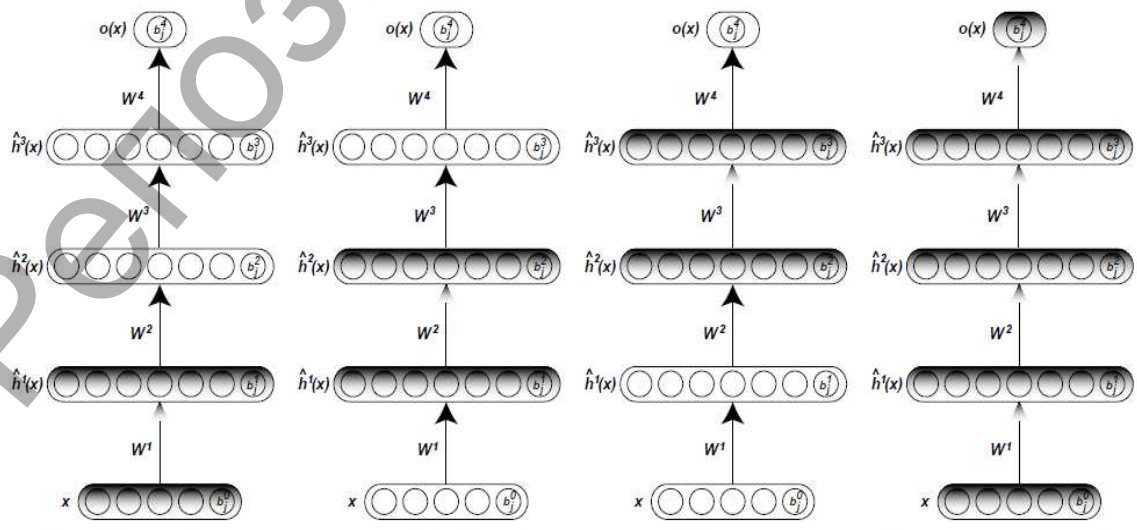
В случае использования функции активации ReLU выходное значение нейронного элемента

$$y_j=F(S_j)=\begin{cases}S_j, S_j>0;\\ kS_j, S_j\le 0,\end{cases}\quad(12)$$

где k=0 или принимает небольшое значение, например, k=0,01 или k=0,001.

Тогда

$$\frac{\partial y_j}{\partial S_j}=F(S_j)=\begin{cases}1, S_j>0;\\ k, S_j\le 0.\end{cases}\quad(13)$$



(a) First hidden layer pre-training (b) Second hidden layer pre-training (c) Third hidden layer pre-training (d) Fine-tuning of whole network

Рисунок 6 – Жадный алгоритм послойного обучения (Greedy layer-wise algorithm)

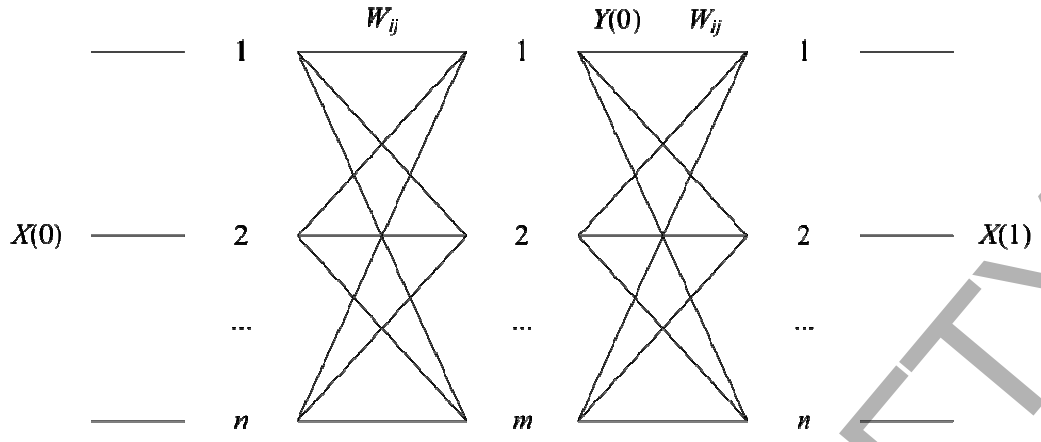


Рисунок 8 – Представление RBM в виде автоэнкодерной сети

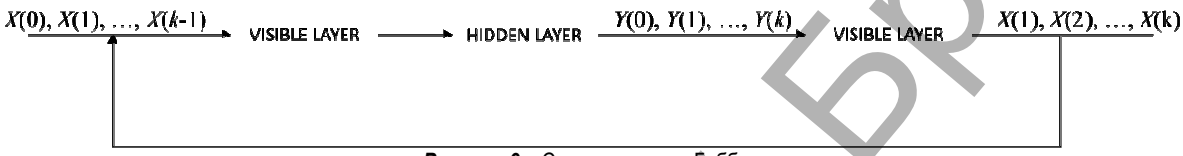


Рисунок 9 – Сэмплирование Гиббса

Используя обобщенное дельта правило, можно осуществить настройку синаптических связей глубокой нейронной сети. Почему при использовании ректификационной функции активации эффективно использование алгоритма обратного распространения ошибки для обучения глубоких нейронных сетей? Как следует из выражения (13), в области положительных значений взвешенной суммы производная ректификационной функции активации равняется единице, что позволяет нейтрализовать проблему исчезающего градиента.

