2. Chao, K., Y.R. Chen, and M. S. Kim. Machine vision technology for agricultural applications // Elsevier science transactions on computers and electronics in agriculture, 2002. – vol. 36. – P. 173-191.
3. N. Kumar, S. Pandey, A. Bhattacharya, and P. S. Ahuja, "Do leaf surface characteristics affect agrobacterium infection in tea [camellia sinensis (1.) o kuntze]?" J. Biosci., vol. 29, no. 3, pp. 309-317, 2004.
4. P. Soille. Morphological image analysis applied to crop field mapping // Image and Vision Computing.-2000.-vol. 18, no. 13.-P. 1025-1032.
5. Panagiotis Tzionas, Stelios E. Papadakis, Dimitris Manolakis. Plant leaves classification based on morphological features and a fuzzy surface selection technique. // 5th Int. Conf. on Technology and Automation ICTA'05, 15-16 October 2005 Thessaloniki, Greece- 2005. – P. 365-370.
6. Margarita Torre, Petia Radeva. Agricultural-Field Extraction on Aerial Images by Region Competition Algorithrn/flnt. Conf. on Pattern Recognition (ICPR'00), September 3-8, 2000, Barcelona, Spain-2000.-vol. 01, no. l. – P. 1313-1316.
7. A.V. Inyutin. The algorithm of image segmentation by gray-scale pseudo-skeleton // Proc. of the III Int. Conf. on Neural Networks and Artificial Intelligence (ICNNAI2003), November 12-14, Minsk. Belarus. - 2003. – P.263-265.
8. Apan, Armando and Kelly, Rob and Jensen, Troy and Butler, David and Strong, Wayne and Basnet, Badri. Spectral Discrimination And Separability Analysis Of Agricultural Crops And Soil Attributes Using Aster Imagery // In 1 lth Australasian Remote Sensing and Photogrammetry Conference, 2-6 September, Brisbane, Queensland.- 2002. – P. 396-411.
9. Burks, T.F., S.A. Shearer, and F.A. Payne. Classification of weed species using color texture features and discriminant analysis // Transactions of ASAE.- 2000.- vol. 43(2). – P. 441:448.
10. Analysis of colour images of infected crop field. Part 2. Two-stage algorithm of segmentation and improvement of color images of infected crop field / Alexander A. Doudkin et. // Wybrane zagadnienia ekologiczne we wspolczesnym rolnictwie - PIMR, Poznan, 2005. – P. 118-122.
11. Выделение областей зараженности сельскохозяйственных полей по цветовым характеристикам изображений / М.Е. Ваткин, А.А.Дудкин, А.В.Инютин и др. // 5-я международная гонференция "Обработка информации и управление в чрезвычайных и экстремальных ситуациях", Минск, Беларусь, 24-26 октября 2006 г. – Мн.: ОИПИ НАН Беларуси, 2006. - Т. 1. – С. 191-195.
12. Boleslaw Sobcowiak, Tadeusz Pawlowsky. Analiza obrazow kolorowych zaifekowanych pqluprawnych. Czesc 1. Preprow-adjizenie testu I ocean wynipow badan // Wybrane zagadnienia ekologiczne we wspolczesnym rolnictwie. - PIMR, Poznan, 2005. – P. 113-117.

Статья поступила в редакцию 28.01.2007

УДК 004.8.032.26

Безобразов С.В., Головко В.А.

НЕЙРОСЕТЕВОЙ ПОДХОД ДЛЯ ФОРМИРОВАНИЯ ДЕТЕКТОРОВ В ИСКУССТВЕННЫХ ИММУННЫХ СИСТЕМАХ ДЛЯ ЗАЩИТЫ ИНФОРМАЦИИ

Введение

Традиционный подход в обнаружении компьютерных вирусов, основанный на сигнатурном поиске, имеет существенный недостаток. Сигнатурный поиск не способен обнаруживать неизвестные вирусы, а существующие эвристические алгоритмы далеки от совершенства. Поэтому для успешной борьбы с вредоносными программами необходимо постоянно пополнять антивирусные базы, которые, как правило, располагаются на web-сайте разработчика антивирусного программного обеспечения (ПО). На отслеживание и скачивание новых антивирусных баз тратится какое-то, иногда продолжительное, время. Компьютер с устаревшими антивирусными базами может оказаться бессильным перед угрозой заражения новым вирусом [1].

