СЕКЦИЯ 5
ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И ПРИНЯТИЕ РЕШЕНИЙ

SECTION 5
ARTIFICIAL INTELLIGENCE AND DECISION MAKING

УДК 004.8

А.Г. ЖИХАРЕВ, В.В. БАСКАКОВА, А.Н. ЗАЙЦЕВ, ЕВ. МАЛКУШ
A.G. ZHIKHAREV, V.V. BASKAKOVA, A.N. ZAITSEV, E.V. MALKUSH

СИСТЕМНО-ОБЪЕКТНОЕ ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

SYSTEM-OBJECT SIMULATION OF ARTIFICIAL NEURAL NETWORKS

Авторы рассматривают применение метода системно-объектного имитационного моделирования в контексте разработки алгоритмов, использующих искусственные нейронные сети. Рассматривается разработка системно-объектной модели искусственного нейрона. Демонстрируется удобство использования программного инструментария UFOModeler в области моделирования искусственных нейронных сетей. В качестве примера рассматривается модель простейшей нейронной сети, состоящей из одного нейрона. Встроенными средствами среды UFOModeler реализован алгоритм обучения искусственного нейрона по правилу Хебба. Демонстрируется целесообразность использования программного инструментария UFOModeler для обучения алгоритмам, основанным на использовании искусственных нейронных сетей.

Ключевые слова: системно-объектная модель, искусственная нейронная сеть, правило Хебба, алгоритм обучения, персептрон, UFOModeler.

In the article, the authors consider the application of the method of system-object simulation in the context of the development of algorithms using artificial neural networks. The development of a system-object model of an artificial neuron is considered. The usability of the UFOModeler software toolkit in the field of artificial neural networks modeling is demonstrated. As an example, we consider a model of the simplest neural network, consisting of one neuron. The built-in tools of the UFOModeler environment implement an algorithm for learning an artificial neuron according to the Hebb rule. The expediency of using the UFOModeler software toolkit for teaching algorithms based on the use of artificial neural networks is demonstrated.

Keywords: system-object model, artificial neural network, Hebb's rule, learning algorithm, perceptron, UFOModeler.

Обучение нейронной сети является процессом с конкретной задачей математического направления [1], в то время как такие задачи можно решать математически, но с возможной визуализацией проделанных уже действий. При помощи инструмента имитационного моделирования UFOModeler [2] есть возможность не только максимально приблизить

визуальное виденье уже смоделированных задач, но и в замедленном режиме симуляции подробно рассмотреть ход данного смоделированного процесса.

На уровне обучения студентов это незаменимый инструмент не только моделирования процессов, но и инструмент визуальной наглядности, крайне важный в учебном процессе студентов. Рассмотрим подробнее пример использования программного инструментария UFOModeler для моделирования простейшей искусственной нейронной сети, состоящей из одного нейрона - персептрона.

Существуют два концептуальных подхода к обучению нейронных сетей: обучение с учителем и обучение без учителя [3]. Обучение нейронной сети с учителем предполагает, что для каждого входного вектора из обучающего множества существует требуемое значение выходного вектора, называемого целевым. Эти вектора образуют обучающую выборку. Веса сети изменяют до тех пор, пока для каждого входного вектора не будет получен приемлемый уровень отклонения выходного вектора от целевого.

Обучение нейронной сети без учителя является намного более правдоподобной моделью обучения с точки зрения биологических корней искусственных нейронных сетей. Обучающее множество состоит лишь из входных векторов. Алгоритм обучения нейронной сети подстраивает веса сети так, чтобы получались согласованные выходные векторы, т.е. чтобы предъявление достаточно близких входных векторов давало одинаковые выходы.

С понятием обучения ассоциируется довольно много видов деятельности, поэтому сложно дать этому процессу однозначное определение. Более того, процесс обучения зависит от точки зрения на него. Именно это делает практически невозможным появление какого-либо точного определения этого понятия. Например, процесс обучения с точки зрения психолога в корне отличается от обучения с точки зрения школьного учителя. С позиций нейронной сети, вероятно, можно использовать следующее определение: обучение - это процесс, в котором свободные параметры нейронной сети настраиваются посредством моделирования среды, в которую эта сеть встроена. Тип обучения определяется способом подстройки этих параметров.