4. Альтернативный подход к глубокому обучению. В данном разделе рассматривается альтернативный взгляд на ограниченную машину Больцмана как автоассоциативную нейронную сеть, которая может функционировать с любыми данными, как бинарными, так и числовыми. Предлагаются новые методы для получения правила обучения ограниченной машины Больцмана [4, 14–18]. Первый метод базируется на минимизации ошибки реконструкции видимых и скрытых образов, которую можно получить, используя простые итерации сэмплирования Гиббса. По сравнению с традиционным подходом, основанным на энергии методе (energy-based method), который базируется на линейном представлении нейронных элементов, предложенный метод позволяет учитывать нелинейную природу нейронных элементов. Второй метод базируется на мнимизации кросс-энтропии видимых и скрытых образов.

Рассмотрим ограниченную машину Больцмана, которую будем представлять в виде трех слоев нейронных элементов [16]: видимый, скрытый и видимый (рис. 8). Такое представление RBM эквивалентно автоэнкодерной нейронной сети, где скрытый и последний видимый слой являются соответственно сжимающим и восстанавливающим слоями.

Процесс сэмплирования Гиббса заключается в следующей процедуре. Пусть $x(0)$ входной вектор, который поступает на видимый слой в момент времени 0. Тогда выходные значения нейронов скрытого слоя:

$$y_j(0) = F\left(S(0)\right), \tag{14}$$

$$S_j(0) = \sum_i \omega_{ij} x_i(0) + T_j. \tag{15}$$

Инверсный (последний) слой реконструирует входной вектор на основе данных со скрытого слоя. В результате получается восстановленный вектор $x(1)$ в момент времени 1:

$$x_i(1) = F\left(S_i(1)\right), \tag{16}$$

$$S_i(1) = \sum_j \omega_{ij} y_j(0) + T_i. \tag{17}$$

Затем вектор $x(1)$ поступает на видимый слой, и вычисляются выходные значения нейронов скрытого слоя:

$$y_j(1) = F\left(S_j(1)\right), \tag{18}$$

$$S_j(1) = \sum_i \omega_{ij} x_i(1) + T_j. \tag{19}$$

Продолжая данный процесс, можно получить на шаге k

$$x_i(k) = F\left(S_i(k)\right),$$

$$S_i(k) = \sum_j \omega_{ij} y_j(k-1) + T_i.$$

$$y_i(k) = F\left(S_j(k)\right),$$

$$S_j(k) = \sum_j \omega_{ij} x_i(k) + T_j.$$

Существуют различные методы обучения RBM сети, которые базируются на использовании разных целевых функций обучения. Как отмечалось ранее, G. Hinton предложил модель, основанную на энергии (energy-based model), которая базируется на максимизации функции логарифмического правдоподобия распределения входных данных $P(x)$. В данном разделе предлагается использовать другие целевые функции. Первый критерий базируется на ***минимизации суммарной квадратичной ошибки сети*** (MSE), а второй – на ***минимизации кросс-энтропии сети*** (CE). Покажем, что использование различных критериев обучения приводит к идентичным правилам обучения.

4.1 Минимизация MSE. Целью обучения ограниченной машины Больцмана является минимизация суммарной квадратичной ошибки реконструкции данных на скрытом и восстанавливающем слое. Суммарная квадратичная ошибка на скрытом слое пропорциональна разнице между выходными значениями нейронов скрытого слоя в различные моменты времени и в случае CD-k определяется следующим образом:

$$E_h(k)=\frac{1}{2}\sum_{l=1}^{L}\sum_{j=1}^{m}\sum_{p=1}^{k}\left(y_j^l(p)-y_j^l(p-1)\right)^2. \quad (20)$$

Аналогичным образом квадратичная ошибка инверсного слоя в разнице между выходными значениями нейронов выходного слоя в различные моменты времени:

$$E_v(k)=\frac{1}{2}\sum_{l=1}^{L}\sum_{i=1}^{n}\sum_{p=1}^{k}\left(x_i^l(p)-x_i^l(p-1)\right)^2, \quad (21)$$

где L – размерность обучающей выборки.

Тогда общая квадратичная ошибка реконструкции данных на скрытом и восстанавливающем слое определяется как сумма соответствующих ошибок:

$$E_s(k)=E_h(k)+E_v(k). \quad (22)$$

Отсюда

$$E_s(k)=\frac{1}{2}\sum_{l=1}^{L}\sum_{j=1}^{m}\sum_{p=1}^{k}\left(y_j^l(p)-y_j^l(p-1)\right)^2+ \quad (23)$$
$$+\frac{1}{2}\sum_{l=1}^{L}\sum_{i=1}^{n}\sum_{p=1}^{k}\left(x_i^l(p)-x_i^l(p-1)\right)^2.$$

Аналогичным образом для CD-1 суммарная квадратичная ошибка:

$$E_s(1)=\frac{1}{2}\sum_{l=1}^{L}\sum_{j=1}^{m}\left(y_j^l(1)-y_j^l(0)\right)^2+ \quad (24)$$
$$+\frac{1}{2}\sum_{l=1}^{L}\sum_{i=1}^{n}\left(x_i^l(1)-x_i^l(0)\right)^2.$$

Теорема 1. Максимизация функции правдоподобия распределения данных $P(x)$ в пространстве синаптических связей ограниченной машины Больцмана эквивалентна минимизации суммарной квадратичной ошибки сети в том же пространстве при использовании линейных нейронов.