В силу сложившейся ситуации те, кто разрабатывают вирусы (вирусописатели), постоянно идут на шаг впереди разработчиков антивирусного ПО. Сначала появляется новый вирус. Через некоторое время этот вирус различными путями попадает к разработчикам антивирусного ПО. Затем специалисты анализируют вирус и включают его сигнатуру в антивирусную базу, и только после этого пользователь может скачать обновленную антивирусную базу.

Сегодняшние исследования в области защиты информации направлены на создание такой антивирусной системы, которая позволяла бы обнаруживать неизвестные вирусы. Такая система повысила бы уровень защиты компьютерных систем и избавила бы пользователей от неудобных операций. Практически во всех существующих антивирусных ПО реализован эвристический анализатор. Эвристический алгоритм анализирует набор команд проверяемого файла. Если команды файла предусматривают деструктивные функции, угрожающие сохранности и целостности данным, то такой файл считается вирусом. Однако существующие эвристические алгоритмы далеки от совершенства. Зачастую анализатор «пропускает» действительно вирус или принимает за вирус чистый файл.

Позаимствованная у природы и построенная по основным принципам биологической иммунной системы искусственная иммунная система позволяет обнаруживать не только известные ей вирусы, но и неизвестные, как это делает иммунная система человека, ежедневно сталкиваясь с большим количеством чужеродных бактерий и вирусов в организме [2]. Основными элементами искусственной иммунной системы (ИИС), которые несут функцию по обнаружению вирусов, являются детекторы (антитела). На стадиях генерации и отбора детекторы приобретают структуру, схожую с чистыми файлами (позитивная селекция), или различную со структурой чистых файлов (негативная селекция), что в дальнейшем позволяет им различать вирусы от незараженных, чистых файлов [3].

В данной статье представлен разработанный нами метод формирования детекторов на основе LVQ-сетей (нейронные сети для векторного квантования), который позволяет уменьшить временные и вычислительные затраты, связанные с проверкой файлов на наличие вирусов. Также этот метод сочетает в себе преимущества обоих методов (негативная и позитивная селекции) отбора нежелательных детекторов.

В первом разделе статьи представлен механизм обнаружения вирусов при помощи ИИС. Второй раздел содержит описание метода формирования детекторов на основе LVQ-сетей. В третьем разделе содержится описание экспериментальной системы обнаружения компьютерных вирусов. В четвертом разделе рассмотрены результаты исследований.

***Безобразова Светлана Владимировна**, аспирант кафедры интеллектуальных информационных технологий БрГТУ.*
Беларусь, Брестский государственный технический университет, 224017, Беларусь, г. Брест, ул. Московская, 267.

