

Ф.Г. НЕСТЕРУК, И.В. КОТЕНКО
**ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА
СОЗДАНИЯ НЕЙРОСЕТЕВЫХ КОМПОНЕНТ
ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ ЗАЩИТЫ ИНФОРМАЦИИ**

Нестерук Ф.Г., Котенко И.В. Инструментальные средства создания нейросетевых компонент интеллектуальных систем защиты информации.

Аннотация. На рынке инструментальных средств создания нейросетевых интеллектуальных систем представлено большое количество программных средств, что объясняется многоплановостью задач интеллектуальной обработки информации. В работе предлагается обзор инструментальных средств, применимых для создания нейросетевых компонент интеллектуальных систем защиты информации.

Ключевые слова: информационно-телекоммуникационные системы, системы защиты компьютерных систем, интеллектуальные системы, нейросетевые компоненты, нейронные сети.

Nesteruk F.G., Kotenko I.V. Tools for development of neural components of intelligent security systems.

Abstract. The market of tools for development of neural intelligent systems presented in thousands of software titles, due to multiple tasks of intelligent information processing. The paper provides an overview of tools that are applicable for development of neural components of intelligent security systems.

Keywords: information and telecommunications systems, security systems of computer systems, intelligent systems, neural networks.

1. Введение. На рынке нейросетевых инструментальных средств создания интеллектуальных систем представлено большое количество различных программных средств, что объясняется многоплановостью задач интеллектуальной обработки информации в различных сферах деятельности [2], в том числе для защиты информации в информационно-телекоммуникационных системах [8, 11].

Актуальность проблемы выбора подходящих под решаемую задачу нейросетевых инструментальных средств в области защиты информации обусловлена большим разнообразием представляемых ими функций, возможностей настройки, характеристик и свойств, динамикой их развития, распределенной и разнородной структурой систем защиты информации (СЗИ) и множеством других факторов.

Внедрение перспективных систем защиты информации, использующих средства интеллектуальной обработки информации на базе нейронных сетей, позволяет наделять системы защиты такими качествами как избыточность, адаптивность, возможность эффективной классификации и кластеризации угроз и атак, выполнение вычислений

в реальном времени, повышение достоверности результатов и др. [3, 5, 6, 7].

В работе предлагается обзор инструментальных средств создания интеллектуальных систем, применимых для проектирования нейросетевых компонент интеллектуальных СЗИ [5-6].

Обзор не претендует на полноту, но включает анализ наиболее известного программного обеспечения (ПО) в обозначенной области. Формат описания каждого средства в начальной части содержит название, авторов, сведения о компании-разработчике, ссылку на официальный сайт, актуальность поддержки, и существует ли она. Далее следует описательная часть ПО.

В обзоре содержатся как описания актуального, поддерживаемого производителем ПО, так и «устаревшего», но уникального, и до сих пор востребованного специалистами предметной области, но лишено поддержки по каким-либо причинам.

Статья имеет следующую структуру. В *разделе 2* описываются примеры сфер применения нейронных сетей (НС) для решения задач защиты информации. Дается общая характеристика средств создания нейронных сетей. В *разделе 3* рассматриваются специализированные инструментальные средства создания нейронных сетей. В *разделе 4* представляются многофункциональные средства создания нейронных сетей или программные комплексы, содержащие инструментарий, позволяющий моделировать нейросетевые компоненты. В *заключении* обобщены выводы по проделанной работе. Отмечено несколько динамично развивающихся программных средств, содержащих наиболее широкий спектр возможностей, в достаточной мере отвечающих потребностям различных пользователей, в том числе, для целей моделирования и анализа нейросетевых компонент защиты информации в информационно-телекоммуникационных системах [8].

2. Общая характеристика нейронных сетей и их применение для задач защиты информации. Средства создания нейронных сетей (НС), как правило, включают средства моделирования НС, называемые имитаторами или *нейропакетами*, средства обучения и графический пользовательский интерфейс. Подобное ПО реализует ряд обязательных функций, таких как формирование модели НС, обучение и тестирование НС [4].

Рассмотрим эти функции.

Формирование моделей нейронных сетей. Модели НС определяются моделями формальных нейронов (ФН) и структурой межнейронных связей.

В зависимости от структуры связей в нейропакетах обычно реализуются следующие *топологии* НС:

- НС *многослойная* (сеть из ряда слоев формальных нейронов);
- НС *полносвязная* (каждый формальный нейрон связан со всеми остальными ФН нейронной сети; в каждом такте на входы НС подаётся входной сигнал + значения выходов ФН, сформированные за предыдущий такт);
- НС *с локальными связями* (ФН располагаются в узлах прямоугольной или гексагональной решетки сети. Каждый нейрон связан с ближайшими соседними);
- НС *неструктурированная* (модель сети, не имеющая структуры, подходящей к описанным выше группам).

Для построения НС под конкретные задачи, применяются процедуры формирования нейронных сетей, использующие распространенные топологии НС, и известные модели формальных нейронов, с различными вариантами нелинейных преобразований или функций активации: сигмоидальных, пороговых, кусочно-линейных, и др.