Доказательство. Рассмотрим последовательное обучение RBM, когда модификация синаптических связей происходит после подачи каждого входного образа на сеть. В соответствии с методом градиентного спуска для минимизации суммарной квадратичной ошибки сети, синаптические связи должны изменяться следующим образом:

$$\omega_{ij}(t+1)=\omega_{ij}(t)-\alpha\frac{\partial E}{\partial\omega_{ij}(t)},$$
$$T_i(t+1)=T_i(t)-\alpha\frac{\partial E}{\partial T_i(t)},$$
$$T_j(t+1)=T_j(t)-\alpha\frac{\partial E}{\partial T_j(t)}.$$

В случае CD-k квадратичная ошибка E для одного образа:

$$E=\frac{1}{2}\sum_{j=1}^{m}\sum_{p=1}^{k}\left(y_j(p)-y_j(p-1)\right)^2+.$$
$$+\frac{1}{2}\sum_{i=1}^{n}\sum_{p=1}^{k}\left(x_i(p)-x_i(p-1)\right)^2.$$

Тогда

$$\frac{\partial E}{\partial\omega_{ij}}=\frac{\partial E}{\partial y_j(p)}\cdot\frac{\partial y_j(p)}{\partial S_j(p)}\cdot\frac{\partial S_j(p)}{\partial\omega_{ij}}+$$
$$+\frac{\partial E}{\partial x_i(p)}\cdot\frac{\partial x_i(p)}{\partial S_i(p)}\cdot\frac{\partial S_i(p)}{\partial\omega_{ij}}=$$
$$=\sum_{p=1}^{k}\left(y_j(p)-y_j(p-1)\right)x_i(p)F'\left(S(p)\right)+$$
$$+\sum_{p=1}^{k}\left(x_i(p)-x_i(p-1)\right)y_j(p-1)F'\left(S_i(p)\right).$$

Если ограниченная машина Больцмана использует линейные нейроны (линейная функция активации), то

$$F'\left(S_j(p)\right)=\frac{\partial S_j(p)}{\partial w_{ij}}=F'\left(S_i(p)\right)=\frac{\partial S_i(p)}{\partial w_{ij}}=1.$$

Тогда

$$\frac{\partial E}{\partial w_{ij}}=\sum_{p=1}^{k}y_j(p)x_i(p)-y_i(p-1)x_i(p-1)=$$
$$=y_j(k)x_i(k)-y_j(0)x_i(0).$$

В результате можно получить CD-k правило обучения RBM:

$$\omega_{ij}(t+1)=\omega_{ij}(t)+\alpha\left(x_i(0)y_j(0)-x_i(k)y_j(k)\right).$$

Аналогичным образом для пороговых значений:

$$T_j(t+1)=T_j(t)+\alpha\left(y_j(0)-y_j(k)\right),$$
$$T_i(t+1)=T_i(t)+\alpha\left(x_i(0)-x_i(k)\right).$$

Как видно, последние выражения совпадают с классическим правилом обучения ограниченной машины Больцмана для CD-k. Отсюда следует, что для линейной RBM максимизация функции правдоподобия распределения данных $P(x)$ эквивалентна минимизации суммарной квадратичной ошибки сети. Теорема доказана.

Таким образом, природа классического правила обучения RBM сети является линейной с точки зрения минимизации MSE. Поэтому назовем такую машину линейной RBM.

Следствие 1. Для линейной ограниченной машины Больцмана правило модификации синаптических связей в случае CD-1 будет следующим:

$$\omega_{ij}(t+1)=\omega_{ij}(t)+\alpha\left(x_i(0)y_j(0)-x_i(1)y_j(1)\right),$$
$$T_i(t+1)=T_i(t)+\alpha\left(x_i(0)-x_i(1)\right),$$
$$T_j(t+1)=T_j(t)+\alpha\left(y_j(0)-y_j(1)\right).$$

Следствие 2. Линейная ограниченная машина Больцмана с точки зрения обучения эквивалентна автоэнкодерной нейронной сети при использовании в ней при обучении сэмплирования Гиббса.

Следствие 3. Для нелинейной ограниченной машины Больцмана правило модификации синаптических связей в случае CD-k будет следующим:

$$\omega_{ij}(t+1)=\omega_{ij}(t)-\alpha\left(\sum_{p=1}^{k}\left(y_j(p)-y_j(p-1)\right)x_i(p)F'\left(S_j(p)\right)+\right.$$
$$\left.+\left(x_i(p)-x_i(p-1)\right)y_j(p-1)F'\left(S_i(p)\right)\right),$$
$$T_j(t+1)=T_j(t)+\alpha\left(\sum_{p=1}^{k}\left(y_j(p)-y_j(p-1)\right)F'\left(S_j(p)\right)\right),$$
$$T_i(t+1)=T_i(t)+\alpha\left(\sum_{p=1}^{k}\left(x_i(p)-x_i(p-1)\right)F'\left(S_i(p)\right)\right).$$

Следствие 4. Для нелинейной ограниченной машины Больцмана правило модификации синаптических связей в случае CD-1 будет следующим:

$$\omega_{ij}(t+1)=\omega_{ij}(t)-\alpha\left(\left(y_j(1)-y_j(0)\right)F'\left(S_j(1)\right)x_i(1)+\right.$$
$$\left.+\left(x_i(1)-x_i(0)\right)F'\left(S_i(1)\right)y_j(0)\right),$$
$$T_i(t+1)=T_i(t)+\alpha\left(x_i(1)-x_i(0)\right)F'\left(S_i(1)\right),$$
$$T_j(t+1)=T_j(t)+\alpha\left(y_j(1)-y_j(0)\right)F'\left(S_j(1)\right).$$

Если используется групповое обучение (batch learning), то в этом случае метод градиентного спуска записывается следующим образом:

$$\omega_{ij}(t+1)=\omega_{ij}(t)-\alpha\frac{\partial E_s(k)}{\partial\omega_{ij}(t)},$$
$$T_i(t+1)=T_i(t)-\alpha\frac{\partial E_s(k)}{\partial T_i(t)},$$

$$T_j(t+1) = T_j(t) - \alpha \frac{\partial E_s(k)}{\partial T_j(t)}.$$

Теорема 2. При использовании CD-k для нелинейной ограниченной машины Больцмана в случае группового обучения правило модификации синаптических связей определяется на основе следующих выражений:

