$x_3 = 10$) соответствует диапазону представления информационных символов, которые соответствуют правильным значениям информационных символов. Если только в одном из двух выбранных символов, например (x_2, x_4), есть ошибка, то среди множества решений не будет правильного, то есть такого, что соответствует диапазону представления информационных символов (рис. 1, кривая *Error:x2, x4*).

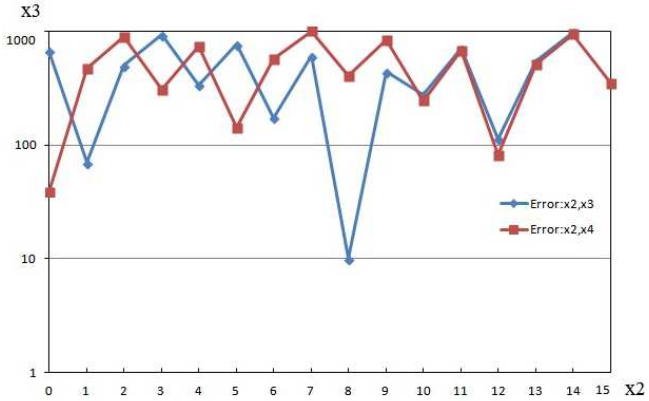

Рис. 1. Множество решений диофантового уравнения (9)

Скорость кода вычисляется по формуле $R = \frac{k}{n}$, где k – разрядность данных, n – общая длина кода. Для рассматриваемого примера: $k = 32$ бит (восемь четырех разрядных символов), $n = 32 + \lceil \log_2 1021 \rceil = 42$, $R = 0.76$. При использовании двух проверочных символов $n = 32 + 2 \cdot \lceil \log_2 1021 \rceil = 52$, $R = 0.62$. Таким образом, использование разработанного метода позволяет примерно на 20% повысить скорость кода.

Разработанный метод обеспечивает исправление ошибок в двух информационных символах с использованием одного проверочного символа и тем самым позволяет увеличить скорость кода примерно на 20%, соответственно уменьшить избыточность модулярного корректирующего кода.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Howard, S.L. Error control coding in low-power wireless sensor networks: When is ECC energy-efficient? / S.L.Howard, C. Schlegel, K. Iniewski // EURASIP Journal on Wireless Communications and Networking. – 2006. – № 2. – P. 29.
2. Яцків, В.В. Модифіковані коректуючі коди системи залишкових класів та їх застосування / В.В. Яцків // Інформаційні технології та комп'ютерна інженерія. – 2013. – № 2. – С. 39–45.
3. Yatskiv, V. Multiple Error Detection and Correction Based on Modular Arithmetic Correcting Codes / V. Yatskiv, T. Tsavolyk, Hu Zhengbing // Proceedings of the 8-th 2015 IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems, IDAACS'2015. – Warszawa. – 2015. – Volume 2. – P. 850–854.

Материал поступил в редакцию 09.12.15

TSAVOLYK T.G., YATSKIV V.V. Error-correcting method based on modular correcting codes

In this paper we proposed correcting codes - based on modular arithmetic - to improve the data transmission robustness in wireless sensor networks. We developed a new method and algorithm for the detection and correction of multiple errors. The codes are characterized by high correction characteristics as well as the simplified coding procedure. These codes provide the possibility to improve adaptively the reliability of data transmission in wireless sensor networks without changing the coding principle.

УДК 004.032.26

Савицкий Ю.В., Хвещук В.И., Савицкий А.Ю.

