

УДК 004.5/.93

МОДЕЛИРОВАНИЕ ПРОЦЕССА РАСПОЗНАВАНИЯ ОБРАЗОВ ДЛЯ SMART-ОБРАЗОВАТЕЛЬНЫХ СИСТЕМ

Обади А.А., Аль-Хашеди А.А., Нуриев Н.К., Печеный Е.А.

ФГБОУ ВО «Казанский национальный исследовательский технологический университет», Казань,
e-mail: 19fattah86@mail.ru, alhashedi@mail.ru

В работе рассматривается проблема распознавания образов применительно к обучающей SMART-системе, используемой для классификации задач учебной дисциплины «Численные методы», входящей в состав учебных планов бакалавриата технических специальностей высшего профессионального образования. С помощью аппарата исчисления предикатов строится вектор, характеризующий предлагаемую задачу, который, после умножения на некоторые весовые коэффициенты, поступает на вход специальной программы классификатора, построенной на базе алгоритма Хебба. Это итерационная процедура, производящая на каждом шаге коррекцию весовых коэффициентов и проверку сформулированного решающего правила. Критерием останова является выполнение решающего правила, что позволяет однозначно поставить в соответствие предъявленной задаче один из известных классов. В тексте статьи приведена блок-схема программы классификатора и соответствующий скриншот. Подробно рассмотрен и прокомментирован демонстрационный пример. Для более эффективной работы классификатора предлагается предварительно ранжировать описанные классы с помощью формулы Байесовских вероятностей.

Ключевые слова: SMART-система, алгоритм Хебба, математическая модель, распознавания образов, перцептрон, нейронная сеть, программное обеспечение, классификация, Байесовская вероятность

MODELING OF PATTERN RECOGNITION PROCESS FOR SMART EDUCATIONAL SYSTEMS

Obadi A.A., Alhashedi A.A., Nuriev N.K., Pecheny E.A.

Federal State Budget Educational Institution of Higher Education «Kazan National Research Technological University», Kazan, e-mail: 19fattah86@mail.ru, alhashedi@mail.ru

The paper deals with the problem of pattern recognition as applied to the training SMART system, used to classify the tasks of the academic discipline «Numerical Methods», which is part of the curriculum of the bachelor's degree in technical specialties of higher professional education. With the help of the predicate calculus apparatus constructed vector characterizing the proposed task, which, after multiplying by some weight coefficients, enters to the input of a special classifier program based on the Hebb algorithm. This an iterative procedure performs at each step of the correction of the weight coefficients and the verification of the formulated decision rule. The stopping criterion is the execution of a decisive rule, which makes it possible to unequivocally align the presented problem to one of the known classes. The text of the article contains a block diagram of the classifier program and a corresponding screenshot. A demo example is detailed reviewed and commented on. For more effective work of the classifier it is proposed to pre-rank the described classes using the Bayesian probabilities formula.

Keywords: SMART system, Hebb algorithm, mathematical model, pattern recognition, perceptron, neural network, software, classification, Bayesian probabilities

В работе [1] рассмотрена проблема компьютерного распознавания задач, поступающих на вход классификатора из предметной области учебной дисциплины «Численные методы». Описание каждого объекта ω (задачи) задается совокупностью признаков, т.е. характерных особенностей присущих каждой задаче [1, 2]. Для каждого объекта ω строится информационный вектор признаков $I(\omega) = \{I_1(\omega), I_2(\omega), \dots, I_m(\omega)\}$, где m – количество признаков. С помощью исчисления предикатов вектор $I(\omega)$ преобразуется в вектор с булевыми переменными 0 или 1 [3, 4]. В результате чего получается совокупность векторов $\Xi = [x_i, \dots, x_N]$, где x_i – вектор признаков, принадлежащих i -тому классу, $i = 1, N$. Через N обозначено количество классов предметной области. Порядок следования элементов определяет-

ся путем ранжирования по убыванию значения байесовской вероятности, что позволяет сократить перебор классов при обучении системы. В работе рассматривается задача обучения системы процедуре безошибочной классификации объектов по классам с помощью перцептрона.

