DOI: 10.17117/na.2016.04.03.211

<http://ucom.ru/doc/na.2016.04.03.211.pdf>

Поступила (Received): 21.04.2016

Цырульник В.Ф., Кадочникова Н.А. Оценка актуальности угроз информационной безопасности с помощью программной реализации обученной нейронной сети

Tsyurulnik V.F., Kadochnikova N.A.
**Evaluation of the relevance of the information security threats with
the help of a software implementation, the trained neural network**

В статье рассматривается метод анализа угроз информационной безопасности с помощью программной реализации обученной нейронной сети. По результатам изучения нейронных сетей была написана программа, реализующая построение обученной нейронной сети для принятия соответствующего решения средств защиты информационной системы при воздействии дестабилизирующих факторов

Ключевые слова: дестабилизирующий фактор, передаточная функция, нейронная сеть, информационная безопасность, порог связи

In the article the method of analysis of information security threats with the help of a software implementation, the trained neural network. The study of neural networks was written the program that realizes the construction of trained neural networks for decision by means of information protection systems under the influence of destabilizing factors

Key words: destabilizing factor, transmission function, neural network, Information Security, connection threshold

Цырульник Валерия Федоровна

Студент

Омский государственный технический университет

г. Омск, пр. Мира, 11

Tsyurulnik Valeria Fedorovna

Student

Omsk state technical university

Omsk, Mira ave., 11

Кадочникова Наталья Александровна

Студент

Омский государственный технический университет

г. Омск, пр. Мира, 11

Kadochnikova Natalia Alexandrovna

Student

Omsk state technical university

Omsk, Mira ave., 11

Важнейшей задачей в обеспечении информационной безопасности является определение и классификация актуальных угроз. В сфере информационной защиты в качестве теоретического базиса могут быть эффективно использованы искусственные нейронные сети. Ниже представлено описание программы по построению обученной нейронной сети, принимающей решения об актуальности угрозы также описан порядок действий по построению обученной нейронной сети.

Первый этап: составление перечня дестабилизирующих факторов.

Ввод данных начинается с ввода перечня дестабилизирующих факторов и действий пользователей (рис.1) – «Параметры → Добавить» события. В появившейся форме AddForm пишется название дестабилизирующего фактора или действия пользователя (далее “событие”). При нажатии кнопки ОК в AddForm программа завершит процесс ввода событий, при нажатии кнопки Add next программа продолжит ввод событий. Нумерация событий автоматическая. При необходимости добавления новых событий повторить действие «Параметры → Добавить события».

Рис. 1. Действие «Параметры → Добавить события» в основной программе и окно добавление событий в форме AddForm

События хранятся в виде списка экземпляров класса “событие”. После ввода множества событий в основной программе в выпадающем меню под надписью события можно выбрать любой из введенных дестабилизирующих факторов и присвоить ему “достоверность” (рис.2). Под “достоверностью” дестабилизирующего фактора понимается достоверность высказывания о том, что произошло данное событие. Для примера $X_1 = ДФ1 = 0,3$; $X_2 = ДФ2 = 0,6$; $X_3 = ДФ3 = 0,4$; $X_4 = ДФ4 = 0,7$.

Рис. 2. Выбор дестабилизирующего фактора из списка (слева) и присвоение ему “достоверности” (справа)

Второй этап: построение логических функций и списка принимаемых решений, выбор передаточной функции.

Для логического решения задач используется следующая передаточная функция (1), определяющая величину V возбуждения нейрона в зависимости от величины V_i возбуждения, связанных с ним нейронов, весов w_i этих связей, а также величину порога h :

$$V := \sum_i w_i V_i;$$

$$V := \text{if } V > h \text{ then } V \text{ else } 0. \quad (1)$$

Для примера возьмем реакцию средств защиты информации на дестабилизирующие факторы:

$$(ДФ1 \wedge ДФ2) \vee (ДФ2 \wedge ДФ4) \rightarrow R_1$$

$$(ДФ1 \wedge ДФ3) \rightarrow R_2$$

Пороги возбуждения соответствующих нейронов входного слоя: $h_1 = 0,2$; $h_2 = 0,9$; $h_3 = 0,3$; $h_4 = 0,5$.