МОДИФИКАЦИЯ АЛГОРИТМА BPE ДЛЯ АДАПТИВНОГО ОБУЧЕНИЯ СИГМОИДАЛЬНЫХ НЕЙРОНОВ В АРХИТЕКТУРЕ МНОГОСЛОЙНОЙ НЕЙРОННОЙ СЕТИ

Введение. В последнее время в мире активизировались исследования в области глубокого обучения многослойных нейронных сетей. Это связано с определенными успехами в данной области, достигнутыми рядом исследователей [1, 2], а также высокой практической значимостью сильно-многослойных нейронных сетей (СМНС). Так, ряд разработчиков интеллектуального программного обеспечения (корпорации Google, Microsoft и др.) с успехом применяют технологии глубоких нейронных сетей в различных своих приложениях. При этом перспективным считается подход к предобучению (pretraining) СМНС не только с помощью ограниченной машины Больцмана (RBM), но и с применением нейросетевых автоэнкодеров (Autoencoder). Каждый такой нейросетевой автоэнкодер представляет собой трехслойный персептрон архитектуры *N→M→N*, где параметр *N* соответствует количеству входов текущего предобучаемого слоя, *M* – количеству нейронов указанного слоя сильно-многослойной нейросетевой архитектуры. Последовательное (начиная с входного слоя СМНС) обучение совокупности таких автоэнкодеров на входной обучающей выборке позволяет получить наборы весовых коэффициентов для финальной настройки синаптических связей всей СМНС (fine-tuning). При этом, для обучения как нейросетевых автоэнкодеров, так и СМНС, как правило, применяется алгоритм обратного распространения ошибки (Back Propagation Error, BPE) [3]. Очевидно, что эффективность алгоритма BPE напрямую определяет эффективность (точность обучения, обобщающие свойства) результирующей модели СНМС в целом.

В данной работе предлагается методика точного обучения нейронных элементов (НЭ) сигмоидального типа в составе многослойной нейронной сети (с целью последующего использования для обучения нейроросетевых автоэнкодеров и СНМС); основные результаты сформулированы в теореме 1 и утверждении 1; выполняется анализ особенностей предложенных решений.

1. Обобщенная архитектура нейронной сети. На рисунках 1 и

***Савицкий Юрий Викторович**, к.т.н., доцент кафедры «Интеллектуальные информационные технологии» Брестского государственного технического университета.*

***Хвещук Владимир Иванович,** к.т.н., профессор кафедры «Интеллектуальные информационные технологии» Брестского государственного технического университета.*

***Савицкий Антон Юрьевич,** студент 4 курса специальности «Автоматизированные системы обработки информации» факультета электронно-информационных систем Брестского государственного технического университета.*

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

2 приведены соответственно обобщенная архитектура многослойной нейронной сети, структура искусственного НЭ, а также введены обозначения параметров сети.

Нейронный элемент слоя l осуществляет функцию преобразования некоторого вектора входных сигналов $Y^{[l-1]}$ в выходную активность $Y^{[l]}$ по следующему правилу:

$$\begin{cases} S_j^{[l]} = \sum_{i=1}^{N^{[l-1]}} y_i^{[l-1]} w_{ij}^{[l]} - w_{bj}^{[l]}, \\ y_j^{[l]} = g^{[l]}(S_j^{[l]}), j = 1,...N^{[l]}, \end{cases} \quad (1)$$

где $S_j^{[l]}$ – взвешенная сумма входных активностей НЭ j, находящегося в слое l;

$w_{ij}^{[l]}$ – значение синаптического веса i-го входа НЭ;

$w_{bj}^{[l]}$ – значение порога активационной функции $g^{[l]}(S_j^{[l]})$ НЭ;

$N^{[l-1]}, N^{[l]}$ - соответственно количество входов НЭ слоя l и количество НЭ данного слоя.