Цель работы

Построение математической модели и программного обеспечения для эффективного распознавания образов с помощью алгоритма обучения Хебба.

Распознавание образов с помощью алгоритма обучения

В самом общем понимании задача распознавания образов состоит в том, чтобы как можно более точно, а в идеале одно-

значно, установить факт принадлежности исследуемого объекта одному из известных классов или доказать отсутствие возможности сделать это [5, 6]. Сложностью и одновременно особенностью задачи распознавания является то обстоятельство, что признаки, характеризующие классифицируемый объект, могут быть элементами описания многих классов. Это означает, что решающие правила классификатора должны ставить в соответствие каждому классу вполне определенный набор признаков, или признаки имманентные этому и только этому классу. Естественно, что при изменении состава класса, например, в случае выделения в нем нескольких подклассов, число классификационных признаков будет возрастать.

Высказанные соображения позволяют утверждать, что, даже если вероятность принадлежности признаков исследуемого объекта к некоторому классу значительно больше, нежели к другим, это не может служить основанием для однозначного отнесения его к данному классу. Тем не менее вероятностный подход остается весьма продуктивным, поскольку на базе байесовских оценок дает возможность сформировать начальный весовой вектор $w = (w_1, w_2, \dots, w_N)$ для вычисления решающей функции, в качестве которой выступает скалярное произведение $d_i(x_i) = (w, x_i)$.

Пример реализации этой идеи для классификации задач учебной дисциплины «Численные методы» был рассмотрен в работе [1].

Весовой вектор $w = (w_1, w_2, w_3, w_4, w_5, w_6) = (0,25, 0,215, 0,17, 0,143, 0,1, 0,0712, 0,0358)$, а решающие функции получим следующим образом:

$$d_1(x_1) = \sum_{i=0}^7 wx_1 = 0,985,$$

аналогично

$$d_2(x_2) = \sum_{i=0}^7 wx_2 = 0,735,$$

$$d_3(x_3) = \sum_{i=0}^7 wx_3 = 0,458,$$

$$d_4(x_4) = \sum_{i=0}^7 wx_4 = 0,5562,$$

$$d_5(x_5) = \sum_{i=0}^7 wx_5 = 0,1712,$$

$$d_6(x_6) = \sum_{i=0}^7 wx_6 = 0,17,$$

$$d_7(x_7) = \sum_{i=0}^7 wx_7 = 0,52.$$

Полученный результат не означает, что система окончательно классифицировала объект. Для однозначной классификации, которую мы в дальнейшем будем именовать правильной, необходимо получить решающую функцию, такую, что $d_i(x_i) > 0$, если $x \in \Omega_i$ и $d_j(x) \leq 0$ для всех $j \neq i$. Это достигается с помощью методов обучения персептрона, в ходе которого меняются компоненты весовых векторов, поступающих на вход персептрона. Последовательность изменений, алгоритм реализации которых будет описан ниже, продолжается до тех пор, пока не будет достигнуто условие правильной классификации. В силу процедуры построения, решающая функция $d(x)$ представляет собой скалярное произведение, а равенство $(w, x) = 0$ определяет положение разделяющей гиперплоскости в пространстве R^n , которая является границей открытого множества, содержащего условие правильной классификации.

Принципиально схема работы персептрона представлена на рис 1. Сигнал $x = (x_1, x_2, \dots, x_n)$ поступает на реагирующий элемент, который вычисляет значение решающей функции и выбирает некоторое действие в зависимости от получаемого результата.