Группы моделей НС различимы по *применению* для решения конкретных задач:

- НС *многослойные и полносвязные с сигмоидальными функциями активации* (для распознавания образов, адаптивного управления);
- НС *с локальными связями* (при обработке изображений);
- НС *многослойные с особыми функциями активации* (для решения задач линейной алгебры).

Большинство задач *распознавания, классификации образов и управления* сводятся к построению многопараметрических отображений.

НС *двухслойные* с сигмоидальными функциями активации, на которых можно реализовать произвольные функции отображения входных сигналов в выходные [1, 9], имеют теоретическое обоснование.

Обучение нейронных сетей. В большинстве нейропакетов используются стандартные процедуры обучения НС, ориентированные на конкретные нейропарадигмы.

Зачастую в нейропакетах возможно задание различных типов данных и размерности входных и выходных сигналов в зависимости от решаемой задачи.

Входными данными могут быть: изображения, числа, распределения значений, а типом входных данных – числа на заданном диапазоне: бинарные, биполярные, целые или действительные. Выходными

сигналами НС являются: целые или действительные числа, или их вектора.

При решении практических задач обычно используются обучающие выборки, что предусмотрено в ряде нейропакетов в виде средств их формирования. Среднеквадратичное отклонение используется, в большинстве случаев, в качестве функции ошибки и позволяет определить соответствие выходных сигналов НС желаемым выходным сигналам обучающей выборки. В ряде нейропакетов существует возможность выбора либо задания функции ошибки.

Реализуемые в нейропакетах алгоритмы обучения НС делятся на 3 группы:

- градиентные,
- стохастические,
- генетические.

В градиентных алгоритмах вычисления частных производных функции ошибки производятся по параметрам НС.

В стохастических алгоритмах минимум функции ошибки ищется случайным образом.

В генетических алгоритмах сочетаются свойства градиентных и стохастических алгоритмов (перебор вариантов реализуется на основе операции наследования, градиентный спуск реализуется на основе операции отбора).

Наличие процедур инициализации НС перед обучением предусмотрено в нейроимитаторах (присваивание параметрам НС начальных значений).

Критерии остановки при обучении НС обычно используются следующие: когда значения функции ошибки достигают заданного; когда успешно решаются все примеры обучающей выборки, где обучение – итерационная процедура. Одна из самых важных характеристик ПО для моделирования НС – скорость сходимости алгоритма обучения.

Тестирование обученных НС. В нейроимитаторах предусмотрены средства тестирования для верификации обученной НС. Подаются входные сигналы, которые зачастую не совпадают с обучающей выборкой и проверяют правильность результата, формируемого на входы НС. Тестирование НС проводится на одиночных входных сигналах либо на тестовой выборке, состоящей из пар (желаемые входные сигналы, желаемые выходные сигналы). Сама тестовая выборка строится индивидуально для каждой решаемой задачи.

Нейронные сети могут использоваться для **решения различных задач защиты информации** [5, 6, 8, 11, 38, 40].

В основном – это задачи классификации, распознавания и кластеризации при обнаружении атак, вторжений и вредоносных программных объектов.

Другими очень важными задачами являются защищённое избыточное хранение информации, поиск скрытых закономерностей в массивах данных, биометрическая идентификация и аутентификация, шифрование данных и др.

Например, при биометрической идентификации может осуществляться преобразование совокупности экземпляров биометрических образов человека, позволяющее описать их стационарную и случайную составляющие (в виде математического ожидания и дисперсий контролируемых параметров или в виде параметров обученной сети искусственных нейронов). При преобразовании «биометрия-код» зараннее обученная искусственная нейронная сеть осуществляет трансформацию вектора входных биометрических параметров «свой» в код криптографического ключа или преобразует любой другой случайный вектор входных данных в случайный выходной код.

В качестве тестовой задачи для проверки инструментального средства может быть выбрана, например, задача прогнозирования сетевой атаки или многомерного временного ряда, например, на основе использования многослойной НС.

Оценку инструментальных средств удобно проводить, исходя из возможности использования различных нейросетевых структур, критериев оптимизации и алгоритмов обучения НС, а также простоты использования нейропакетов и наглядности представления информации. При тестировании разумно учитывать сферу применения, функционал использования ПО, для разработки нейросетевых систем, широту реализуемых возможностей: создания, обучения НС, интерфейса; подготовки обучающих выборок; целостности, конфиденциальности, доступности представляемой информации при создании, обучении НС; применения нейросетевых парадигм, критериев, алгоритмов обучения НС; создания пользовательских нейросетей, использования пользовательских критериев оптимизации и алгоритмов обучения НС; обмена данными между нейропакетом и приложениями операционной системы; расширения нейропакета за счет своих программных модулей; генерации исходных кодов; возможности по интеграции с другим ПО.

