

Для классификации растений необходимо найти лист на изображении. Для этого лучше использовать алгоритм разрастания областей [2, с. 192]. В качестве входных данных этот метод принимает изображения и набор семян. Семена отмечают объекты, которые нужно выделить. Области постепенно разрастаются, сравнивая все незанятые соседние пиксели с областью. Разность между яркостью пикселя и средней яркостью области используется как мера схожести. Пиксель с наименьшей такой напрасностью добавляется в соответствующую область. Процесс продолжается, пока все пиксели не будут добавлены в один из регионов. Значение яркостей пикселей, вокруг которых производится наращивание областей, могут быть заранее заданы, это так называемые «центры кристаллизации», или, например, в процессе сегментации значение яркости, с которым сравнивается каждый новый проверяемый пиксел, может каждый раз вычисляться как среднее значение яркости, уже выращенной к данному моменту области.

Когда лист найден, вычисляется его площадь и периметр. После чего находится их отношение по формуле.

$$D=S/P,$$

где  $S$  – площадь листа растения,  $P$  – периметр листа растения.

После каждого уровня классификатора осуществляется сравнение признаков эталона и обрабатываемого изображения. Если признаки схожи, то выполняется переход на следующий уровень, где осуществляется дальнейшее сравнение.

Эталоны, в качестве изображений, и их признаки хранятся в специально созданной базе данных. Эталонами являются заранее сфотографированные листья с точным определением сорта.

В результате апробации разработанного программного обеспечения было выявлено стопроцентная идентификация на эталонных изображениях и 70% на реальных изображениях объектов.

#### **Список цитированных источников**

1. Красильников, Н.Н. Цифровая обработка 2D- и 3D-изображений: учеб. пособие. – СПб: БХВ-Петербург, 2011. – 608 с.
2. Pratt, W.K. Digital Image Processing 4th Edition / John Wiley & Sons, Inc. – Los Altos, California, 2007.

УДК 004.8

## **АДАПТИВНОЕ ОБУЧЕНИЕ СИГМОИДАЛЬНЫХ НЕЙРОЭЛЕМЕНТОВ В АРХИТЕКТУРЕ МНОГОСЛОЙНОЙ НЕЙРОННОЙ СЕТИ**

**Савицкий А.Ю., Савицкий Ю.В.**

*Брестский государственный технический университет, г. Брест*

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). При этом перспективным считается подход к предобучению (pre-training) СМНС не только с помощью ограниченной машины Больцмана (RBM), но и с применением нейросетевых автоэнкодеров (AutoEncoder, AE). Каждый такой AE представляет собой трехслойный персептрон архитектуры  $N \rightarrow M \rightarrow N$ , где параметр  $N$  соответствует количеству входов текущего предобучаемого слоя,  $M$  – количеству нейронов указанного слоя СМНС. Последовательное (начиная с входного слоя СМНС) обучение совокупности таких AE позволяет получить наборы весовых коэффициентов для финальной настройки синаптических связей всей СМНС (fine-tuning). При этом, для обучения как AE, 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 СМНС в целом.

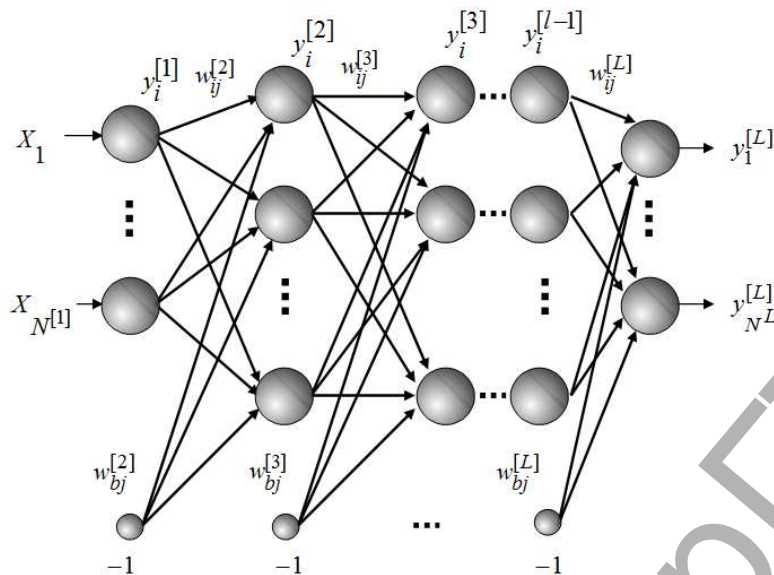


Рисунок 1 – Архитектура многослойной нейронной сети

На рисунке 1 приведена 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:

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

где  $S_j^{[l]}$  – взвешенная сумма входных активностей НЭ  $j$ ;  $w_{ij}^{[l]}$ ,  $w_{bj}^{[l]}$  – весовые коэффициенты;  $g^{[l]}(S_j^{[l]})$  – активационная функция НЭ.