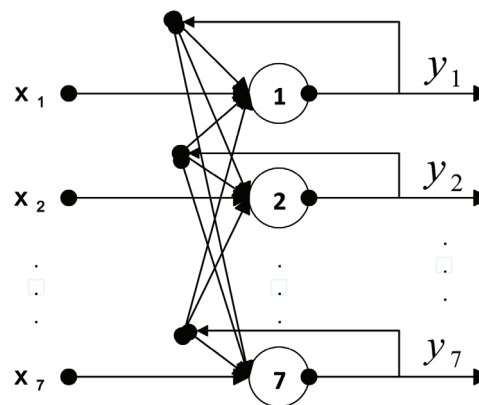


Рис. 1. Архитектура полносвязной нейронной сети

Говоря более конкретно, вектор $x = (x_1, x_2, \dots, x_n)$ скалярно умножается на весовой вектор $w = (w_1, w_2, \dots, w_n)$, если для i -го нейрона величина скалярного произведения (w, x_i) стремится к определенному значению y_i , то весовой вектор не меняется, в противном случае вычисляется величина ошибки и элементы весового вектора пересчитываются. Коррекция весовых векторов может быть произведена различными методами, такими как итерационный метод, метод Хебба, метод обратного распространения

ошибки и т.п. Результат скалярного произведения $(w, x_k) = t$ преобразуется в сигнал $\varphi(t)$ с помощью функции активации, в качестве которой могут быть использованы такие функции, как функция Хэвсиса, функция знака (сигнум), тождественная функция, гиперболический тангенс и т.д. [3].

Таким образом, проблема обучения персептрона практически сводится к задаче построения решающей функции $d_i(x) = (w, x)$, которая бы за конечное число шагов разделяла элементы обучающей выборки, в соответствии с условием правильной классификации $x \in \Omega_i$, если $d_i(x) > 0$ и $d_i(x) \leq 0$, для всех $j \neq i$.

Для реализации построения такой линейной решающей функции с помощью итерационного алгоритма предварительно обучающую выборку запишем в виде бесконечной циклической последовательности $\Xi_\infty = [x_1, \dots, x_7, x_1, \dots, x_7, \dots]$ [3].

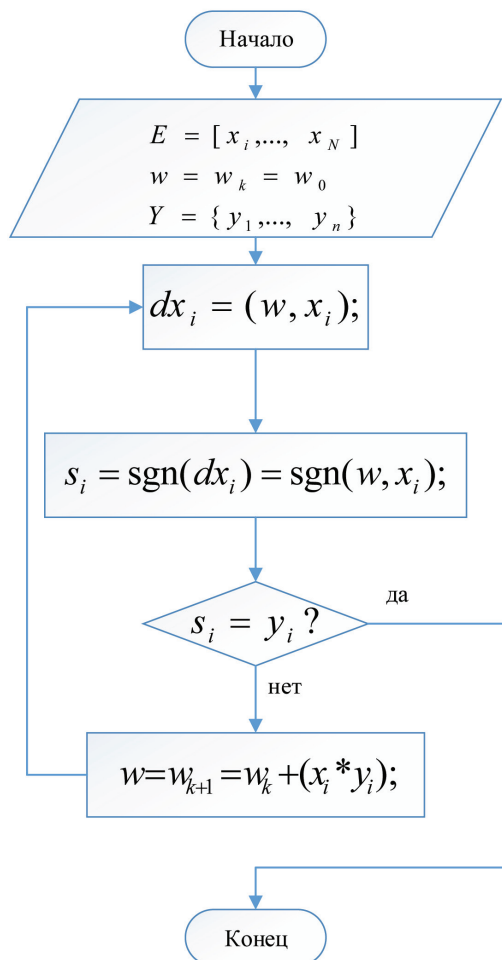


Рис. 2. Блок-схема алгоритма обучения

Идея алгоритма обучения заключается в том, что весовой вектор не меняется, если

«предъявленный» вектор классифицируется правильно и увеличивается или уменьшается на величину x_k при неправильной классификации (рис. 2) [4].