Представляется, что вопрос о пригодности того или иного программного средства для конкретной задачи защиты информации заключается также в наличии и/или превосходстве какой-либо функции (например, вида обучения, тестирования, переобучения или «дообуче-

ния» сети) или возможности задания определенных параметров (например, видов задаваемых сетей, их связей, количества входов, выходов, весов связей, количества слоёв, число эпох обучения сети, и т.п.).

У исследователя может существовать необходимость в моделировании только на одном виде нейронной сети – тогда ему достаточно использовать конкретный узкоспециализированный инструмент. Наоборот, в ситуации, когда нужно исследовать модель на различных нейронных сетях с различными параметрами требуется использовать универсальный программный комплекс.

3. Специализированные инструментальные средства создания нейронных сетей. Нейропакет *SNNS (Stuttgart Neural Network Simulator)* [25, 26] разработан в Институте параллельных и распределенных высокопроизводительных систем (*IPVR*) Университета Штутгарта (*University of Stuttgart*) и поддерживается университетом Тюбингена (*University of Tübingen*). Он изначально представлял собой программный имитатор для нейронных сетей на рабочих станциях Unix. В данный момент доступны версии под *MS Windows*. Целью проекта *SNNS* является создание эффективной и гибкой среды моделирования для исследования и применения нейронных сетей. *SNNS* включает два основных компонента: эмулятор ядра, написанный на *C*, и графический интерфейс пользователя. Эмулятор ядра работает с внутренними сетями структур данных нейронных сетей и выполняет операции обучения и переобучения. Он может быть использован как программа на *C*, встроенная в пользовательские приложения. *SNNS* может расширяться с помощью определенных пользователем функций активации, выходных функций, функций сайта и процедур обучения, которые записываются в виде простых программ на *C* и связываются с эмулятором ядра.

Нейропакет *JavaNNS: Java Neural Network Simulator* [25], разработанный в университете Тюбингена (*University of Tübingen*), является преемником *SNNS*. Он основан на собственном вычислительном ядре, обладает вновь разработанным графическим интерфейсом пользователя на *Java*, который совместим с нейропакетом *SNNS* и позволяет увеличить платформенную независимость.

Версии нейропакета *Emergent* (ранее *PDP + +*) [26, 27] разрабатывались с 1995 года в Университете Карнеги-Меллона (*Carnegie Mellon University*), а с 2010 года – в университете Колорадо в Боулдере (*University of Colorado at Boulder*). *Emergent* применяется для моделирования и предназначен в первую очередь для создания сложных мо-

делей мозга и имитации когнитивных процессов, но также может быть использован для решения любой задачи, применимой к теории нейронных сетей. *Emergent* имеет модульную конструкцию, основанную на принципах объектно-ориентированного программирования. Он функционирует на платформах *Microsoft Windows*, *Darwin / Mac OS X* и *Linux*. *Emergent* использует C++-подобный интерпретируемый язык сценариев *C-Super-Script*, который позволяет получить доступ практически ко всем объектам моделирования.

Нейропакет *Neural Lab*, реализованный Серджио Ладесма (*Sergio Ledesma*) из *University of Guanajuato* [26, 28], предоставляет визуальную среду для разработки и тестирования искусственных нейронных сетей, интегрируется с *Microsoft Visual Studio*, с C++, включает искусственные нейронные сети (ИНС) в пользовательские приложения, научно-исследовательское моделирование, интерфейсы конечных пользователей. *Neural Lab* выполняет проверку на наличие ошибок наборы данных, прежде чем использовать их для обучения, сохраняя значительное количество времени. Интерфейс активации позволяет тестировать ИНС в режиме реального времени, что делает возможным сравнить визуально фактический выход нейронной сети с желаемым.

Нейропакет *NEURON*, разработанный в Йельском университете и университете Дьюка (США) [26, 29], представляет собой среду моделирования отдельных нейронов и сетей нейронов.

NEURON позволяет моделировать отдельные нейроны через использование секций, которые автоматически подразделяются программой вместо их ручного создания пользователем. Основной язык сценариев, который используется для взаимодействия внешнего мира с *NEURON* свой собственный, но интерфейс Python также доступен при необходимости. Данные в программу записываются в интерактивном режиме или загружаются из файла. *NEURON* поддерживает распараллеливание по протоколу *MPI*, возможное с версии 7.0, через внутренние многопоточные процедуры для использования на компьютерах с несколькими ядрами. Свойства мембраны каналов нейрона моделируются с использованием собранных механизмов, написанных с использованием языка *NMODL* или скомпилированных процедур, работающих с внутренними структурами данных, созданных с помощью инструмента *GUI (Channel Builder)*.

NEURON наряду с аналогичной программной платформой *GENESIS* используется в качестве основы для обучения ИНС по всему миру.