Ввод конъюнкций осуществляется аналогично вводу событий (рис.3). В основной программе жмем «Параметры → Добавить конъюнкции». В появившейся форме AddAnd выбираем дестабилизирующие факторы, участвующие в данной конъюнкции, кнопки OK и Add next в AddAnd действуют как аналогичные кнопки в форме AddForm. В выпадающем меню выставляем порог на конъюнкцию (рис.4).

Рис. 3. Действие «Параметры → Добавить конъюнкции» (слева) и выбор элементов, участвующих в конъюнкции(справа)

Рис. 4. Установка порога на конъюнкцию в которой участвуют элементы ДФ1 и ДФ3

Аналогично добавляются решения, являющиеся дизъюнкциями конъюнкций (рис.5). Установка порогов возбуждения нейронов выходного слоя (рис.6).

Рис. 5. Действие «Параметры → Добавить решения» (слева) и выбор конъюнкций, участвующих в дизъюнкции (справа)

Рис. 6. Установка порога для решения R_2

Третий этап: возбуждение нейронов промежуточного слоя, возбуждение нейронов выходного слоя, расчет принимаемых решений.

После ввода данных, построения логических функций, построения списка решений, после задания порогов получится сеть следующего вида (рис. 7):

Рис. 7. Нейросеть

Значение порога конъюнкции определяется аналогично передаточной функции, где вместо V и V_i соответственно стоит h и $h_i(2)$:

$$h := \sum_i \omega_i h_i \quad (2)$$

Нейроны выходного слоя представляют собой список экземпляров класса LogAnd. В качестве полей этот класс содержит номер нейрона, конъюнкции, входящие в дизъюнкцию, значение порога и указатель на следующий в списке нейрон.

Конъюнкции, содержащиеся в поле конъюнкций нейрона выходного слоя разделены пробелом, т.е. запись имеет вид (X_1, X_2, X_2, X_4). Строка парсится по пробелам и результат записывается в массив строк. Каждый элемент этого массива парсится по знаку “,” (запятая), т.к. элементы, входящие в конъюнкцию разделены запятой. Этим получив номера элементов событий, входящих в конкретную конъюнкцию. Зная число элементов (n), входящих в данную конъюнкцию и все угрозы равнозначны, то $\omega_i = \frac{1}{n}$ – веса связей нейронов. Проверяется список событий, если встречается номер события, входящего в данную конъюнкцию, считываем его “достоверность” и вычисляется уровень возбуждения нейрона, моделирующего конъюнкцию, по формуле 1. Аналогично, зная уровень возбуждения нейрона, моделирующего конъюнкцию, и зная количество конъюнкций, участвующих в дизъюнкции определяется уровень возбуждения нейрона выходного слоя по формуле 1. Если у нейрона выходного слоя уровень возбуждения V не равен 0, то следует принять решение, которое моделирует данный нейрон.

Четвертый этап: построение структуры, обученной нейросети.

Графическое построение сети (рис.8) производится нажатием кнопки “построить сеть”. Нейрон, чей уровень возбуждения больше порога h_i представляется закрашенным бордовым кругом. Нейроны выходного слоя, которые будут окрашены в бордовые цвет и будут теми решениями, которые необходимо принять.

Рис. 8. Графическая интерпретация нейронной сети

Заключение. Представленные результаты работы программы, реализующей работу нейронной сети для принятия решений, наглядно показывают, что нейронная сеть анализирует решение, которое должна принять система защиты при текущем наборе дестабилизирующих факторов каждой угрозы.

Список используемых источников:

1. Комашинский В.И., Смирнов Д.А. Введение в нейроинформационные технологии. СПб.: Тема, 1999. 361 с.
2. Новиков П.С. Элементы математической логики. М.: Государственное издательство физико-математической литературы, 1959. 400 с.
3. Амосов Т.Н. Байдык А.Д. и др. Нейрокомпьютеры и интеллектуальные роботы. К.: Наукова думка, 1991. 271 с.

© 2016, Цырульник В.Ф., Кадочникова Н.А.
Оценка актуальности угроз информационной безопасности с помощью программной реализации обученной нейронной сети

© 2016, Tsyurulnik V.F., Kadochnikova N.A.
Evaluation of the relevance of the information security threats with the help of a software implementation, the trained neural network