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \alpha \sum_{l=1}^{L} \sum_{p=1}^{k} \begin{pmatrix} \left(y_j^l(p) - y_j^l(p-1)\right) x_i^l(p) F'\left(S_j^l(p)\right) + \\ + \left(x_i^l(p) - x_i^l(p-1)\right) y_j^l(p-1) F'\left(S_i^l(p)\right) \end{pmatrix},$$

$$T_j(t+1) = T_j(t) - \alpha \sum_{l=1}^{L} \sum_{p=1}^{k} \left(y_j^l(p) - y_j^l(p-1)\right) F'\left(S_j^l(p)\right),$$

$$T_i(t+1) = T_i(t) - \alpha \sum_{l=1}^{L} \sum_{p=1}^{k} \left(x_i^l(p) - x_i^l(p-1)\right) F'\left(S_i^l(p)\right).$$

Здесь L – размер группы образов. Процесс доказательства данной теоремы является аналогичным доказательству теоремы 1.

Следствие 5. При использовании CD-1 для нелинейной ограниченной машины Больцмана в случае группового обучения правило модификации синаптических связей определяется на основе следующих выражений:

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \alpha \sum_{l=1}^{L} \left(\left(y_j^l(1) - y_j^l(0)\right) x_i^l(1) F'\left(S_j^l(1)\right) + \right.$$

$$\left. \left(x_i^l(1) - x_i^l(0)\right) y_j^l(0) F'\left(S_i^l(1)\right) \right),$$

$$T_j(t+1) = T_j(t) - \alpha \sum_{l=1}^{L} \left(y_j^l(1) - y_j^l(0)\right) F'\left(S_j^l(1)\right),$$

$$T_i(t+1) = T_i(t) - \alpha \sum_{l=1}^{L} \left(x_i^l(1) - x_i^l(0)\right) F'\left(S_i^l(1)\right).$$

Следствие 6. При использовании CD-k для линейной ограниченной машины Больцмана в случае группового обучения правило модификации синаптических связей определяется на основе следующих выражений:

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \alpha \sum_{l=1}^{L} \left(x_i^l(0) y_j^l(0) - x_i^l(k) y_j^l(k)\right),$$

$$T_j(t+1) = T_j(t) - \alpha \sum_{l=1}^{L} \left(y_j^l(0) - y_j^l(k)\right),$$

$$T_i(t+1) = T_i(t) - \alpha \sum_{l=1}^{L} \left(x_i^l(0) - x_i^l(k)\right).$$

Следствие 7. При использовании CD-1 для линейной ограниченной машины Больцмана в случае группового обучения правило модификации синаптических связей определяется на основе следующих выражений:

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \alpha \sum_{l=1}^{L} \left(x_i^l(0) y_j^l(0) - x_i^l(1) y_j^l(1)\right),$$

$$T_j(t+1) = T_j(t) - \alpha \sum_{l=1}^{L} \left(y_j^l(0) - y_j^l(1)\right),$$

$$T_i(t+1) = T_i(t) - \alpha \sum_{l=1}^{L} \left(x_i^l(0) - x_i^l(1)\right).$$

Примечание: При использовании группового обучения обычно вторые слагаемые в выражениях для модификации синаптических связей делятся на количество образов L.

В данном разделе получены правила обучения для ограниченной машины Больцмана, которые базируются на минимизации квадратичной ошибки восстановления информации в скрытом и видимом слоях. Предложенный метод обучения позволяет учитывать нелинейную природу нейронных элементов с точки зрения минимизации суммарной квадратичной ошибки сети. Назовем его REBA (reconstruction error-based approach) [16]. Показано, что классические выражения для обучения ограниченной машины являются частным случаем предложенного метода. Доказана теорема об эквивалентности максимизации функции правдоподобия распределения входных данных $P(x)$ в пространстве синаптических связей и минимизации суммарной квадратичной ошибки сети в том же пространстве для линейной ограниченной машины Больцмана. Впервые приведенные выше выражения были получены в работах [4, 14, 15].

4.2 Минимизация кросс-энтропии

Кросс-энтропия (CE) может использоваться в качестве альтернативы целевой функции квадратичной ошибки (MSE). Рассмотрим ограниченную машину Больцмана, нейронные элементы которой используют сигмоидную функцию активации. Тогда целью обучения RBM является минимизация кросс-энтропии в скрытом и видимом слоях. В случае CD-k кросс-энтропия функции ошибки для инверсного (восстанавливающего) слоя определяется как

$$CE_v(k) = -\sum_{l=1}^{L} \left[\sum_{p=1}^{k} \sum_{i=1}^{n} \left(x_i^l(p-1) \log\left(x_i^l(p)\right) + \right. \right. \quad (25)$$

$$\left. \left. + \left(1 - x_i^l(p-1)\right) \log\left(1 - x_i^l(p)\right) \right) \right].$$

Аналогично определяется кросс-энтропия функции ошибки для скрытого слоя

$$CE_h(k) = -\sum_{l=1}^{L} \left[\sum_{p=1}^{k} \sum_{j=1}^{m} \left(y_j^l(p-1) \log\left(y_j^l(p)\right) + \right. \right. \quad (26)$$

$$\left. \left. + \left(1 - y_j^l(p-1)\right) \log\left(1 - y_j^l(p)\right) \right) \right].$$

Общая кросс-энтропия функции ошибки для RBM в случае CD-k определяется как сумма соответствующих ошибок:

$$CE_s(k) = CE_h(k) + CE_v(k). \quad (27)$$

Теорема 3. Максимизация функции правдоподобия распределения входных данных $P(x)$ в пространстве синаптических связей ограниченной машины Больцмана эквивалентна минимизации кросс-энтропии функции ошибки сети $CE_s(k)$ в том же пространстве.