Представленная на рис. 2 последовательность процедур алгоритма обучения осуществляется в следующем порядке:

1. Формируется циклическая последовательность векторов обучающей выборки. Порядок их определяется по убыванию вероятностей, посчитанных с помощью формулы Байеса, например, (см. таблицу), $\Xi_\infty = [x_1, x_2, x_{3,7}, x_4, x_{5,3}, x_{6,5}, x_{7,6}, x_1, x_2, x_{3,7}, x_4, x_{5,3}, x_{6,5}, x_{7,6}, \dots] = (0,25; 0,215; 0,17; 0,143; 0,1; 0,0712; 0,0358)$. В целом такая методика сокращает цикл обучения системы.

2. Устанавливаются начальные значения w_i^0, \dots, w_m^0 весовых векторов всех персептронов, например $(0, 0, \dots, 0)$.

3. Устанавливается множество желаемых выходов $Y = \{y_1, \dots, y_N\}$.

4. Вычисляется решающая функция на выходе нейрона $d_i(x) = (w, x)$.

5. Вычисляются значения, $s_i = \text{sgn}((w, x_i))$ для всех (x_i, y_i) , и если $s_i = y_i$, то алгоритм завершается, в противном случае осуществляется поправка весового вектора по правилу

$$w^{(k+1)} = w^k + x_i y_i$$

и осуществляется переход к пункту 4.

Пример – применение алгоритма Хейбба для обучения одного нейрона классифицировать задачи численных методов

Пусть требуется настроить нейрон на распознавание образов из семи классов, т.е. для заданного множества векторов обучающей выборки $\Xi = [x_1, \dots, x_7]$, где $x_1 = (1, 1, 1, 1, 1, 1) \in \Omega_1$, $x_2 = (0, 1, 1, 1, 1, 1) \in \Omega_2$, $x_3 = (0, 1, 0, 1, 1, 0) \in \Omega_3$, $x_4 = (0, 1, 1, 0, 1, 1) \in \Omega_4$, $x_5 = (0, 0, 0, 1, 1, 0) \in \Omega_5$, $x_6 = (0, 0, 1, 0, 0, 0) \in \Omega_6$, $x_7 = (0, 0, 1, 1, 1, 1) \in \Omega_7$, данную обучающую выборку получили в приведенном примере 2 в работе [1] и проранжировали их по значению весового вектора $w^0 = (0,25; 0,215; 0,17; 0,143; 0,1; 0,0712; 0,0358)$, в результате получилась обучающая выборка с очередностью, установленной в следующем порядке $\Xi = [x_1, x_2, x_{3,7}, x_4, x_{5,3}, x_{6,5}, x_{7,6}]$ (см. таблицу), требуется так настроить синапсы этого нейрона, чтобы он давал на выходе правильные отклики при предъявлении сети векторов обучающей выборки. Эталонным выходом сети должно быть значение $y_i = 1$, если $x_i \in \Omega_1$, и $y_i = -1$, если $x_i \notin \Omega_1$. Таким образом, имеем множество $Y = \{y_1, \dots, y_N\} = \{1, -1, -1, -1, -1, -1, -1\}$ эталонных откликов нейрона. Будем считать, что, входные сигналы являются биполярными, т.е. $x_i = (x_{ij}), x_{ij} \in \{-1, 1\}$,

а в качестве функции нейронов используется функция сигнума $\text{sgn}(t)$, (функция сигнума равняется 1, если $t > 0$, и 0, если $t \leq 0$). Применительно к обучению одного нейрона правило Хебба формируется следующим образом: если нейрон правильно классифицирует вектор, то порог синоптических связей снижается пропорционально этому вектору [3].

Таким образом, выполнением последовательности действий алгоритма обучения, представленная на рис. 1 с начальным нулевым весовым вектором: $w^0 = (0,0,0,0,0,0,0)$ (таблица), получаем следующий результат обучения (таблица).