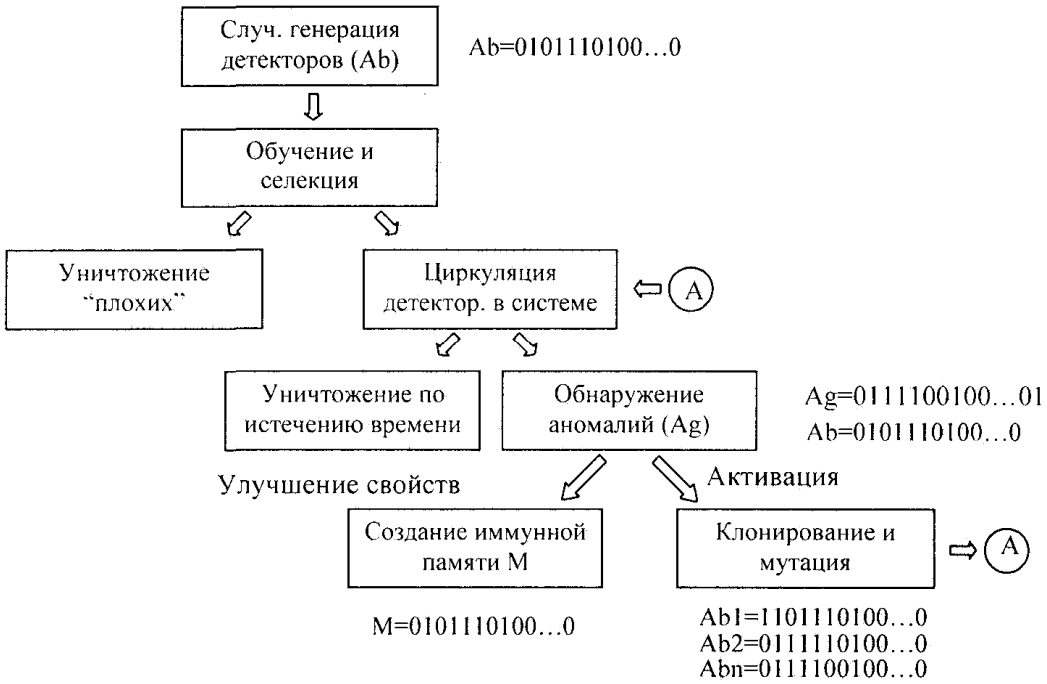

Рис. 1. *Модель искусственной иммунной системы*

1. Механизм обнаружения компьютерных вирусов

Биологический иммунитет основан на синтезе специальных белков, так называемых антител, способных вступать в соединение с чужеродными веществами – антигенами. Но для того, чтобы стать «зрелыми» и научиться обнаруживать антигены, антитела проходят стадии обучения и отбора, на которых отсеиваются те из них, которые реагируют на клетки собственного организма. Зрелые антитела имеют на своей поверхности детекторы, которые реагируют на специфический антиген [4].

Синтез (или генерация) детекторов в компьютерной системе представляет собой случайный процесс [5]. Случайным образом генерируется набор (популяция) детекторов. Каждый детектор представляет собой бинарную (двоичную) строку (рис. 1). После генерации детекторы проходят стадии обучения и селекции. Существует два основных метода отбора нежелательных детекторов: метод позитивной селекции и метод негативной селекции [6]. Метод позитивной селекции позволяет детекторам приобрести структуру, схожую со структурой чистых файлов. Если, на стадии проверки, обнаруживается файл, отличный от структуры детектора, то этот файл считается вирусом. Метод негативной селекции прямо противоположен методу позитивной селекции, т.е. детекторы приобретают структуру, отличную от чистых файлов. Соответственно, если происходит обнаружение совпадения в структурах файла и детектора, то файл считается вирусом.

Для обеспечения большого многообразия детекторов, каждому детектору выделяется определенное время, называемое жизненным циклом, на протяжении которого детектор находится в системе [5]. Детектор, который не обнаружил вирус в течение жизненного цикла, уничтожается, а на его место приходит новый детектор.

Мутация позволяет детекторам приобрести структуру, максимально схожую с обнаруженным вирусом, а благодаря механизму клонирования иммунная система создает большое количество однообразных детекторов, которые быстро справляются с заражением.

ИИС способна запоминать информацию о предыдущих заражениях компьютерной системы. Для этого существует иммунная память. Клетки памяти являются копиями детекторов, которые обнаруживали вирус [2]. Совокупность клеток памяти формируют иммунную (генетическую) память, в которой хранится информация обо всех вирусах, заражавших систему. Механизм иммунной памяти позволяет быстро реагировать на повторные заражения системы.