Доказательство. Для упрощения доказательства теоремы рассмотрим кросс-энтропию для CD-1. Тогда кросс-энтропия функции ошибки сети для одного образа

$$CE(1) = -\sum_{i=1}^{n} \left(x_i(0) \log\left(x_i(1)\right) + \left(1 - x_i(0)\right) \log\left(1 - x_i(1)\right) \right) -$$

$$- \sum_{j=1}^{m} \left(y_j(0) \log\left(y_j(1)\right) + \left(1 - y_j(0)\right) \log\left(1 - y_j(1)\right) \right).$$

Тогда для последовательного обучения

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \alpha \frac{\partial CE(1)}{\partial \omega_{ij}(t)},$$

$$T_i(t+1) = T_i(t) - \alpha \frac{\partial CE(1)}{\partial T_i(t)},$$

$$T_j(t+1) = T_j(t) - \alpha \frac{\partial CE(1)}{\partial T_j(t)}.$$

Отсюда можно получить, что

$$\frac{\partial CE(1)}{\partial \omega_{ij}} = -\frac{x_i(0)}{x_i(1)} x_1(1)\left(1 - x_i(1)\right) y_j(0) +$$

$$+ \frac{1 - x_i(0)}{1 - x_i(1)} x_i(1)\left(1 - x_i(1)\right) y_j(0) - y_j(0)\left(1 - y_j(1)\right) x_i(1) +$$

$$+ \left(1 - y_j(0)\right) y_j(1) x_i(1) = -x_i(0)\left(1 - x_i(1)\right) y_j(0) +$$

$$+ \left(1 - x_i(0)\right) x_i(1) y_j(0) - y_j(0)\left(1 - y_j(1)\right) x_i(1) +$$

$$+ \left(1 - y_j(0)\right) y_j(1) x_i(1) = -x_i(0) y_j(0) + x_i(0) x_i(1) y_j(0) +$$

$$+ x_i(1) y_j(0) - x_i(0) x_i(1) y_j(0) - y_j(0) x_i(1) +$$

$$+ y_j(0) y_j(1) x_i(1) + y_j(1) x_i(1) - y_j(0) y_j(1) x_i(1) =$$

$$= x_i(1) y_j(1) - x_i(0) y_j(0).$$

Аналогично для пороговых значений

$$\frac{\partial CE(1)}{\partial T_i} = x_i(1) - x_i(0),$$

$$\frac{\partial CE(1)}{\partial T_j} = y_j(1) - y_j(0).$$

Теорема доказана.

Как следует из теоремы, классические правила обучения RBM сети могут быть получены более простым путем по сравнению с конвенциальным методом, основанным на энергии. Таким образом, используя минимизацию функции кросс-энтропии и сэмплирование Гиббса, можно получить классические выражения для обучения RBM.

Полученные результаты могут быть обобщены в виде следующей теоремы.

Теорема 4. Максимизация функции правдоподобия распределения входных данных $P(x)$ в пространстве синаптических связей ограниченной машины Больцмана эквивалентна минимизации кросс-энтропии функции ошибки сети и минимизации суммарной квадратичной ошибки сети в том же пространстве при использовании линейных нейронов.

$$\max\left(\ln P(x)\right) = \min\left(CE_s\right) = \min\left(E_s\right).$$

Из теоремы следует, что использование различных критериев обучения приводит к одинаковым правилам обучения. Поэтому природа неконтролируемого обучения (обучение без учителя) в RBM сети является идентичной при использовании различных целевых функций.

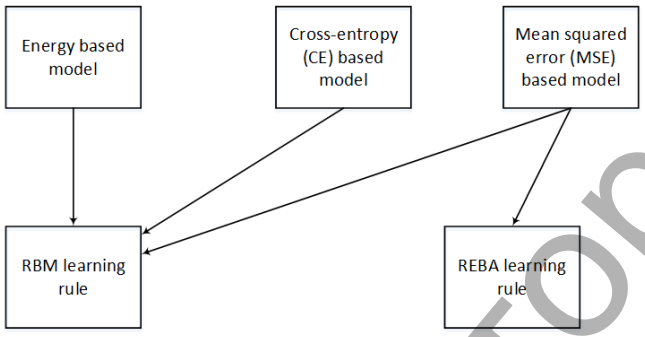


Рисунок 10 – Взаимосвязь между различными методами неконтролируемого обучения

Максимизация функции правдоподобия распределения входных данных и минимизация кросс-энтропии функции ошибки приводят к линейному представлению нейронных элементов с точки зрения минимизации MSE. Применение MSE в качестве целевой функции позволяет получить как линейные, так и нелинейные правила обучения, но не наоборот. Поэтому метод, основанный на минимизации суммарной квадратичной ошибки, является более универсальным и открывает новые возможности для неконтролируемого обучения в глубоких нейронных сетях. Впервые приведенный выше метод был предложен в работе [16]. Взаимосвязь между различными методами неконтролируемого обучения изображена на рис. 10.

5. Применение нейронных сетей глубокого доверия. Глубокие нейронные сети применяются для сжатия и визуализации данных, распознавания образов, обработки речи и т. д. Рассмотрим применение автоэнкодерных и пресептронных глубоких нейронных сетей для решения различных задач обработки информации.