Нейропакет *GENESIS* (*The GEneral NEural Simulation System – Общяя нейронная система моделирования*), изначально был разработан в Калифорнийском технологическом институте (*CalTech*) [30]. Он является средой моделирования для построения реалистичных моделей нейробиологических систем разных уровней приближения, в том числе субклеточных процессов, отдельных нейронов, сети нейронов и нейронных систем. Главной целью компьютерных реализаций моделей *GENESIS* является ответ на вопрос: «что известно об анатомическом строении и физиологических особенностях нейронной системы?». *GENESIS* предназначен для количественной оценки нейронной системы таким образом, чтобы было легко понять физическую структуру нейрона. *GENESIS* позволяет распараллелить моделирование отдельных нейронов и сетей по принципу «multiple-instruction-multiple-data parallel computers» (множество команд - множество данных на параллельных компьютерах) [26].

Нейропакет *NEST* [31] в настоящее время развивается и поддерживается по инициативе *NEST*. *NEST* – это программное обеспечение для моделирования пиков моделей нейронных сетей.

Нейропакет *Brian* [32] является программой с открытым исходным кодом Python для разработки, моделирования сетей пиков нейронов.

Нейропакет *XNBC* [33] является инструментом моделирования биологических нейронных сетей с открытым кодом.

Существует множество нейропакетов, специально созданных для решения задач различных фирм-производителей [24, 34].

Нейропакет *Braincel* от *Promised Land Technologies Inc.* [12] представляет собой программную надстройку для табличной среды *MS Excel*. Он позволяет реализовывать НС прямого распространения с одним-двумя скрытыми слоями. *Braincel* использует графические возможности табличного процессора *MS Excel* и алгоритм обучения *BackPercolation*. Он позволяет обрабатывать числовые и символьные данные. *Braincel* предоставляет возможности по определению значимости входов, выбору НС с лучшей структурой и использованию для прогноза одновременно нескольких НС.

Нейропакет *Excel Neural Package* от *HeipOK* [13] дополняет возможности *MS Excel*, по работе с нейросетевыми алгоритмами. Пакет состоит из 2-х компонент: 1. *Winnet*, предназначенный для поиска и моделирования скрытых зависимостей в данных, реализует функции многослойного персептрона, имеет дружественный графический интерфейс, визуальные средства контроля, обучения, предсказания НС, тестирования, остановку обучения при достижении различных крите-

риев 2. *Kohonen Map*, предназначенный для анализа и построения самоорганизующихся карт Кохонена, решающих задачи кластеризации, визуализации. Пакет *Excel Neural Package* удобен при визуализации результатов в виде двумерной карты, но имеет ограниченный функционал.

Нейропакет *Neural Planner* от *Steve Wolstenholme* [15] – оболочка моделирования НС различной конфигурации в среде *MS Windows* использующая 2 алгоритма обучения НС с учителем: *On-Line Back Propagation*, и *Batch Back Propagation*. Нейропакет поддерживается, и сайт находится в актуальном состоянии. *Neural Planner* позволяет решать задачи классификации, обработки данных, некоторые математические задачи, создавать экспертные системы. Основными компонентами ПО являются: графический, и табличный редакторы [4]. В *Neural Planner* удобно создавать сложно топологические (например, рекуррентные) НС при ограниченном числе алгоритмов обучения.

Свободно распространяемый нейропакет *NeuroPro*, созданный в Институте вычислительного моделирования *СО РАН* [16], предназначен для классификации, прогнозирования, извлечения знаний из данных с помощью НС, в среде *MS Windows*. Среда позволяет работать с данными в форматах: *Clipper, dBase, FoxBase, FoxPro, Paradox*; создавать слои НС, одновременно решать ряд задач прогнозирования.

Обучение НС производится с применением одного из нескольких методов оптимизации. Среда позволяет производить тестирование НС, вычисление, значимости входных сигналов НС, случайные изменения весов синапсов ФН, упрощения НС. Нейропакет русскоязычен, удобен в использовании, имеет возможности упрощения НС, выявления значимых входов, но возможность сохранения результатов обученной НС недоработана.

Нейропакет *Partek Discovery*, разработанный в *Partek, Inc.* [17], создан для нейросетевого анализа данных и моделирования НС. Компонент *Partek Discovery* обеспечивает визуальную среду и реализацию алгоритмов числового анализа кластеров данных, что полезно при сжатии многомерных данных для последующей визуализации, анализа или моделирования. Компоненты *Partek Discovery* позволяют проводить прогнозирующее моделирование определения оптимального набора переменных, например, использующий статистические методы, НС, генетические алгоритмы.

Нейропакет *QwikNet32* [18] предназначен для работы в среде *MS Windows*, реализует многослойную сеть прямого распространения (до 5 скрытых слоев, с набором из 6 модификаций алгоритма обратного

распространения ошибки). По умолчанию работает алгоритм *Online Backprop*, при котором веса и смещения НС корректируются после предъявления каждого вектора обучающей выборки.

QwikNet32 компактен, удобен в использовании за счёт возможностей: графики, сохранения обученной НС на языке C, моделирования, обучения НС, ручного масштабирования. Авторская поддержка и актуальная версия не обнаружены.