В таблице ячейки первой строки указывают на шаг итерации. Во второй строке пересчитываются элементы обучающей выборки, в следующих семи строках записаны коэффициенты весового вектора w^k , в следующей строке записан результат вычислений скалярных произведений (w^k, x_i) , в предпоследней строке проверяется условие останова найденного весового вектора w , вычисляются значения $s_i = \text{sgn}((w, x_i))$. Если $s_i = y_i$ для всех $i = 1, N$, то алгоритм прекращает свою работу, в противном случае – переход к пункту 2. В последней строке знак «+» означает, что вектор классифицировался правильно и коррекцию делать не нужно, а знак «-» означает наоборот. После семи подряд идущих правильных классификаций алгоритм завершает свою работу, в результате получаем требуемый весовой вектор $w = (1,0,0,0,0,0,0)^T$.

На рис. 3 приведен скриншот интерфейса разработанного программного обеспечения для обучения системы с помощью нейронной сети (блок-схема, рис. 1). Через этот интерфейс вводятся входные параметры, т.е. векторы признаков объекта с булевыми переменными и начальные веса, а также, в неко-

торых случаях оказывается целесообразным задать итерации. Это делается потому, что иногда процесс обучения продолжается очень долго и ограничение на число итераций мы определяем по системе «уровень (шаг)» завершения обучения, даже если искомым весовой вектор не найден. Через этот же интерфейс система сообщает результат обучения, т.е. текущие весовые векторы в каждом шаге обучения и искомым весовой вектор.

В ходе реализации алгоритма обучения Хебба минимизируется следующий функционал

$$F(w) = - \sum_{x_k \in I} \text{sgn}((w, x_k)) y_k .$$

Его минимальное значение будет равно $(-m)$, где m – количество классов, и равно 7. Следовательно

$$\begin{aligned} F(w) &= -((\text{sgn}((w, x_1))y_1) + (\text{sgn}((w, x_2))y_2) + \\ &+ (\text{sgn}((w, x_7))y_7) + (\text{sgn}((w, x_3))y_3) + \\ &+ (\text{sgn}((w, x_3))y_3) + (\text{sgn}((w, x_5))y_5) + \\ &+ (\text{sgn}((w, x_6))y_6)) = -((1*1) + (-1*-1) + \\ &+ (-1*-1) + (-1*-1) + (-1*-1) + \\ &+ (-1*-1) + (-1*-1)) = \\ &= -(1 + 1 + 1 + 1 + 1 + 1 + 1) = -7. \end{aligned}$$

Это функция заменяется ее «гладкой аппроксимацией» – функцией гиперболического тангенса $\text{th}(t)$, получим функционал

$$\begin{aligned} \tilde{F}(w) &= - \sum_i \text{th}((w, x_i)) y_i = \\ &= -(((\text{th}(w, x_1))*y_1) + ((\text{th}(w, x_2))*y_2) + \\ &+ ((\text{th}(w, x_7))*y_7) + ((\text{th}(w, x_4))*y_4) + \\ &+ ((\text{th}(w, x_3))*y_3) + ((\text{th}(w, x_5))*y_5) + \\ &+ ((\text{th}(w, x_6))*y_6)) = (0,7616), \end{aligned}$$

что позволяет при необходимости уточнить положение минимума.

Результат обучения с начальным нулевым весовым вектором

Номер шага k	0	1	2	3	4	5	6	7	8
векторы	-	x_1	x_2	$x_{3,7}$	x_4	$x_{5,3}$	$x_{6,5}$	$x_{7,6}$	x_1
w^k	0	1	1	1	1	1	1	1	1
	0	1	0	0	0	0	0	0	0
	0	1	0	0	0	0	0	0	0
	0	1	0	0	0	0	0	0	0
	0	1	0	0	0	0	0	0	0
	0	1	0	0	0	0	0	0	0
	0	1	0	0	0	0	0	0	0
(w^k, x_i)	0	0	0	0	0	0	0	0	1
$s_i = \text{sgn}((w, x_i))$		-1	-1	-1	-1	-1	-1	-1	1
Коррекции		-	+	+	+	+	+	+	+