Таким образом, мы вкратце рассмотрели механизм обнаружения вирусов в ИИС. Такая система действительно способна обнаруживать неизвестные вирусы и не требует постоянного обновления как в случае с сигнатурным поиском. Как уже отмечалось ранее, детектор представляет собой бинарную строку определенной размерности. Однако такая структура детектора накладывает определенные ограничения. Как известно, сравнение является одной из самых медленных процессорных операций. В результате требуется достаточно много временных и вычислительных затрат на стадии проверки файлов, что не приемлемо для системы защиты информации. Для решения этой проблемы нами было предложено применение нейронных сетей, а именно LVQ-сетей, для формирования детекторов.

2. LVQ-сети для формирования детекторов

Нейронная сеть для векторного квантования была предложена в 1982 году Кохоненом и называется обучающим векторным квантователем (learning vector quantization – LVQ) [7]. LVQ-сеть представляет собой двухслойную нейронную сеть (конкурирующий и линейный слои) с прямым распространением сигналов (рис. 2).

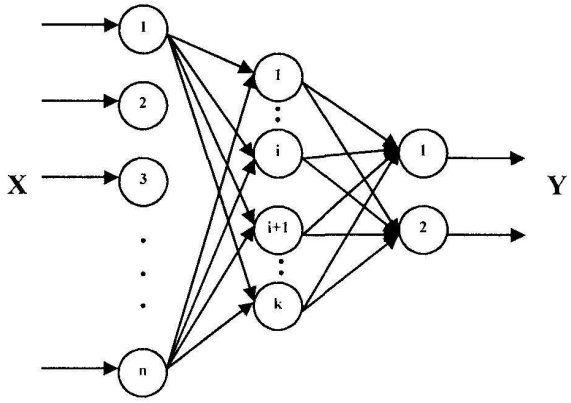

Рис. 2. *Нейронная сеть для векторного квантования*

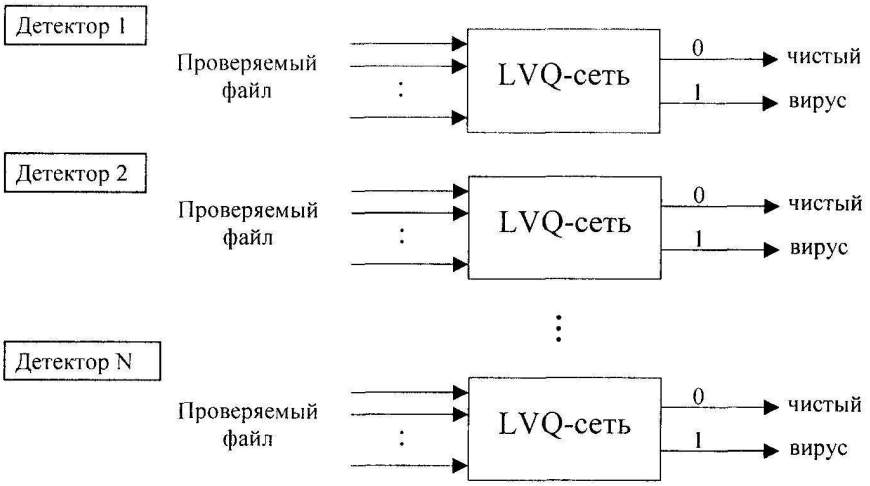

Рис. 3. *Модель работы детекторов на основе LVQ-сетей*

Оба слоя нейронной сети содержат по одному конкурирующему на каждый кластер и одному линейному нейрону на каждый целевой класс. Конкурирующий слой выполняет кластеризацию векторов, а линейный слой соотносит кластеры с целевыми классами, заданными пользователем [8]. Векторный квантователь обучается в процессе поступления эталонных векторов. В процессе обучения образуются кластеры различных эталонов, каждому из которых соответствует свой нейрон. При поступлении на вход такой нейронной сети неизвестного образа, он идентифицируется в соответствии с мерой близости к эталонным векторам и кодируется на выходе сети номером нейрона. Совокупность кодовых векторов называется кодовой книгой. При поступлении входного вектора на сеть происходит его сравнение с вектором из кодовой книги. В процессе этого выбирается такой кодовый вектор, который наилучшим образом аппроксимирует входной вектор и его номер используется в качестве кода. В качестве меры близости может использоваться евклидово расстояние [9].