В рамках данной архитектуры сформулирована и доказана теорема, позволяющая осуществлять точную настройку весовых коэффициентов нейроэлементов с сигмоидной функцией активации.

**Теорема.** Правила модификации синаптических связей НЭ  $j$ , находящегося в слое  $L$ , с сигмоидной 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:

$$\left\{ \begin{aligned} w_{ij}^{[L]}(t+1) &= w_{ij}^{[L]}(t) - \frac{S_j^{[L],p}(t) - \ln\left(\frac{D_j^p}{1-D_j^p}\right)}{1 + \sum_{k=1}^{N^{[L-1]}} (y_k^{[L-1],p})^2} y_i^{[L-1],p}(t), \\ w_{bj}^{[L]}(t+1) &= w_{bj}^{[L]}(t) + \frac{S_j^{[L],p}(t) - \ln\left(\frac{D_j^p}{1-D_j^p}\right)}{1 + \sum_{k=1}^{N^{[L-1]}} (y_k^{[L-1],p})^2}, \end{aligned} \right. \quad (2)$$

где  $D_j^p$  — эталонное выходное значение  $j$ -го НЭ.

Авторами также получены аналогичные правила модификации синаптических связей для НЭ с функциями активации биполярная сигмоидная и гиперболический тангенс.

Следует также отметить, что результаты теоремы (с учетом алгоритма ВРЕ) можно обобщить на НЭ последующих слоев многослойной нейронной сети.

Ряд вычислительных экспериментов был проведен при обучении нейросетевых энкодеров архитектуры 900→50→900. С этой целью для обучения были сгенерированы битовые образы печатных символов размером 30×30 и количеством 100 экземпляров. При выполнении 100 итераций обучения среднеквадратичная ошибка обучения (в расчете на один эталон) достигла значения 0.000945, чего не удавалось достичь методом эмпирического подбора шага. Таким образом, полученные правила (2) можно применять как на этапе предобучения в нейросетевых энкодерах, так на этапе финальной настройки целостной архитектуры СМНС.

#### **Список цитированных источников**

1. Hinton G.E. A fast learning algorithm for deep belief networks / G.E. Hinton, S. Osindero, Y.-W. Teh // *Neural Computation*. – 2006. – Vol. 18. – №. 7. – P. 1527–1554.
2. Bengio, Y. Learning deep architectures for AI // *Foundations and Trends in Machine Learning*. – Vol. 2. – №. 1. – 2009. – P. 1–127.
3. Rumelhart D.E. Learning representations by back-propagating errors / D.E. Rumelhart, G.E. Hinton, R.J. Williams // *Nature*, Vol. 323. – 1986. – P. 533–536.

УДК 004.9

## **ОСОБЕННОСТИ ПРИМЕНЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОРГАНИЗАЦИЯХ РАЗЛИЧНОГО ТИПА**

*Швецова Е.В.*

*Брестский государственный технический университет, г. Брест*

Выбор информационной технологии в той или иной предметной области определяется следующими моментами: сферой деятельности организации; принятой моделью управления; новыми задачами, стоящими перед организацией; существующей информационной инфраструктурой; величиной организации. С учетом этих критериев делается выбор программно-аппаратного обеспечения информационной технологии.

Так на малых предприятиях информационные технологии, как правило, связаны с решением задач бухгалтерского учета, накоплением информации, созданием информационных баз данных и организации телекоммуникационной среды для связи пользователей между собой и с другими предприятиями и организациями. Персонал малых предприятий работает в среде локальных вычислительных сетей различной топологии.

Функционал для специалистов среднего уровня реализуется в виде различного рода АРМ, а банк данных используется для информационной поддержки работы верхнего управленческого звена. Таким образом, на малых предприятиях более рационально комбинировать информационную архитектуру, сочетающую в себе распределенную обработку данных с централизованным хранением информационных ресурсов в едином банке данных [1]. В качестве центральной вычислительной системы можно использовать, например, UNIX- сервер, под управлением которого функционирует сетевая СУБД, например MS SQL Server, Oracle или PostgreSQL, накопление информации в базу данных которой осуществляется посредством все тех же АРМ или иного клиентского ПО.

Комбинированная сетевая организация автоматизированной информационной технологии дает такие преимущества, как экономия ресурсов, гибкость к требованиям пользователей за счет широкого сочетания аппаратных и программных средств и т.д.

Тем не менее опора на единственный сервер не всегда является лучшим выходом, т.к. существуют ограничения на количество клиентских подключений, а так же снижение отказоустойчивости. Увеличение числа клиентов приводит к замедлению реакции сис-