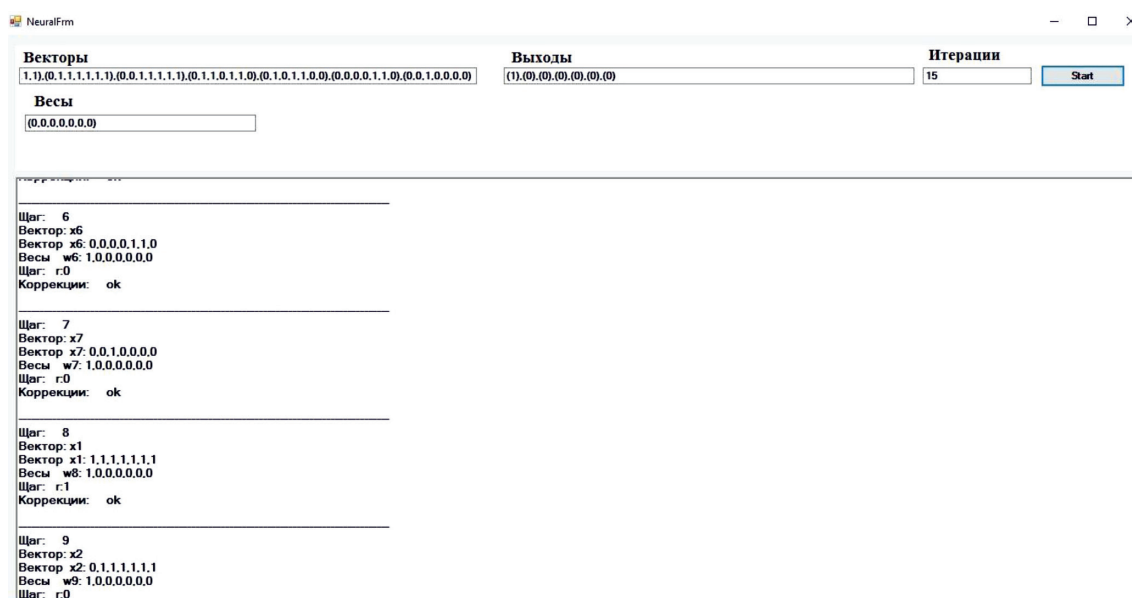


Рис. 3. Скриншоты системы (нейронной сети)

Заключение

В работе были достигнуты следующие результаты:

- Построена математическая модель распознавания образов для SMART-образовательных систем.
- Разработано программное обеспечение для реализации алгоритма обучения и распознавания.
- Приведен пример распознавания и классификации задач из предметной области учебной дисциплины «Численные методы».

В заключение следует отметить, что данная работа является продлением работы [1], в которой приведены входные данные задачи распознавания и подробнее о таблице (таблица). Несмотря на то, что представленные материалы носят образовательный характер, можно отметить, что предложенный алгоритм можно использовать для решения задач большой размерности, что является

их преимуществом перед множеством известных простых методов классификации.

Список литературы

1. Обади А.А., Аль-Хашеди А.А., Нуриев Н.К., Печеный Е.А. Проектирование математической модели и модуля распознавания образов для смарт-обучающей системы // Вестник Казанского технологического университета. – 2017. – Т. 20, № 8. – С. 109–114.
2. Аль-Хашеди А.А., Обади А.А., Нуриев Н.К., Печеный Е.А. Разработка математической модели распознавания запросов-задач коммуникационных услуг // Фундаментальные исследования. – 2017. – № 6. – С. 9–14.
3. Лепский А.Е. Математические методы распознавания образов / А.Е. Лепский, А.Г. Броневич. – Таганрог: Курс лекций, 2009. – 155 с.
4. Чабан Л.Н. Теория и алгоритмы распознавания образов / Л.Н. Чабан. – М.: МИИГАиК, 2004. – 70 с.
5. Мерков А.Б. Распознавание образов [Текст]: Построение и обучение вероятностных моделей; Российская академия наук, Ин-т системного анализа. – М.: URSS, 2014. – 238 с.
6. Патрик Э.А. Основы теории распознавания образов / Пер. с англ.; под ред. Б.Р. Левина. – М.: Советское радио, 1980. – 407 с.