Рассмотрим процесс формирования детекторов на основе LVQ-сети. Первоначально определяется набор чистых файлов: это могут быть утилиты операционной системы, различные документы, файлы разнообразного программного обеспечения. Из этих файлов случайным образом выбираются участки определенной длины (к примеру, бинарные строки размерностью 128 бит) и подаются на вход векторного квантователя. Для обучения нейронной сети также необходимо наличие какого-нибудь вируса (или его сигнатуры), из которого подобным образом выбирается битовая строка и подается на вход LVQ-сети. Таким образом, указывая нейронной сети явные структурные различия чистых файлов и вирусов (а компьютерные вирусы структурно отличаются от чистых файлов, так как подразумевают деструктивные действия), мы обучаем ее обнаруживать аномалии, т.е. компьютерные вирусы. В процессе проверки файлов LVQ-сеть идентифицирует неизвестный образ и определяет его близость к тому или иному эталонному вектору. Наличие разнообразных чистых файлов для обучения и элемента случайности в формировании входных векторов дает возможность получить большое количество различных по своей структуре детекторов.

Совокупность таких нейронных сетей образует популяцию детекторов, которые выполняют функцию по обнаружению вирусов.

3. Описание экспериментальной системы

Нами была построена модель ИИС, формирование детекторов в которой происходило на основе LVQ-сети (рис. 3).

Для обучения нейронной сети выбирались несколько, заведомо известно, чистых файлов (утилиты операционной системы Microsoft Windows XP) и один вирус. Из каждого файла случайным образом выбирались пять фрагментов – бинарные строки размерностью 128 бит, и подавались на вход нейронной сети. Нейронная сеть состоит из 128 элементов входного слоя (входной слой выполняет распределительную функцию), 10 элементов скрытого слоя (скрытый слой выполняет кластеризацию векторов), 2 элементов выходного слоя (выходной слой соотносит кластеры с целевыми классами). Для настройки сети использовалось правило конкурентного обучения с одним победителем, т.е. все множество образов разбивалось на кластеры, каждому из которых соответствовал свой нейронный элемент [8].

При поступлении на вход LVQ-сети неизвестного образа, она соотносит его к такому кластеру, на который он больше всего похож. Так как анализируемый файл разбивается на фрагменты по 128 бит, детектор высчитывает вероятности выходных значений, и, исходя из полученных результатов, классифицирует тестируемый файл. В проводимых экспериментах LVQ-сеть обучалась на четырех чистых файлах и одном вирусе. Соответственно выходные вероятности для идентификации проверяемого файла как «чистого» либо как вируса составили $P_ч = 4/5 = 0{,}8$ и $P_в = 1/5 = 0{,}2$ соответственно. Т.е. если $P_ч > 0{,}8$, то проверяемый файл является «чистым», а если $P_ч < 0{,}8$, то проверяемый файл считается зараженным. Если же выходные вероятности соответствовали $P_ч = 0{,}8$ и $P_в = 0{,}2$, то с уверенностью нельзя определить, к какому из классов (чистый или вирус) относится данный файл. Такой файл является «подозрительным» и требует дополнительных проверок.

4. Результаты исследований

Модель ИИС тестировалась по следующей схеме: детекторы, которые прошли стадию отбора, проверялись сначала на невосприимчивость к чистым файлам. Затем они же проверялись на способность обнаруживать различные вирусы и семейства вирусов. Таблица 1. и рисунок 4. демонстрируют результаты проверок чистых файлов тремя различными детекторами.

Так как нежелательные детекторы уничтожаются на стадии отбора, то не происходит ложного срабатывания, т.е. детекторы не реагируют на чистые файлы, что подтверждают полученные результаты.