Нейропакет *STATISTICA Automated Neural Networks* от *StatSoft* [19] является технологически развитым ПО для разработки нейросетевых приложений, предназначенным для специалистов (предложен широкий набор типов НС и алгоритмов обучения) и неподготовленных пользователей в области нейросетевых вычислений.

Нейропакеты *THINKS* и *ThinksPro*, созданные в компании *Logical Designs Consulting Inc.* [20], являются комплексом инструментальных средств для разработки нейросетевых систем. Компонент *THINKS* – нейросетевая среда разработки приложений, заточенная как инструмент обучения, за счет наличия опций выбора нейросетевой архитектуры и определения блока обработки, оператор может работать с новыми сетевыми конфигурациями. *ThinksPro* – нейросетевая среда разработки для динамической визуализации в различных форматах значений входов, состояний, весов, выходов.

4. Многофункциональные программные комплексы с возможностями создания нейронных сетей. Используя данные, приведенные в [2 - 4, 10, 11, 35], рассмотрим ряд многофункциональных программных комплексов, включающих возможности нейропакетов.

Нейропакет *MATLAB Neural Network Toolbox* от *MathWorks* [14] является компонентом пакета *MATLAB* и имеет модульную, открытую и расширяемую архитектуру, обеспечивающую поддержку для многих нейросетевых парадигм. Он включает многослойный персептрон с методом обратного распространения ошибки, рекуррентные сети, НС с соревновательными слоями и самоорганизующиеся карты *SOM*.

Среда *Neural Network Toolbox* имеет дружественный графический пользовательский интерфейс разработки, исследования и управления процессами в режимах обучения и работы НС. Проект поддерживается производителем, сайт актуален.

Система *RapidMiner* от *Rapid-I* [37] является системой интеллектуального анализа данных.

Перечислим базовые особенности *RapidMiner*:

- открытый исходный код;

- работает на всех основных платформах и операционных системах;
- многослойное представление данных;
- используется графический режим;
- простой механизм расширения;
- возможность построения мощных, многомерных объектов;
- доступно более 500 операторов для интеграции и преобразования данных, интеллектуального анализа данных, оценки и визуализации;
- автоматические схемы мета оптимизации;
- возможность определения повторно используемых блоков;
- стандартизированный XML-формат обмена для процессов;
- библиотека методов интеллектуального анализа данных *WEKA* полностью интегрирована;
- доступ к источникам данных (*Excel, Access, Oracle, IBM DB2, Microsoft SQL Server, Sybase, Ingres, MySQL, Postgres, SPSS, DBase* и др.).

Пакет ***BrainMaker*** от *California Scientific Software* [21] является нейросетевым ПО, используемым для распознавания образов, медицинской диагностики, бизнеса и т.д.

BrainMaker предназначен для построения многослойных НС с алгоритмом обучения *Back Propagation* (по алгоритму обратного распространения ошибки). Он включает программу *NetMaker (NM)* подготовки и анализа, преобразования исходных данных (из файлов табличных и текстовых форматов, например, *dBase, MS Excel*), предназначенную для анализа данных, создания входных файлов для программы *BrainMaker*, и саму программу построения, обучения, тестирования НС с набором утилит – *BrainMaker (BM)*. Программа способна обрабатывать выходные данные НС с последующим выводом статистики её обучения, тестирования.

Назначение компонента *BrainMaker* варьируется от построения, обучения нейронной сети в различных режимах до модификации параметров НС, и обладает функциями оптимизации процесса обучения, реализации методов анализа выходных данных НС по критерию чувствительности к разнообразию входных данных, формирования отчетов для оценки степени зависимости входных и выходных параметров, системой команд пакетного запуска, интерфейсом включения обученных НС в пользовательские программы.

Возможности *BrainMaker*:

- ввод количества, размера скрытых слоев (по умолчанию число ФН в скрытых слоях равно числу входов);
- определение коэффициента скорости обучения (при разных степенях обученности НС предусмотрены линейная или экспоненциальная зависимости убывания коэффициента);
- реализация динамического сокращения числа ошибок обучения (вначале нейронная сеть обучается с заданной точностью, затем, по выполнению условия адекватности выходов, допустимое значение ошибки умножается на понижающий множитель);
- контроль зависания НС (визуальный контроль за распределением весов НС, для скрытых и выходных слоев; хорошие способности к обучению; опция автоматического уменьшения изменения весов, при критических значениях);
- включение шумов в процесс обучения (умножение отдельных величин вектора на малую константу, сильно увеличивает способность НС к обобщению);
- привнесён элемент случайности в процессы связанные с обучением (создание, обучение нескольких НС; ручная оптимизация модели НС).

Моделирование в *BrainMaker* производится вручную, приводя к значительным временным затратам при создании моделей средней сложности, что компенсируется множеством настроек алгоритма обучения, но, в связи с запутанностью интерфейса, при продолжительной работе оператор может ошибиться в последовательности операций.

Пакет *NeuroShell 2* от *Ward Systems Group Inc.* [39] представляет собой нейропакет, объединяющий известные нейросетевые архитектуры, графический интерфейс оператора, сложные утилиты и популярные опции для управления нейросетевой экспериментальной средой. Пакет включает следующие компоненты: *NeuroShell Predictor*, *NeuroShell Classifier* и *NeuroShell Trader*.