5.1 Сжатие данных. Пусть дана система трех динамических уравнений [4], где параметр времени t генерируется в диапазоне [−1,1]:

$$\begin{cases} x_1 = \sin(\pi t) + \mu \\ x_2 = \cos(\pi t) + \mu \\ x_3 = t + \mu \end{cases}.$$

Здесь μ – гауссовский шум с нулевым средним и квадратичным отклонением, равным 0,05. Рассмотрим отображение входного трехмерного пространства данных в одномерное пространство при помощи глубокого автоэнкодера, который состоит из семи слоев нейронных элементов (рис. 11). Для обучения сети возьмем обучающую выборку, состоящую из 1000 тренировочных наборов. Тестирование сети проведем на данных, не входящих в обучающую выборку, количество которых равняется 1000 образов. В качестве функции активации нейронных элементов для всех слоев, кроме сжимающего, использовалась сигмоидная функция. Для сжимающего нейрона использовалась линейная функция активации. Для обучения каждого слоя нейронной сети использовалось 50 эпох, а для точной настройки параметров сети при помощи алгоритма обратного распространения ошибки использовалось 200 эпох.

Результаты экспериментов приведены в таблице 1. Здесь MSE – суммарная квадратичная ошибка на обучающей выборке, MS – квадратичная ошибка на тестовой выборке (ошибка обобщения), RBM – метод обучения на основе ограниченной машины Больцмана, REBA – предложенный метод.

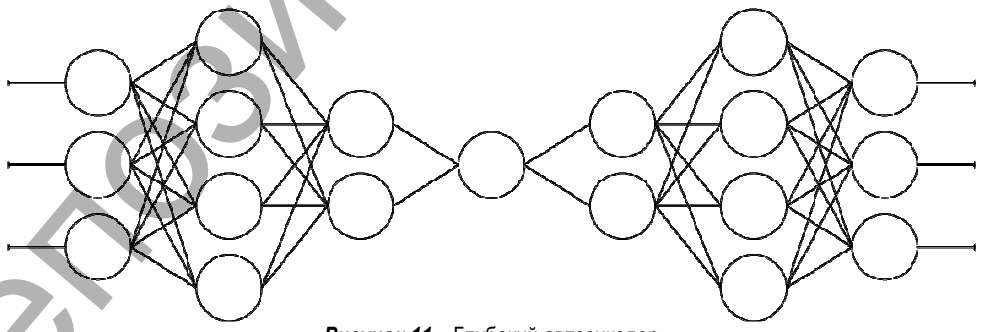


Рисунок 11 – Глубокий автоэнкодер

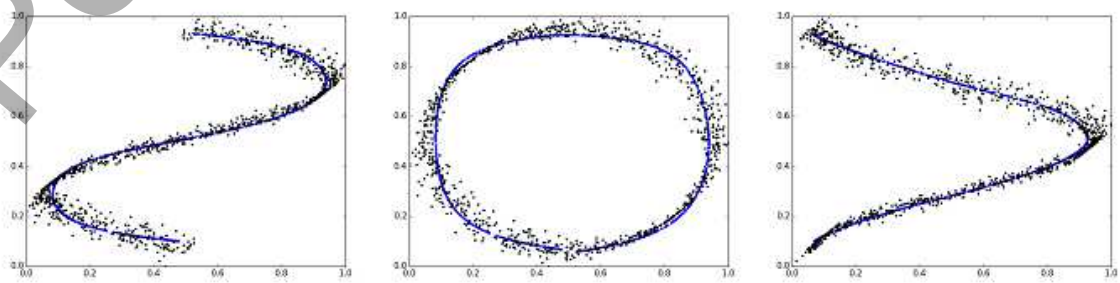


Рисунок 12 – Нелинейная ось первой главной компоненты в двумерном пространстве

Как следует из табл. 1, метод REBA является более эффективным по сравнению с традиционным подходом для CD-1 и CD-10.

На рис. 12 и 13 изображена нелинейная ось первой главной компоненты, на которую проецируется входное пространство образов.

Таблица 1

Метод обучения	CD-k	MSE	MS
RBM	1	**0,699**	**0,886**
	5	0,710	0,932
	10	**0,689**	**0,916**
	15	0,688	0,873
REBA	1	**0,673**	**0,851**
	5	0,719	0,966
	10	**0,677**	**0,907**
	15	0,700	0,895

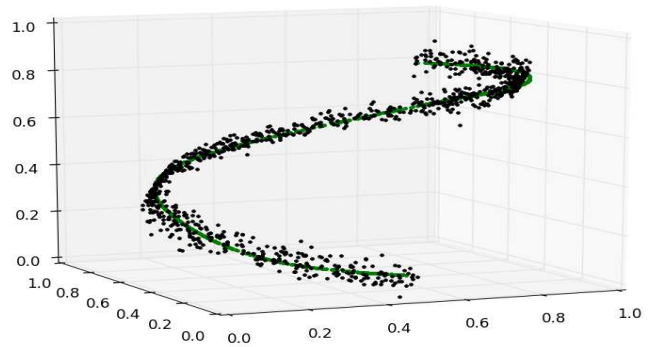


Рисунок 13 – Нелинейная ось первой главной компоненты в трехмерном пространстве

5.2. Визуализация данных. Рассмотрим визуализацию рукописных цифр с использованием глубокого автоэнкодера на основе базы данных MNIST. Она содержит 60000 образов рукописных цифр для обучения и 10000 образов для тестирования. Каждый образ представляет собой изображение 28×28 пикселей в градациях серого. Для отображения 784-мерных образов в двумерное пространство признаков будем использовать глубокий автоэнкодер с архитектурой 784-1000-500-250-2-250-500-1000-784. В среднем слое нейронной сети, который состоит из двух нейронов, применяется линейная функция активации. В остальных слоях используется сигмоидная функция активации. Для предварительного обучения глубокого автоэнкодера рассмотрим алгоритм послойного обучения на основе RBM и REBA методов. Данная процедура начинается с первого слоя и выполняется без учителя. После этого выполняется обучение всей нейронной сети, используя алгоритм обратного распространения ошибки. Для настройки синаптических связей сети использовались следующие параметры: скорость предварительного обучения равняется 0,2 для REBA и 0,05 для классического RBM метода для всех слоев, за исключением среднего слоя. Скорость обучения для среднего слоя равняется 0,001. Сравнительный анализ обоих методов представлен в табл. 2. Как следует из таблицы, предложенный подход REBA показывает лучшую обобщающую способность.