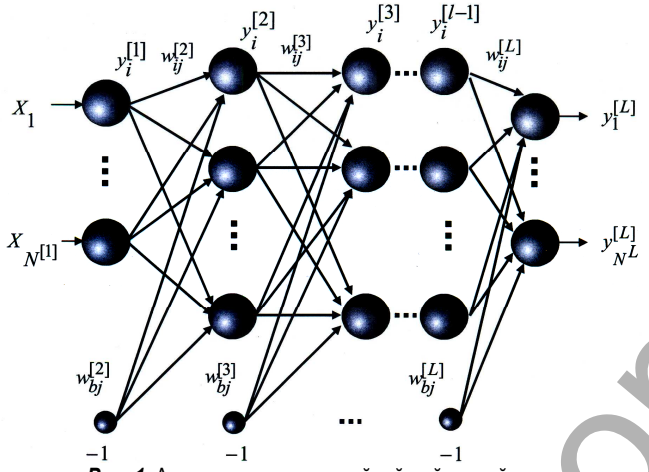

Рис. 1. Архитектура многослойной нейронной сети

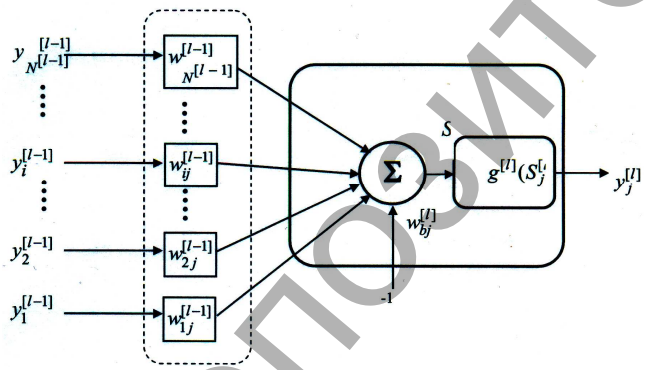

Рис. 2. Структура искусственного нейроэлемента j в слое l нейронной сети

2. Адаптивное обучение нейронов сигмоидального типа. Как было сказано выше, точность решения практических задач в большой степени определяется параметрами обучающих процедур. Детальный анализ литературных источников показал, что наиболее распространенным методом обучения базовых архитектур многослойных нейронных сетей является алгоритм BPE. Он относится к алгоритмам обучения с учителем и нуждается в организации обучающих выборок. Классический алгоритм BPE использует метод градиентного спуска для минимизации функции среднеквадратичной погрешности. Благодаря высокой точности и малой вычислительной сложности градиентных методов оптимизации данный алгоритм позволяет достигать малой погрешности обучения, что является крайне важным фактором для решения большинства практических задач в нейросетевом базисе.

Рассмотрим применение алгоритма BPE для обучения многослойной нейронной сети. Пусть для обучения сформировано обучающее множество, состоящее из пар векторов $T = \{(X^p, D^p)\}, p = 1,...,P$ размерностью, соответствующей количеству входов и выходов сети. Тогда задача процедуры обучения заключается в адаптации параметров сети (синаптических связей и порогов нейронов) таким образом, чтобы на любой входной вектор X^p обучающей выборки было сформировано корректное отображение Y^p, отличающееся от желаемого D^p с минимальной ошибкой.

Согласно методу градиентного спуска, итерационное изменение весовых коэффициентов и порогов НЭ для каждого слоя нейронной сети происходит по следующим правилам:

$$\begin{cases} w_{ij}^{[l],p}(t+1) = w_{ij}^{[l],p}(t) - \alpha \dfrac{\partial E^p(t)}{\partial w_{ij}^{[l],p}(t)} = \\ = w_{ij}^{[l],p}(t) - \alpha \cdot \gamma_j^{[l],p}(t)(g^{[l]})'(S_j^{[l],p}(t)) y_i^{[l-1],p}(t), \\ w_{bj}^{[l],p}(t+1) = w_{bj}^{[l],p}(t) - \alpha \dfrac{\partial E^p(t)}{\partial w_{bj}^{[l],p}(t)} = \\ = w_{bj}^{[l],p}(t) + \alpha \cdot \gamma_j^{[l],p}(t)(g^{[l]})'(S_j^{[l],p}(t)), \end{cases} \quad (2)$$

где α – константа, определяющая шаг обучения;

$(g^{[l]})'(S_j^{[l],p}(t))$ – производная активационной функции НЭ;

$\dfrac{\partial E^p(t)}{\partial w_{ij}^{[l],p}(t)}, \dfrac{\partial E^p(t)}{\partial w_{bj}^{[l],p}(t)}$ – частные производные функции ошибки нейронных связей, вычисляемые на каждой итерации обучения для каждого эталона p, $p = \{1,...,P\}$;

$\gamma_j^{[l],p}(t)$ – ошибка НЭ j, определяемая как:

$$\gamma_j^{[L],p}(t) = y_j^{[L],p}(t) - D_j^p \quad (3)$$

для нейронов выходного слоя, либо:

$$\gamma_j^{[L],p}(t) = \sum_{j=1}^{N^{[l]}} y_j^{[L],p}(t)(g^{[l]})'(S_j^{[l],p}(t)) w_{ij}^{[l],p}(t) \quad (4)$$

для нейронов скрытых слоев сети;

$E^p(t)$ - среднеквадратичная ошибка нейронной сети для эталона p, определяемая как:

$$E^p(t) = \sum_{j=1}^{N^{[L]}} E_j^p(t), \quad (5)$$

$$E_j^p(t) = \frac{1}{2}(y_j^{[L],p}(t) - D_j^p)^2, \quad (6)$$

где D_j^p — эталонное выходное значение j-го нейрона слоя L.