Пакет *NeuroSolutions* от *NeuroDimension Inc.* [22] является ПО, объединяющим модульный графический сетевой интерфейс с выполнением процедур обучения, таких как *Back Propagation* и *Back Propagation with time*. Среда *NeuroSolutions* включает графический интерфейс оператора, генератор исходных кодов C++, и содержит три главных компонента: *Educator* – компонент педагога для изучения нейронных сетей; *Users* – компонент оператора, расширяющий возможности *Educator* за счёт нейросетевых моделей распознавания образов; *Consultants* – компонент консультанта, дополнительно предлагающий мо-

дели динамического распознавания образов, предсказания временных рядов, управления производственными процессами.

NeuroSolutions предназначен для моделирования большого набора НС. Его достоинство – гибкость применения, так как помимо традиционных нейросетевых парадигм (полносвязных и многослойных НС, самоорганизующихся карт Кохонена) нейропакет включает в себя редактор визуального проектирования НС, позволяющий создавать топологии НС и алгоритмы их обучения, а также вводить собственные критерии обучения. *NeuroSolutions* имеет развитые средства визуализации, выводящие его на уровень систем автоматизированного проектирования, моделирования НС.

Пакет предназначен для работы под *MS Windows*. Помимо средств взаимодействия с операционной системой по технологии *OLE*, нейропакет снабжен генератором исходного кода, позволяет использовать внешние модули при создании, обучении НС. Пакет поддерживает программы, написанные на языке *C++* для компиляторов *MS Visual C++* и *Borland C++*, а также в виде кода *DLL*. *NeuroSolutions* является гибкой дополняемой, модифицируемой, открытой системой, содержащей встроенный макроязык, для настройки под задачу.

В пакете поддерживаются типы формальных нейронов, включая ФН первого порядка (взвешенный сумматор), ФН высших порядков (с перемножением входов), ФН непрерывный интегрирующий. Функция активации ФН может выбираться из пяти функций (3 сигмоидальных, кусочно-линейная, функция знака) и задаваться оператором. На этапе проектирования задаются связи (прямые, перекрестные, обратные) между ФН, которые могут быть изменены в процессе работы. Разумно реализована схема организации связей, в плане возможности задания одной лишь векторной связи, с заданной весовой матрицей, а не набора связей с весовыми коэффициентами.

Пакет *NeuralWorks Professional II/Plus* от *NeuralWare, Inc.* [23] представляет собой инструментальное средство для моделирования НС, реализации десятков нейросетевых парадигм и большого количества алгоритмов обучения. С помощью модуля *UDND (User Define Neural Dynamics)* можно создавать новые структуры нейросетей.

В *NeuralWorks Professional 2+* представлены возможности по визуализации данных: топологии НС, ошибок обучения, весов связей, их корреляции при обучении (особенность пакета, упрощающая анализ НС). Нейропакет имеет встроенный генератор кода, поддерживающий компилятор *MS Visual C++*, и возможность интегрировать в него внешние программные компоненты.

Пакет *Process Advisor* от *AIWare Inc.* [4, 36] является нейропакетом, предназначенным для решения задач управления динамическими процессами (например, управления внешними аппаратными контроллерами, подключаемыми к PC). В нём реализована многослойная НС прямого распространения, обучаемая с помощью модификации алгоритма с обратным распространением ошибки, предусмотрена работа с входными сигналами как с функциями времени.

По функциональным возможностям нейропакеты можно распределить следующим образом: *MATLAB Neural Network Toolbox – 1*, *RapidMiner – 2*, *NeuroSolutions – 3*, *NeuralWorks Professional II/ Plus – 4*, *NeuroShell 2 – 5*, *Process Advisor – 6*, *BrainMaker Pro – 7*.

Заключение. В обзоре содержится анализ инструментальных средств создания средств интеллектуальной обработки информации, которые могут использоваться в нейросетевых компонентах защиты информации в информационно-телекоммуникационных системах.

Интеллектуальные средства на базе нейронных сетей в составе гибридных систем защиты позволяют использовать принципы *подобия биосистемам*, в частности адаптации и иммунной защиты для обеспечения реализации механизмов самоорганизации.

Приведенные характеристики нейропакетов и их сравнение с точки зрения простоты использования и спектра предоставляемых услуг для моделирования НС дают возможность из большого числа существующих инструментальных средств выбрать подходящее для решения конкретных практических задач защиты информации.

Анализ инструментальных средств показал, что каждое специализированное средство по своему уникально и способно решить заданную частную задачу, но такие средства, как *MATLAB* и *RapidMiner* имеют больший набор функций и возможности по технической поддержке [14, 37]. Отметим, что появляются новые интересные проекты и программы, подлежащие исследованию, в концепции которых заложен открытый исходный код. Это, например, программное средство *Sage* [24], распространяемое на условиях лицензии *GPL* и сочетающее функциональные возможности существующих пакетов с открытым кодом.