Это определение процесса обучения нейронной сети предполагает следующую последовательность событий:

1. В нейронную сеть поступают стимулы из внешней среды.
2. В результате первого пункта изменяются свободные параметры нейронной сети.
3. После изменения внутренней структуры нейронная сеть отвечает на возбуждения уже иным образом.

Вышеуказанный список четких правил решения проблемы обучения нейронной сети называется алгоритмом обучения. Несложно догадаться, что не существует универсального алгоритма обучения, подходящего для всех архитектур нейронных сетей. Существует лишь набор средств, представленный множеством алгоритмов обучения, каждый из которых имеет свои достоинства. Алгоритмы обучения отличаются друг от друга способом настройки синаптических весов нейронов. Еще одной отличительной характеристикой является способ связи обучаемой нейронной сети с внешним миром. В этом контексте говорят о парадигме обучения, связанной с моделью окружающей среды, в которой функционирует данная нейронная сеть.

Самым важным свойством нейронных сетей является их способность обучаться на основе данных окружающей среды и в результате обучения повышать свою производительность. Повышение производительности происходит со временем в соответствии с определенными правилами. Обучение нейронной сети происходит посредством интерактивного процесса корректировки синаптических весов и порогов. В идеальном случае нейронная сеть получает знания об окружающей среде на каждой итерации процесса обучения.

Также немаловажна возможность анализа эффективности функционирования системы. Наряду с этим важным функционалом, нельзя не заметить возможности в поле действия над одной моделью вывести описание всего моделируемого процесса с учетом тонкостей требуемого уровня декомпозиции. Как со стороны студента, так и на стороне преподавателя очень выгодна и важна возможность при помощи данного имитационного инструмента визуализировать результаты моделирования процесса.

Искусственная нейронная сеть обычно обучается с учителем. Это означает наличие обучающего набора (датасета), который содержит примеры с истинными значениями: тегами, классами, показателями.

Произвольный вид обучения в общем случае - многоэкстремальная невыпуклая задача оптимизации. Для решения этой задачи могут использоваться следующие (итерационные) алгоритмы:

1) алгоритмы локальной оптимизации с вычислением частных производных первого порядка:
   - градиентный алгоритм (метод наискорейшего спуска),
   - методы с одномерной и двумерной оптимизацией целевой функции в направлении антиградиента,
   - метод сопряженных градиентов,
   - методы, учитывающие направление антиградиента на нескольких шагах алгоритма;
2) алгоритмы локальной оптимизации с вычислением частных производных первого и второго порядка:
   - метод Ньютона,
   - методы оптимизации с разреженными матрицами Гессе,
   - квазиньютоновские методы,
   - метод Гаусса-Ньютона,
   - метод Левенберга-Марквардта и др.;
3) стохастические алгоритмы оптимизации:
   - поиск в случайном направлении,
   - имитация отжига,
   - метод Монте-Карло (численный метод статистических испытаний);
4) алгоритмы глобальной оптимизации (задачи глобальной оптимизации решаются с помощью перебора значений переменных, от которых зависит целевая функция).

Рассмотрим подробнее процесс симуляции обучения персептрона для решения задачи классификации изображений, состоящих из девяти пикселей. При чем нейрон должен уметь отнести поданное изображение к одному из двух классов. Форма представления сигналов в рассматриваемом примере - биполярная, т.е. сигнал может принимать значение -1 или 1.

Для создания системно-объектной модели персептрона запустим программный инструментарий UFOModeler и создадим новую модель. Собственно, сам нейрон в терминах системно-объектного моделирования удобно представлять с помощью узла [4], который формально имеет следующий вид:

$$5 = [(L?, L!); f(L?)L!; (O?, O!, Of)], \quad (1)$$

где: (L?, L!) - представляет собой интерфейсные характеристики системы, а именно входящие и исходящие связи; f(L?)L! - функция системы, за счет которой входящие связи преобразуются в выходящие; (O?, O!, Of) - объектные характеристики системы, склюючающие значимые показатели функционирвоания системы.