В результате выполнения каждой новой итерации обучения t происходит минимизация общей ошибки сети, определяемой как:

$$E(t) = \sum_{p=1}^{P} E^p(t). \quad (7)$$

Таким образом, задача вычисления градиентов функции ошибки для нейроэлементов сети и модификации значений синаптических связей сводится к послойному определению ошибок нейронов в направлении от выходного слоя сети по правилам (3), (4) и использовании выражений (2).

Теорема 1. *Правила модификации синаптических связей НЭ j, находящегося в слое L, с функцией активации $g^{[L]}$, минимизирующие среднеквадратичную ошибку $E_j^p(t) = 1/2(y_j^{[L],p}(t) - D_j^p)^2$ данного НЭ для эталона p на*

итерации обучения t, определяются следующим образом:

1.1 для сигмоидной функции $g^{[L]}$:

$$\begin{cases} w_{ij}^{[L],p}(t+1)=w_{ij}^{[L],p}(t)-\dfrac{S_j^{[L],p}(t)-\ln\left(\dfrac{D_j^p}{1-D_j^p}\right)}{1+\sum\limits_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}\right)^2}y_i^{[L-1],p}(t), \\ W_{bj}^{[L],p}(t+1)=w_{bj}^{[L],p}(t)+\dfrac{S_j^{[L],p}(t)-\ln\left(\dfrac{D_j^p}{1-D_j^p}\right)}{1+\sum\limits_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}\right)^2}, \end{cases} \quad (8)$$

1.2 для биполярной сигмоидной функции $g^{[L]}$:

$$\begin{cases} w_{ij}^{[L],p}(t+1)=w_{ij}^{[L],p}(t)-\dfrac{S_j^{[L],p}(t)-\ln\left(\dfrac{1+D_j^p}{1-D_j^p}\right)}{1+\sum\limits_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}\right)^2}y_i^{[L-1],p}(t), \\ W_{bj}^{[L],p}(t+1)=w_{bj}^{[L],p}(t)+\dfrac{S_j^{[L],p}(t)-\ln\left(\dfrac{1+D_j^p}{1-D_j^p}\right)}{1+\sum\limits_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}\right)^2}, \end{cases} \quad (9)$$

1.3 для функции $g^{[L]}$ гиперболический тангенс:

$$\begin{cases} w_{ij}^{[L],p}(t+1)=w_{ij}^{[L],p}(t)-\dfrac{S_j^{[L],p}(t)-\frac{1}{2}\ln\left(\dfrac{1+D_j^p}{1-D_j^p}\right)}{1+\sum\limits_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}\right)^2}y_i^{[L-1],p}(t), \\ W_{bj}^{[L],p}(t+1)=w_{bj}^{[L],p}(t)+\dfrac{S_j^{[L],p}(t)-\frac{1}{2}\ln\left(\dfrac{1+D_j^p}{1-D_j^p}\right)}{1+\sum\limits_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}\right)^2}. \end{cases} \quad (10)$$

Докажем п. 1.3 теоремы. Пусть $\alpha_j^{[L],p}(t+1)$ - шаг обучения, используемый в правилах (2) алгоритма BPE для получения модифицированных весовых коэффициентов $w_{ij}^{[L],p}(t+1)$, $w_{bj}^{[L],p}(t+1)$ рассматриваемого НЭ для эталона p. Определим $\alpha_j^{[L],p}(t+1)$ на базе метода наискорейшего спуска с целью минимизации функции ошибки, рассчитываемой для модифицированных весовых коэффициентов по формуле:

$$E_j^p(t+1)=\frac{1}{2}(y_j^{[L],p}(t+1)-D_j^p)^2, \quad (11)$$

что предполагает решение уравнения:

$$\frac{\partial E_j^p(t+1)}{\partial\alpha_j^{[L],p}(t+1)}=0. \quad (12)$$

Первоначально выразим взвешенную сумму входов рассматриваемого НЭ с учетом модифицированных весовых коэффициентов и производной сигмоидной функции активации (13).

В выражении (13) $y_i^{[L-1],p}(t)$ представляют собой текущие значения входной активности рассматриваемого НЭ.