Литература

1. Горбань А. Н., Дунин-Барковский В. Л., Кирдин А. Н. и др. Нейроинформатика // Новосибирск: Наука. Сиб. отд-ние, 1998. 296 С.
2. Дюк В., Самойленко А. Data Mining: учебный курс (+ CD) // СПб: Питер, 2001. 368 С.
3. Искусственные иммунные системы и их применение. // Под ред. Д. Дасгутты. Пер. с англ. под ред. А.А. Романохи. — М.: ФИЗМАТЛИТ, 2006. 344 С.

4. *Круглов В. В., Борисов В. В.* Искусственные нейронные сети. Теория и практика. - 2-е изд., стереотип. – М.: Горячая линия - Телеком, 2002. 382 С.
5. *Котенко И.В., Шоров А.В., Нестерук Ф.Г.* Анализ биоинспирированных подходов для защиты компьютерных систем и сетей // Труды СПИИРАН. Вып.3 (18). СПб.: Наука, 2011. С.19–73.
6. *Нестерук Ф.Г., Суханов А. В, Нестерук Л. Г, Нестерук Г. Ф.* Адаптивные средства обеспечения безопасности информационных систем // Монография. СПб.: Изд-во Политехнического университета, 2008. 626 С.
7. *Нестерук Л. Г., Розов Е. С., Нестерук Г. Ф., Нестерук Т. Н.* Применение модели адаптивной защиты для оценки инвестиционных проектов систем информационной безопасности // Известия вузов. Приборостроение. 2007. т. 46, № 5. С.40–46.
8. *Котенко И.В., Юсупов Р.М.* Перспективные направления исследований в области компьютерной безопасности // Защита информации. Инсайд, № 2, 2006. С.46–57.
9. *Michalewicz I.*, Genetic Algorithms + Data Structures = Evolution Programs // Springer-Verlag, 1992. 387 P.
10. *Hecht-Nielsen R.* Kolmogorov's Mapping Neural Network Existence Theorem // IEEE First Annual Int. Conf. on Neural Networks, San Diego, 1987. V. 3, P. 11–13
11. *Котенко И.В.* Интеллектуальные механизмы управления кибербезопасностью // Управление рисками и безопасностью. Труды Института системного анализа Российской академии наук (ИСА РАН). Т.41, Москва, URSS, 2009. С.74-103.
<http://www.patentgenius.com/patent/5241620.html>
12. <http://www.neurok.ru>
13. <http://www.mathworks.com/products/neuralnet/>
14. <http://www.easynn.com/>
15. http://www.kinnet.ru/cterra/445/18040_2.html
16. <http://www.partek.com/html/products/products.html>
17. <http://retro.icequake.net/numbers/www.seven7seven.com/f351-375.htm>
18. <http://www.statsoft.com/products/statistica-automated-neural-networks/>
19. <http://www.logicaldesigns.com/ThinksPro.htm>
20. <http://www.calsci.com/>
21. <http://www.nd.com/>
22. http://www.neuralware.com/products_pro2.jsp
23. <http://www.sagemath.org/>
24. <http://www.ra.cs.uni-tuebingen.de/SNNS/>
25. http://en.wikipedia.org/wiki/Neural_network_software
26. http://grey.colorado.edu/emergent/index.php/Main_Page
27. <http://www.dicis.ugto.mx/profesores/sledesma/documentos/index.htm>
28. <http://www.neuron.yale.edu/neuron/>
29. <http://genesis-sim.org/>
30. http://www.nest-initiative.org/index.php/Software>About_NEST
31. <http://briansimulator.org/>
32. <http://www.b3e.jussieu.fr/xnbc/>
33. http://en.wikipedia.org/wiki/Neural_network_software/
34. <http://www.radiotec.ru/catalog.php?cat=jr7>
35. <http://members.core.com/17/27/pegasustec/aipe.html>
36. <http://rapid-i.com/content/view/181/196/> (дата обращения 14.02.2013)
37. <http://bookos.org/book/803863>
38. <http://www.wardsystems.com>
- 39.

40. [Электронный ресурс. ФСТЭК России. ГОСТ Р 52633.0-2006] — Режим доступа — URL: http://www.posoh.ru/auto_ident/metki/doc/52633.0-2006.doc (дата обращения 05.04.2013)

Котенко Игорь Витальевич — д.т.н., проф.; заведующий лабораторией проблем компьютерной безопасности, СПИИРАН. Область научных интересов: безопасность компьютерных сетей, в том числе управление политиками безопасности, разграничение доступа, аутентификация, анализ защищенности, обнаружение компьютерных атак, межсетевые экраны, защита от вирусов и сетевых червей, анализ и верификация протоколов безопасности и систем защиты информации, защита программного обеспечения от взлома и управление цифровыми правами, технологии моделирования и визуализации для противодействия кибер-терроризму, искусственный интеллект, в том числе многоагентные системы, мягкие и эволюционные вычисления, машинное обучение, извлечение знаний, анализ и объединение данных, интеллектуальные системы поддержки принятия решений, телекоммуникационные системы, в том числе поддержка принятия решений и планирование для систем связи. Число научных публикаций — более 450. ivkote@comsec.spb.ru, www.comsec.spb.ru; СПИИРАН, 14-я линия В.О., д.39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328-2642, факс +7(812)328-4450.