Таблица 2

Метод обучения	MSE	MS
RBM	3,7801	4,0115
REBA	3,6490	3,8726

Визуализация рукописных цифр, выполненная на основе REBA, представлена на рис. 14 для первых 500 тестовых изображений каждого класса.

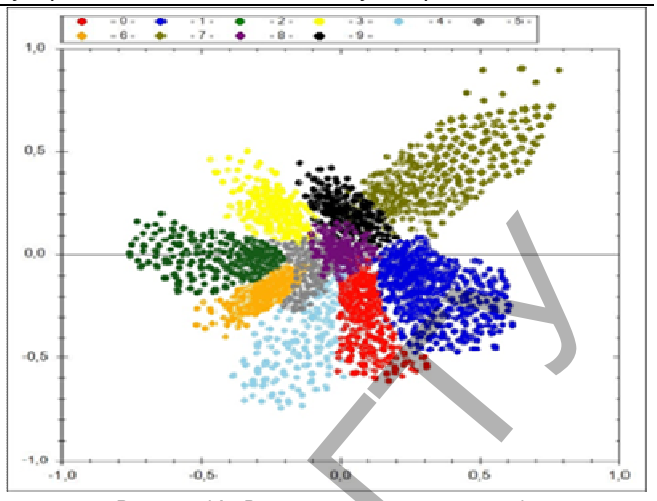


Рисунок 14 – Визуализация рукописных цифр

5.3 Классификация образов. Рассмотрим классификацию рукописных цифр с использованием базы данных MNIST. В качестве классификатора возьмем глубокий персептрон с архитектурой 784-500-500-2000-10 и сигмоидной функцией активации. Будем использовать следующую кодировку выходных классов. Будем использовать кодировку выходных классов на основе номера нейрона последнего слоя. Для этого необходимо определить номер k нейронного элемента, который имеет максимальное выходное значение:

$$y_k = \arg\max y_j .$$

Выходному значению нейрона с номером k присваивается единичное значение, а выходные значения остальных нейронных элементов равняются нулю:

$$y_j = \begin{cases} 1, \text{ j = k;} \\ 0, \textit{иначе}. \end{cases}$$

Для обучения использовался групповой метод со следующими параметрами: скорость обучения равняется 0,1 для REBA и 0,2 для классического RBM метода; размер группы (mini batch) составляет 100 образов; количество эпох предварительного обучения равняется 10; количество эпох алгоритма обратного распространения ошибки равняется 100; параметр регуляризации $\lambda = 0{,}00001$. Лучшие результаты были получены при использовании гибридного подхода, который представляет собой комбинацию REBA и RBM метода. Результаты экспериментов представлены в таблице 3.

Таблица 3

Метод обучения	MSE	MS	Ошибка тестирования (%)	NIT
RBM	6,178e-6	0,0235	1,23	10
Гибридный RBM+REBA (9+1)	5,962e-6	0,0224	**1,09**	9+1

Здесь NIT обозначает количество эпох предварительного обучения каждого слоя сети. Для гибридного подхода в процессе преобучения использовалось 9 эпох RBM и 1 эпоха REBA метода. Как следует из таблицы, ошибка тестирования в этом случае составляет 1,09 %.

Заключение. В данной статье рассматриваются и анализируются глубокие нейронные сети, которые считаются революционным шагом в области интеллектуальной обработки данных. Данные сети успешно применяются для решения различных проблем в области искусственного интеллекта, таких как обработка и распознавание речи, образов, естественного языка, визуализации данных и т. д.

Рассмотрены основные парадигмы обучения глубоких нейронных сетей (метод с предварительным обучением и метод стохастического градиента). Предложен новый метод для обучения ограниченной машины Больцмана и показано, что правило обучения ограниченной машины Больцмана является частным случаем предложенного мето-