Введем следующее обозначение:

$$B_j^{[L],p}(t)=\left(y_j^{[L],p}(t)-D_j^p\right)y_j^{[L],p}(t)\times \\ \times\left(1-y_j^{[L],p}(t)\right)\left(1+\sum_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}(t)\right)^2\right) \quad (14)$$

Тогда выражение (12) примет следующий вид:

$$S_j^{[L],p}(t+1)=S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)\cdot B_j^{[L],p}(t). \quad (15)$$

Выходная активность рассматриваемого НЭ с сигмоидной функции активации определяется выражением:

$$y_j^{[L],p}(t+1)=g^{[L]}\left(S_j^{[L],p}(t+1)\right)=\frac{1}{1+e^{-S_j^{[L],p}(t+1)}},\quad 0<g^{[L]}<1. \quad (16)$$

На основе выражений (11), (15) и (16) составим и решим уравнение вида (12):

$$\frac{\partial E_j^p(t+1)}{\alpha_j^{[L],p}(t+1)}=\left(\frac{1}{1+e^{-\left(S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)(B_j^{[L],p}(t)\right)}}-D_j^p\right)\times \\ \times\left(\frac{1}{1+e^{-\left(S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)(B_j^{[L],p}(t)\right)}}\right)\times \\ \times\left(1-\frac{1}{1+e^{-\left(S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)(B_j^{[L],p}(t)\right)}}\right)\cdot B_j^{[L],p}(t)=0. \quad (17)$$

$$S_j^p(t+1)=\sum_{i=1}^{N^{[L-1]}}y_i^{[L-1],p}w_{ij}^{[L]}(t+1)-w_{bj}^{[L]}(t+1)= \\ =\sum_{i=1}^{N^{[L-1]}}\left[\begin{array}{l} y_i^{[L-1],p}\left(w_{ij}^{[L]}(t)-\alpha_j^{[L],p}(t+1)\left(y_j^{[L],p}(t)-D_j^p\right)y_j^{[L],p}(t)\left(1-y_j^{[L],p}(t)\right)y_i^{[L-1],p}\right)- \\ -\left(w_{bj}^{[L]}(t)+\alpha_j^{[L],p}(t+1)\left(y_j^{[L],p}(t)-D_j^p\right)y_j^{[L],p}(t)\left(1-y_j^{[L],p}(t)\right)\right) \end{array}\right]= \\ =\sum_{i=1}^{N^{[L-1]}}\left[\begin{array}{l} y_i^{[L-1],p}w_{ij}^{[L]}(t)-w_{bj}^{[L]}(t)-\alpha_j^{[L],p}(t+1)\left(y_j^{[L],p}(t)-D_j^p\right)y_j^{[L],p}(t)\left(1-y_j^{[L],p}(t)\right)y_i^{[L-1],p}(t)^2- \\ -\alpha_j^{[L],p}(t+1)\left(y_j^{[L],p}(t)-D_j^p\right)y_j^{[L],p}(t)\left(1-y_j^{[L],p}(t)\right) \end{array}\right]= \\ =S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)\left(y_j^{[L],p}(t)-D_j^p\right)y_j^{[L],p}(t)\left(1-y_j^{[L],p}(t)\right)\left(1+\sum_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}(t)\right)^2\right). \quad (13)$$

В результате решение уравнения (17) сводится к решению совокупности следующих уравнений:

$$\frac{1}{1+e^{-\left(S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)\left(B_j^{[L],p}(t)\right)\right)}}-D_j^p=0$$

$$\text{или}\ \frac{1}{1+e^{-\left(S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)\left(B_j^{[L],p}(t)\right)\right)}}=0 \quad (18)$$

$$\text{или}\ 1-\frac{1}{1+e^{-\left(S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)\left(B_j^{[L],p}(t)\right)\right)}}=0.$$

Поскольку для сигмоидной активационной функции имеют место ограничения $0<g^{[L]}<1$, то два последних уравнения из (19) решения не имеют.