Kotenko Igor Vitalievich — Prof. of Computer Science; head of Laboratory of Computer Security Problems, SPIIRAS. Research interests: computer network security, including security policy management, access control, authentication, network security analysis, intrusion detection, firewalls, deception systems, malware protection, verification of security systems, digital right management, modeling, simulation and visualization technologies for counteraction to cyber terrorism, artificial intelligence, including multi-agent frameworks and systems, agent-based modeling and simulation, soft and evolutionary computing, machine learning, data mining, data and information fusion, telecommunications, including decision making and planning for telecommunication systems. The number of publications — 450. ivkote@comsec.spb.ru, www.comsec.spb.ru; SPIIRAS, 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone +7(812) 328-2642, fax +7(812)328-4450.

Нестерук Филипп Геннадьевич — старший научный сотрудник лаборатории проблем компьютерной безопасности СПИИРАН. Область научных интересов: адаптивные системы защиты информации, интеллектуальный анализ данных, нейронные сети, нечеткая логика, экспертные системы, генетические алгоритмы, искусственный интеллект, извлечение знаний, комплексные системы защиты информации. Число научных публикаций — более 100. 08p@mail.ru, www.comsec.spb.ru; СПИИРАН, 14-я линия В.О., д.39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328-2642, факс +7(812)328-4450.

Nesteruk Filipp Gennadyevich — PhD. of Computer Science, Senior researcher of Laboratory of Computer Security Problems, SPIIRAS. Research interests: adaptive systems of information security, data mining, neural networks, fuzzy logic, expert systems, genetic algorithms, artificial intelligence, knowledge extraction, complex systems of information security. The number of publications — over 100. 08p@mail.ru, www.comsec.spb.ru; SPIIRAS, 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone +7(812) 328-2642, fax +7(812)328-4450.

Поддержка исследований. Работа выполняется при финансовой поддержке РФФИ (проект 13-01-00843-а), программы фундаментальных исследований ОНИТ РАН (проект № 2.2), Министерства образования и науки Российской Федерации (государственный

контракт 11.519.11.4008), Комитета по науке и высшей школе Правительства Санкт-Петербурга, при частичной финансовой поддержке, осуществляемой в рамках проектов Евросоюза SecFuture и MASSIF, а также в рамках других проектов.

Рекомендовано СПИИРАН, лабораторией проблем компьютерной безопасности, заведующий лабораторией Котенко И.В., д-р техн. наук, проф., отделом проблем информационной безопасности, заведующий лабораторией Молдовян А.А., д-р техн. наук, проф., заведующий лабораторией криптологии Молдовян Н.А., д-р техн. наук, проф.

Статья поступила в редакцию 01.03.2013.

РЕФЕРАТ

Нестерук Ф.Г., Котенко И.В. **Инструментальные средства создания нейросетевых компонент интеллектуальных систем защиты информации.**

На рынке нейросетевых инструментальных средств создания интеллектуальных систем представлено большое количество различных программных средств, что объясняется сложностью и многоплановостью задач интеллектуальной обработки информации в различных сферах деятельности, в том числе для защиты информации в информационно-телекоммуникационных системах.

Актуальность проблемы выбора подходящих под решаемую задачу нейросетевых инструментальных средств в области защиты информации обусловлена большим разнообразием представляемых ими функций, возможностей настройки, характеристик и свойств, динамикой их развития, распределенной и разнородной структурой систем защиты и множеством других факторов.

В работе предлагается обзор инструментальных средств, применимых для создания нейросетевых компонент интеллектуальных систем защиты информации.

Отмечено несколько динамично развивающихся программных средств, содержащих наиболее широкий спектр возможностей, в достаточной мере отвечающих потребностям различных пользователей, в том числе, для целей моделирования и анализа нейросетевых компонент защиты информации в информационно-телекоммуникационных системах

SUMMARY

Nesteruk F.G., Kotenko I.V. **Tools for development of neural components of intelligent security systems.**

A large number of different software tools exists in the market of neural tools for development of intelligent systems. This fact can be explained by complexity and multilateral character of intelligent information processing in various fields of activity, including for the protection of information in information and telecommunication systems.

The relevance of selecting suitable neural tools in the field of information security is stipulated by a great variety of their functions, customization options, characteristics and properties, dynamics of their development, distributed and heterogeneous structure of security systems and many other factors.

The paper provides an overview of tools that are applicable for development of neural components of intelligent security systems.

There are several dynamically developing software tools, containing the widest range of possibilities, which sufficiently meet to the needs of different users, including to the aims of simulation and analysis of neural components for information security in information and telecommunication systems.