Таблица 1. *Результаты проверки чистых файлов*

Имя файла	Детектор 1	Детектор 2	Детектор 3	Среднее значение
ctfmon.exe	0,83	0,84	0,81	0,83
dcomcnfg.exe	0,9	0,89	0,87	0,89
diskcopy.com	0,88	0,83	0,86	0,86
hh.exe	0,87	0,84	0,85	0,85
notepad.exe	0,85	0,83	0,86	0,85
regedit.exe	0,84	0,81	0,85	0,83
template.exe	0,85	0,83	0,85	0,84
cacls.exe	0,84	0,82	0,87	0,84
dbexplor.exe	0,91	0,89	0,9	0,90
dllhost.exe	0,87	0,87	0,85	0,86
etm70.exe	0,91	0,87	0,9	0,89

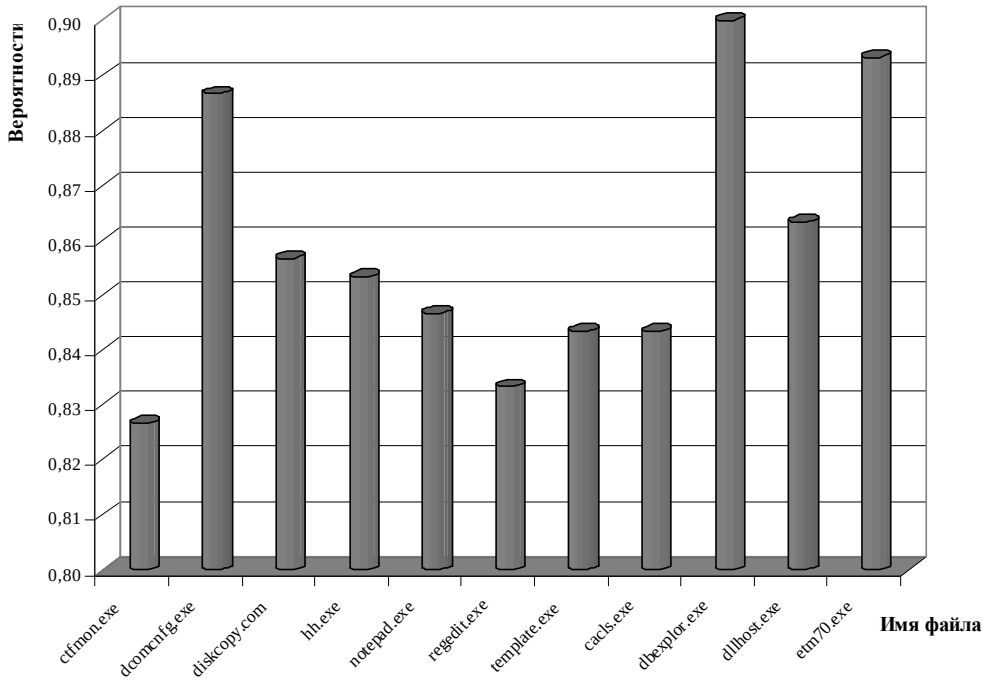

Рис. 4. *Усредненные результаты проверки чистых файлов*

Таблица 2. *Результаты проверки вирусов*

Имя вируса	Детектор 1	Детектор 2	Детектор 3
Maslan	0,83	0,77	0,83
DebPloit	0,94	0,92	0,73
Lovesan	0,74	0,71	0,72
Hidrag	0,75	0,77	0,75
LazyMin	0,75	0,76	0,75
Sober	0,68	0,71	0,69
Trojan.VB	0,8	0,79	0,81
Bagle	0,79	0,68	0,7
Win95.cih	0,8	0,82	0,78
Bagle.bn	0,69	0,67	0,68
Bagle.bj	0,96	0,93	0,79

Результаты проверок компьютерных вирусов представлены в таблице 2 и на рисунке 5. Из результатов исследований видно, что детекторы способны распознавать вирусы (значения вероятностей $P_в > 0{,}2$).