В результате получаем следующее решение:

$$\frac{1}{1+e^{-\left(S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)\left(B_j^{[L],p}(t)\right)\right)}}=D_j^p,$$

$$e^{S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)\left(B_j^{[L],p}(t)\right)}=\frac{D_j^p}{1-D_j^p}, \quad (19)$$

$$S_j^{[L],p}(t)-\alpha_j^{[L],p}(t+1)\left(B_j^{[L],p}(t)\right)=\ln\left(\frac{D_j^p}{1-D_j^p}\right),$$

$$\alpha_j^{[L],p}(t+1)=\frac{S_j^{[L],p}(t)-\ln\left(\frac{D_j^p}{1-D_j^p}\right)}{B_j^{[L],p}(t)}. \quad (20)$$

С учетом (14) получаем (21).

$$\alpha_j^{[L],p}(t+1)=\frac{S_j^{[L],p}(t)-\ln\left(\frac{D_j^p}{1-D_j^p}\right)}{\left(y_j^{[L],p}(t)-D_j^p\right)y_j^{[L],p}(t)\left(1-y_j^{[L],p}(t)\right)\left(1+\sum_{k=1}^{N^{[L-1]}}\left(y_k^{[L-1],p}(t)\right)^2\right)}. \quad (21)$$

Подставив (21) в правила модификации весовых коэффициентов НЭ (2), получим искомые выражения (8).

Аналогичным образом доказываются п. 1.2 и п. 1.3 теоремы для НЭ с функциями активации биполярная сигмоидная и гиперболический тангенс соответственно.

Результаты теоремы 1 (с учетом алгоритма BPE) можно обобщить на НЭ последующих слоев многослойной нейронной сети.

Утверждение 1. *Правила модификации синаптических связей НЭ j, находящегося в скрытом слое l, с функцией активации $g^{[l]}$, минимизирующие ошибку данного НЭ для эталона p на итерации обучения t, можно определить следующим образом:*

1.1 для сигмоидной функции $g^{[l]}$:

$$\begin{cases} w_{ij}^{[l],p}(t+1)=w_{ij}^{[l],p}(t)-\dfrac{S_j^{[l],p}(t)-\ln\left(\dfrac{y_j^{[l],p}(t)-\gamma_j^{[l],p}(t)}{1-y_j^{[l],p}(t)+\gamma_j^{[l],p}(t)}\right)}{1+\sum_{k=1}^{N^{[l-1]}}\left(y_k^{[l-1],p}\right)^2}y_i^{[l-1],p}(t), \\ W_{bj}^{[l],p}(t+1)=w_{bj}^{[l],p}(t)+\dfrac{S_j^{[l],p}(t)-\ln\left(\dfrac{y_j^{[l],p}(t)-\gamma_j^{[l],p}(t)}{1-y_j^{[l],p}(t)+\gamma_j^{[l],p}(t)}\right)}{1+\sum_{k=1}^{N^{[l-1]}}\left(y_k^{[l-1],p}\right)^2}, \end{cases} \quad (22)$$

1.2 для биполярной сигмоидной функции $g^{[l]}$:

$$\begin{cases} w_{ij}^{[l],p}(t+1)=w_{ij}^{[l],p}(t)-\dfrac{S_j^{[l],p}(t)-\ln\left(\dfrac{1+y_j^{[l],p}(t)-\gamma_j^{[l],p}(t)}{1-y_j^{[l],p}(t)+\gamma_j^{[l],p}(t)}\right)}{1+\sum_{k=1}^{N^{[l-1]}}\left(y_k^{[L-1],p}\right)^2}y_i^{[l-1],p}(t), \\ W_{bj}^{[l],p}(t+1)=w_{bj}^{[l],p}(t)+\dfrac{S_j^{[L],p}(t)-\ln\left(\dfrac{1+y_j^{[l],p}(t)-\gamma_j^{[l],p}(t)}{1-y_j^{[l],p}(t)+\gamma_j^{[l],p}(t)}\right)}{1+\sum_{k=1}^{N^{[l-1]}}\left(y_k^{[l-1],p}\right)^2}, \end{cases} \quad (23)$$

1.3 для функции $g^{[l]}$ гиперболический тангенс:

$$\begin{cases} w_{ij}^{[l],p}(t+1)=w_{ij}^{[l],p}(t)-\dfrac{S_j^{[l],p}(t)-\frac{1}{2}\ln\left(\dfrac{1+y_j^{[l],p}(t)-\gamma_j^{[l],p}(t)}{1-y_j^{[l],p}(t)+\gamma_j^{[l],p}(t)}\right)}{1+\sum_{k=1}^{N^{[l-1]}}\left(y_k^{[L-1],p}\right)^2}y_i^{[l-1],p}(t), \\ W_{bj}^{[l],p}(t+1)=w_{bj}^{[l],p}(t)+\dfrac{S_j^{[L],p}(t)-\frac{1}{2}\ln\left(\dfrac{1+y_j^{[l],p}(t)-\gamma_j^{[l],p}(t)}{1-y_j^{[l],p}(t)+\gamma_j^{[l],p}(t)}\right)}{1+\sum_{k=1}^{N^{[l-1]}}\left(y_k^{[l-1],p}\right)^2}, \end{cases} \quad (24)$$

где $\gamma_j^{[l],p}(t)$ - ошибка j-го НЭ.