В качестве связей нейрона будем использовать потоковые объекты модели. Тогда для создания искусственного нейрона с девятью входами и одним выходом разместим на модели следующие графические элементы (рис. 1):

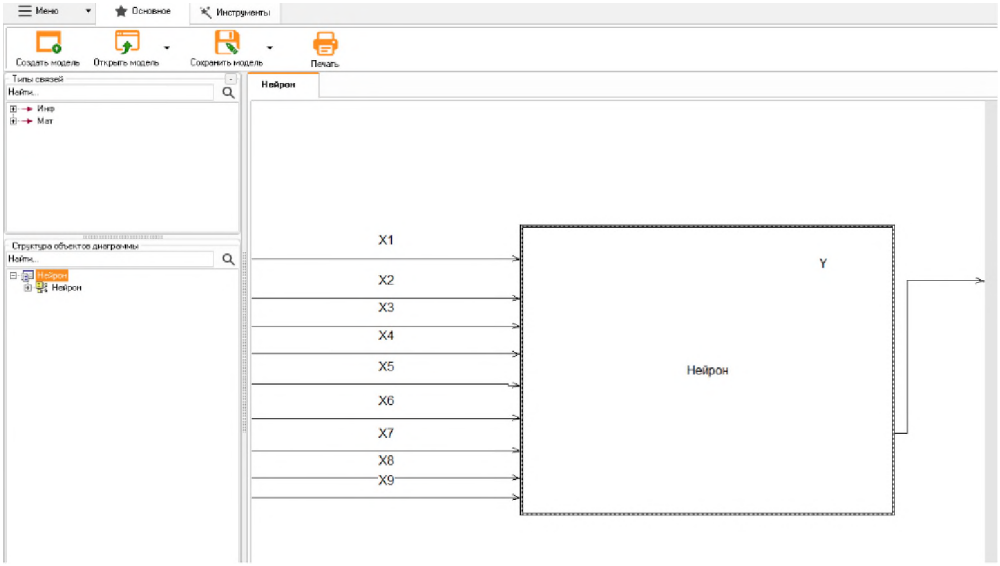

Рисунок 1 - Графическое представление модели искусственного нейрона

Как видно из рисунка 1, нейрон представлен узлом с девятью входными сигналами и одним выходным. Для каждой связи необходимо задать параметры и тип данных для параметров. При чем каждая связь нейрона будет иметь два значения: значение сигнала и вес связи. Параметры связей нейрона представлены на рисунке 2.

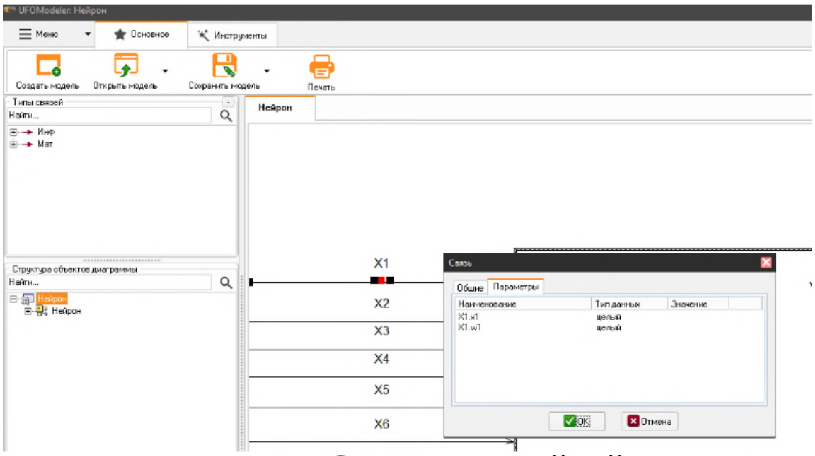

Рисунок 2 - Структура связей нейрона

После реализации структурных характеристик модели необходимо реализовать функцию нейрона с помощью встроенного скриптового языка. Сперва объявляются переменные, которые будут использованы в алгоритме:

```
var
i: integer;
X1,X2, W: array of integer;
f,n,m,s,l;v,y1,y2,sum,sum 1,Y:integer;
```