Для следующего теста были выбраны два известных антивирусных продукта: Антивирус Касперского и разработка компании Eset антивирус NOD32. Набор файлов, состоящий из вирусов и инфицированных файлов, проверялся антивирусом Касперского с актуальными вирусными базами, с устаревшими базами, антивирусом NOD32 с отключенными антивирусными базами (был задействован только эвристический анализатор) и разработанной нами искусственной иммунной системой. Результаты тестирования представлены в таблице 3.

Антивирус с актуальными вирусными базами обнаружил все вирусы, так как их сигнатуры оказались в базе.

Антивирус с устаревшими базами обнаружил только шестнадцать вирусов из двадцати трех присутствующих. Это объясняется тем, что «старые» вирусы, уже находившиеся в базе, были обнаружены, а вот сигнатур новых вирусов в ней не оказалось. Следует отметить, что антивирус с устаревшими базами не смог обнаружить все модификации вируса Worm.Win32.Bagle, хотя одна версия этого вируса от другой отличаются незначительно.

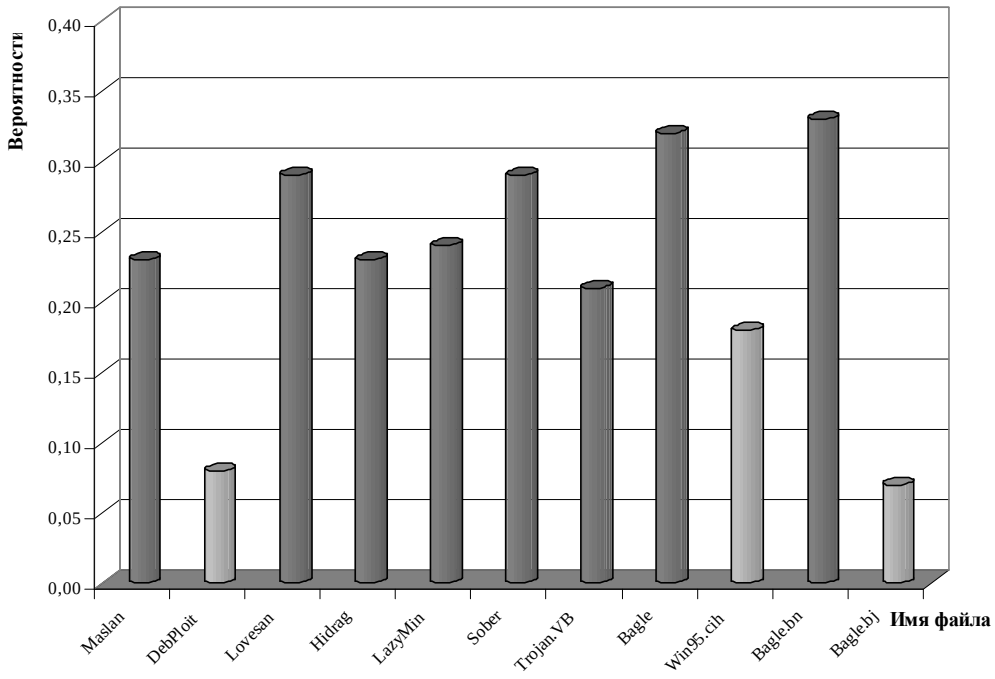

Рис. 5. *Результаты проверки вирусов вторым детектором*

Таблица 3. *Сравнительный анализ обнаружения вирусов*

Имя проверяемого файла	Антивирус Касперского с актуальными базами	Антивирус Касперского с устаревшими базами	Антивирус NOD 32	ИИС
Ex_DebPloit.vir	Exploit.Win32.DebPloit	Exploit.Win32.DebPloit	OK	OK
N_Lovesan.vir	Worm.Win32.Lovesan.a	Worm.Win32.Lovesan.a	OK	Вирус
Trojan_LdPinch.akv	Trojan.Win32.LdPinch	-	Win32.Delf.AJD	Вирус
V_Hidrag.vir	Virus.Win32.Hidrag.a	Virus.Win32.Hidrag.a	Win32/Jeefo	Вирус
W_BadTrans_a.vir	Worm.Win32.BadTrans.a	Worm.Win32.BadTrans.a	Вероят.неизв.вирус	Вирус
W_BadTransII.vir	Worm.Win32.BadTransII	Worm.Win32.BadTransII	Вероят.неизв.вирус	Вирус
W_Bagle_ai.vir	Worm.Win32.Bagle.ai	-	Вероят.неизв.вирус	Вирус
W_Bagle_be.vir	Worm.Win32.Bagle.be	-	Вероят.неизв.вирус	Вирус
W_Bagle_bj.vir	Worm.Win32.Bagle.bj	-	Вероят.неизв.вирус	OK
W_Bagle_bn.vir	Worm.Win32.Bagle.bn	-	Вероят.неизв.вирус	Вирус
W_Bagle_fn.vir	Worm.Win32.Bagle.fn	-	Вероят.неизв.вирус	Вирус
W_Bagle_i.vir	Worm.Win32.Bagle.i	Worm.Win32.Bagle.i	Вероят.неизв.вирус	Вирус
W_Hybris_b.vir	Worm.Win32.Hybris.b	Worm.Win32.Hybris.b	Вероят.неизв.вирус	Вирус