Утверждение 1 вытекает из теоремы 1, а также из допущения, что ошибку НЭ скрытого слоя $\gamma_j^{[l],p}(t)$ можно интерпретировать как разность между текущим значением выходной активности $y_j^{[l],p}(t)$ и желаемым значением выходной активности данного НЭ.

Наибольшую эффективность полученных результатов (теоремы 1 и утверждения 1) продемонстрировал *алгоритм последовательного послойного (по ходу распространения активностей) обучения*. В частности, для индивидуальных эталонов различного вида (в том числе из баз MNIST и NIST), нормализованных в диапазоне значений соответствующей функции активации, данный алгоритм продемонстрировал сходимость процесса обучения многослойной НС до нулевой ошибки E^p за одну итерацию обучения.

Следует также отметить, что более перспективным авторы считают применение полученных результатов в алгоритме группового (batch) обучения; именно в этом направлении в настоящий момент проводятся экспериментальные исследования.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Choi, C.H. Construction of Neural Networks to Approximate Arbitrary Continuous Functions Of One Variable / C.H. Choi, J.Y. Choi // Electron. Lett. – 1992. – №2. – P. 151–153.
2. Fahlman, Scott E. The Cascade–Correlation Learning Architecture / Scott E. Fahlman, C. Lebiere // Neural Information Processing System. – 1990. – № 2. – P. 524–532.
3. Fahlman, Scott E. The Cascade–Correlation Learning Architecture / Scott E. Fahlman, C. Lebiere // Neural Information Processing System. – 1990. – № 2. – P. 524–532.
4. Wynne–Jones, M. Node Splitting: A Constructive Algorithm For Feed–Forward Neural Networks // Neural Computing And Applications. – 1993. – Vol. 1. – № 1. – P. 17–22.

Материал поступил в редакцию 28.12.15

SAVITSKY Yu.V., HVESHCHUK V.I., SAVITSKY A.Yu. Modification of algorithm of BPE for adaptive training of sigmoidalny neurons in architecture of the multilayered neural network

In article it is formulated and modification of algorithm of the return distribution of a mistake (BPE) for exact training of neural elements with sigmoidalny functions of activation in architecture of a multilayered neural network is mathematically proved. The offered rules of modification of synoptic communications of neurons of day off and the hidden layers can be used for creation of the effective algorithms providing reduction of temporary and computing complexity of process of training of multilayered neural network architecture.

УДК 581.3

Николайчук Я.Н., Ивасьев С.В., Якименко И.З., Касянчук М.Н.

МЕТОД ФАКТОРИЗАЦИИ МНОГОРАЗРЯДНЫХ ЧИСЕЛ НА ОСНОВЕ СВОЙСТВ КВАДРАТИЧНОСТИ ВЫЧЕТОВ В СИСТЕМЕ ОСТАТОЧНЫХ КЛАССОВ

Введение. Факторизация является одной из важнейших задач теории чисел [1] и современной асимметричной криптографии [2]. Ее суть заключается в разложении некоторого целого числа в произведение простых сомножителей. Известные методы факторизации, в зависимости от их производительности, разбиваются на две группы: экспоненциальные и субэкспоненциальные [3]. Все они достаточно громоздки, поэтому требуют значительных вычислительных ресурсов для обработки многоразрядных чисел. Однако теоретическое обоснование необходимой сложности таких вычислений или, другими словами, существование нижних оценок не доказано, поэтому вопрос о существовании алгоритма факторизации с полиномиальной сложностью является одной из открытых проблем современной теории чисел. Кроме того, актуальность проблемы факторизации продиктована также неопределенностью относительно теоретического обоснования устойчивости к раскрытию асимметричных криптосистем [4].

Наиболее распространенными для факторизации являются алгоритмы, основанные на теореме Ферма [3].