да обучения, который базируется на минимизации суммарной квадратичной ошибки восстановления информации в скрытом и видимом слоях. Предложенный метод позволяет учитывать нелинейную природу нейронных элементов. Доказана теорема об эквивалентности максимизации функции правдоподобия распределения входных данных $P(x)$ в пространстве синаптических связей и минимизации суммарной квадратичной ошибки сети при использовании линейных нейронов, а также минимизации кросс-энтропийной функции ошибки сети в том же пространстве. Таким образом, подтверждается факт независимости природы обучения без учителя от выбора целевой функции. Рассматривается применение глубоких нейронных сетей для решения задач сжатия, визуализации и классификации образов на примере данных из базы MNIST.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Hinton, G. A fast learning algorithm for deep belief nets / G. Hinton, S. Osindero, Y. Teh // Neural Computation. – 2006. – Vol. 18. – P. 1527–1554.
2. Hinton, G. Reducing the dimensionality of data with neural networks / G. Hinton, R. Salakhutdinov // Science, 313 (5786). – 2006. – P. 504–507.
3. Hinton, G. A practical guide to training restricted Boltzmann machines // Tech. Rep. 2010-000. – Toronto: Machine Learning Group, University of Toronto, 2010.
4. Головко, В.А. От многослойных персептронов к нейронным сетям глубокого доверия: парадигмы обучения и применение / В.А. Головко // Лекции по Нейроинформатике. – М.: НИЯУ МИФИ, 2015. – С. 47–84.
5. Krizhevsky, A. ImageNet classification with deep convolutional neural networks / A. Krizhevsky, L. Sutskever, G. Hinton // In Proc. Advances in Neural information Processing Systems, 25. – 2012. – P. 1090–1098.
6. LeCun, Y. Deep learning / Y. LeCun, Y. Bengio, G. Hinton // Nature, 521 (7553). – 2015. – P. 436–444.
7. Mikolov, T. Strategies for training large scale neural network language models / T. Mikolov, A. Deoras, D. Povey, L. Burget, J. Cernocky // In Automatic Speech Recognition and Understanding. – 2011. – P. 195–201.
8. Hinton, G. Deep neural network for acoustic modeling in speech recognition / G. Hinton at all // IEEE Signal Processing Magazine, 29. – 2012. – P. 82–97.
9. Bengio, Y. Learning deep architectures for AI // Foundations and Trends in Machine Learning. – 2009. – Vol. 2(1). – P. 1–127.
10. Bengio, Y. Greedy layer-wise training of deep networks / Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle // In book Schölkopf, J. C. Platt, T. Hoffman (Eds.), Advances in neural information processing systems, 11. – MA: MIT Press, Cambridge, 2007. – P. 153–160.
11. Erhan, D. Why does unsupervised pre-training help deep learning? / D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, S. Bengio // Journal of Machine Learning Research. – 2010. – Vol. 11. – P. 625–660.
12. Larochelle H. Exploring strategies for training deep neural networks / H. Larochelle, Y. Bengio, J. Louradour, P. Lamblin // Journal of Machine Learning Research 1. – 2009. – P. 1–40.
13. Glorot, X. Deep sparse rectifier networks / X. Glorot, A. Bordes, Y. Bengio // In Proceedings of the 14th International Conference on Artificial Intelligence and Statistics. JMLR W&CP Volume. – 2011. – Vol. 15. – P. 315–323.
14. Golovko, V. A. Learning Technique for Deep Belief Neural Networks / V. Golovko, A. Kroshchanka, U. Rubanau, S. Jankowski // in book Neural Networks and Artificial Intelligence. – Springer, 2014. – Vol. 440. Communication in Computer and Information Science. – P. 136–146.
15. Golovko, V. A New Technique for Restricted Boltzmann Machine Learning / Aliaksandr Kroshchanka, Volodymyr Turchenko, Stanislaw Jankowski, Douglas Treadwell // Proceedings of the 8th IEEE International Conference IDAACS-2015. – Warsaw, 2015. – P.182–186.
16. Golovko, V. The Nature of Unsupervised Learning in Deep Neural Networks: A New Understanding and Novel Approach / Vladimir Golovko, Aliaksandr Kroshchanka, Douglas Treadwell // Optical Memory And Neural Networks (Springer Link). – 2016. – Vol. 25, № 3. – P. 127–141.
17. Golovko, V. Deep Neural Networks: A theory, application and new trends / V. Golovko // Proceedings of the 13-th International Conference on Pattern recognition and Information Processing. – Minsk: publishing Center of BSU, 2016. – P. 33–37.
18. Jankowski, S. Deep learning classifier based on NPCA and orthogonal feature selection // Stanisław Jankowski, Zbigniew Szymański, Uladzimir Dziomin, Vladimir Golovko, Aleksy Barcz / Proceedings of the International Conference on Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments. – 2016. – P. 5–9.

Материал поступил в редакцию 05.01.2017

GOLOVKO V.A., KROSHCHENKO A.A., KHATSKEVICH M.V. Theory of deep training: conventional and new approach

Over the last decade, the deep neural networks are a hot topic in machine learning. It is breakthrough technology in processing images, video, speech, text and audio. Deep neural network permits us to overcome some limitations of a shallow neural network due to its deep architecture. In this paper we investigate the nature of unsupervised learning in restricted Boltzmann machine. We have proved that maximization of the log-likelihood input data distribution of restricted Boltzmann machine is equivalent to minimizing the cross-entropy and to special case of minimizing the mean squared error. Thus the nature of unsupervised learning is invariant to different training criteria. As a result we propose a new technique called "REBA" for the unsupervised training of deep neural networks. In contrast to Hinton's conventional approach to the learning of restricted Boltzmann machine, which is based on linear nature of training rule, the proposed technique is founded on nonlinear training rule. We have shown that the classical equations for RBM learning are a special case of the proposed technique. As a result the proposed approach is more universal in contrast to the traditional energy-based model. We demonstrate the performance of the REBA technique using wellknown benchmark problem. The main contribution of this paper is a novel view and new understanding of an unsupervised learning in deep neural networks.

УДК 004.9:378

Лендюк Т.В.

ЗНАНИЕ-ОРИЕНТИРОВАННАЯ ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ ДЛЯ ПОСТРОЕНИЯ СИСТЕМЫ АДАПТИРОВАННОГО ОБУЧЕНИЯ

Введение. В настоящее время существуют и развиваются различные методы представления и описания знаний, например: продукционные модели, семантические сети, фреймы, таксономии, онтологии и так далее. В качестве наиболее перспективной модели пред-

Лендюк Тарас Васильевич, *преподаватель кафедры информационно-вычислительных систем и управления Тернопольского национального экономического университета,*
Украина, ТНЕУ, 46020, г. Тернополь, площадь Победы, 3.