В алгоритме мы используем три массива. Первый массив X1 представляет собой первое изображение, которому мы будем обучать нейрон по правилу Хебба, X2, соответственно, -второе изображение. Массив W представляет собой веса связей нейрона. Далее необходимо инициализировать соответствующие переменные:

```
begin
n:=9;
setlength(X1,n);
for i:=0 to n-1 do
```

```
X1[i]:=1;
X1[1]:=-1;
X1[6]:=-1;
X1[7]:=-1;
se tlength(X2,n);
for i:=0 to n-1 do
X2[i]:=1;
X2[4]:=-1;
X2[7]:=-1;
```

После инициализации массивов сигналами - изображениями начинаем корректировать весовые коэффициенты нейрона по правилу Хебба.

```
while true do begin
for i:=0 to n-1 do
 W[i]:=0;
y1:=1;
y2:=-1;
f:=0;
setlength(W,n);
while f=0 do begin
for i:=0 to n-1 do
 W[i]:=^1[i] *y1+X2[i] *y2;
sum:=0;
for i:=0 to n-1 do
sum: = sum+W[i] *X1[i];
if sum>0 then Y:=1;
if sum<0 then Y:=-1;
if Y=y1 then f:=1;
if Y<>y1 then f:=0;
sum1:=0;
for i:=0 to n-1 do
sum1: =sum1+W[i] *X2[i];
if sum1>0 then Y:=1;
if sum1<0 then Y:=-1;
if Y=y2 then f:=1;
if Y<>y2 then f:=0;
end;
end;
end.
```

После завершения обучения весовые коэффициенты нейрона принимают нужные веса и, соответственно, можно приступать к использованию нейрона.

Программное средство UFOModeler позволяет запустить модель на исполнение, управляя при этом временным масштабом, что дает возможность в реальном режиме времени отслеживать процесс корректировки весовых коэффициентов в зависимости от выполняемых в данный момент времени действия. Таким образом, системно-объектная модель представляет собой наглядный симулятор процесса обучения и использования искусственного нейрона.

Благодарности

Исследования выполнены при финансовой поддержке проектов Российского фонда фундаментальных исследований № 18-07-00355, 19-07-00290, 19-07-00111, 18-07-00356.

СПИСОК ЛИТЕРАТУРЫ

1. С. Хайкин. Нейронные сети: полный курс. 2-е изд. М., Вильямс, 2006.
2. С.И. Маторин, А.Г. Жихарев Закономерности эволюции систем при изменении функциональных требований // Труды Института системного анализа Российской академии наук. 2020. Т. 70. № 2. С. 30-38.
3. А.И. Галушкин. Нейронные сети. Основы теории. М., Горячая линия - Телеком, 2010.

Жихарев Александр Геннадиевич
Белгородский государственный национальный исследовательский университет, г. Белгород
К.т.н., доцент кафедры информационных и робототехнических систем.
Тел.: 89511559075
E-mail: zhikharev@bsu.edu.ru

Баскакова Валентина Валерьевна
Белгородский государственный национальный исследовательский университет, г. Белгород
Аспирант кафедры информационных и робототехнических систем.
Тел.: 89803204294
E-mail: 522540@bsu.edu.ru

Зайцев Анатолий Николаевич
Белгородский государственный национальный исследовательский университет, г. Белгород
Аспирант кафедры информационных и робототехнических систем.
Тел.: 89803781139
E-mail: zaitsev_an@bsu.edu.ru

Малкуш Елена Викторовна
Белгородский государственный национальный исследовательский университет, г. Белгород
Аспирант кафедры информационных и робототехнических систем.
Тел.: 89205937303
E-mail: malkush@bsu.edu.ru