Анализ метода Ферма для факторизации многоразрядных чисел. Пусть P_0 – известное целое число, являющееся произведением двух простых чисел, которые нужно найти. Большинство современных методов факторизации основываются на идее, предложенной Пьером Ферма, которая заключается в поиске пар натуральных чисел A и B таких, что выполняется соотношение: $P_0 = A^2 - B^2$.

Метод Ферма описывается таким выражением:

$$\Delta_n = \sqrt{n^2 - P_0}\ , \qquad (1)$$

где $n = \left\lceil \sqrt{P_0} \right\rceil + k$, k=1, 2, 3, … .

Количество итераций k равняется значению, когда параметр Δ_n будет целым числом. Отсюда можно найти искомое разложение на множители:

$$P_0 = (n - \Delta_n)(n + \Delta_n)\ . \qquad (2)$$

Вычислительная сложность метода Ферма для многоразрядных чисел достаточно большая, поскольку количество итерацій k может быть порядка 2^{300}–2^{400} и только на единственном шагу возможно однозначное решение задачи факторизации. Причем операции необходимо виполнять над числами, разрядность которых примерно 300–500 бит.

Для упрощения этой задачи целесообразно использовать систему остаточных классов (СОК) [5], которая позволит выполнить распараллеливание процесса вычислений и будет выступать индикатором квадратичности вычетов.

Исходя из вышесказанного, целью данной работы является усовершенствование алгоритма факторизации на основании теоремы Ферма с использованием СОК для уменьшения вычислительной сложности при работе с многоразрядными числами.

Метод факторизации многоразрядных чисел на основании теоремы Ферма с помощью использования свойств квадратичности остатков. Известно, что квадраты целых чисел можно представить в виде суммы нечетных чисел, количество которых равняется данному числу [6]:

$$n^2 = \sum_{i=1}^{n} (2i - 1)\ . \qquad (3)$$

Поэтому, отыскав Δ_n по формуле (1) при k=1, следующие итерации выполняются согласно выражению $S_k = \sqrt{(\Delta_0)^2 + (2n - 1)}$. Итерации продолжаются до тех пор, пока параметр S_k не будет целым числом, причем количество итераций в обоих методах одинаково.

В таблице 1 представлен пример факторизации с помощью классического и усовершенствованного методов Ферма для P_0=3811.

Таким образом, получено разложение числа 3811 на простые сомножители:

$$3811 = 70^2 - 33^2 = (70 + 33)(70 - 33) = 103*37.$$

Из таблицы 1 видно, что в совершенствованном методе исключается операция возведения в квадрат. Кроме этого, арифметические действия выполняются над числами, размерность которых на несколько порядков меньше, чем в классическом методе. Однако следует отметить, что количество итераций в обоих случаях будет одинаково, а самой сложной остается операция проверки квадратичности остатка. Для уменьшения ее вычислительной сложности можно использовать СОК.

Исследование свойств квадратов в СОК. Рассмотрим остатки квадратов целых чисел по нескольким простым модулям p_j, то есть

$a_1(p_1, p_2, ..., p_m) = b_1^1, b_2^1, \ldots, b_m^1$,

$a_2(p_1, p_2, ..., p_m) = b_1^2, b_2^2, \ldots, b_m^2$,

$a_n(p_1, p_2, ..., p_m) = b_1^n, b_2^n, \ldots, b_m^n$, где $a_i = i^2$,

$b_j^i = a_i (\mathrm{mod}\, p_j), 1 \le i \le n,\ 1 \le j \le m$, m – количество модулей.

Используя свойство (3), искомые остатки можно получить с помощью рекурентной формулы $b_j^i = \left(b_j^{i-1} + z_i^j\right) \mathrm{mod}\, p_j$, где $z_j^i = z_i \bmod p_j$, z_i=2i-1.

Николайчук Ярослав Николаевич, *д.т.н., профессор, заведующий кафедрой специализированных компьютерных систем Тернопольского национального экономического университета.*

Ивасьев Степан Владимирович, *старший преподаватель кафедры компьютерной инженерии Тернопольского национального экономического университета.*

Якименко Игорь Зиновьевич, *к.т.н., доцент кафедры компьютерной инженерии Тернопольского национального экономического университета.*

Касянчук Михаил Николаевич*, к.ф.-м.н., доцент кафедры компьютерной инженерии Тернопольского национального экономического университета.*

Украина, 46009, г. Тернополь, Тернопольская область, ул. Львовская, 11.