W_Klez_h.vir	Worm.Win32.Klez.h	Worm.Win32.Klez.h	OK	Вирус
W_MTX.vir	Worm.Win32.MTX	Worm.Win32.MTX	Вероят.неизв.вирус	Вирус
W_Mydoom_m.vir	Worm.Win32.Mydoom.m	-	Вероят.неизв.вирус	Вирус
W_NetSky_d.vir	Worm.Win32.NetSky.d	Worm.Win32.NetSky.d	OK	Вирус
W_NetSky_q.vir	Worm.Win32.NetSky.q	Worm.Win32.NetSky.q	OK	Вирус
W_Roron55f.vir	Worm.Win32.Roron.55.f	Worm.Win32.Roron.55.f	Вероят.неизв.вирус	Вирус
W_Sober_f.vir	Worm.Win32.Sober.f	Worm.Win32.Sober.f	Вероят.неизв.вирус	Вирус
W_Sobig_c.vir	Worm.Win32.Sobig.c	Worm.Win32.Sobig.c	Вероят.неизв.вирус	Вирус
W_Sobig_f.vir	Worm.Win32.Sobig.f	Worm.Win32.Sobig.f	Вероят.неизв.вирус	Вирус
W_Tanatos_a.vir	Worm.Win32.Tanatos.a	Worm.Win32.Tanatos.a	Вероят.неизв.вирус	Вирус

Антивирус NOD32, использующий эвристический анализатор для обнаружения вирусов, хоть и показал неплохой результат, также не обнаружил все вирусы, что говорит о несовершенстве существующих эвристических алгоритмов.

С помощью ИИС были обнаружены практически все вирусы и инфицированные файлы, за исключением Ex_DebPloit.vir и W_Bagle_bj.vir. В первом случае это можно объяснить тем, что при обучении детектора использовались сигнатуры вируса-червя, который, по своим функциям, кардинально отличается от exploit-вируса. Т.е. для обнаружения таких вирусов необходимо включать их сигнатуры в процессе обучения детектора. В случае с файлом W_Bagle_bj.vir мы сталкиваемся с явной ошибкой ИИС, т.е. файл-вирус был классифицирован как чистый файл.

Выводы

1. Разработан метод формирования детекторов искусственной иммунной системы для защиты информации на основе LVQ-сети. Данный метод позволяет значительно уменьшить затраты на вычислительные ресурсы компьютерной системы, следовательно сокращается время проверки файлов на наличие вирусов. Благодаря свойствам LVQ-сети уменьшается время, необходимое для обучения детектора. Размер детекторов увеличивается, так как каждый детектор представляет собой нейронную сеть, однако это компенсируется повышением уровня обнаружения,

т.е. один детектор способен обнаружить большее количество вирусов.
2. Один (или однотипные) детектор не в состоянии обнаружить все разновидности вирусов, так как невозможно приобрести структуру детектора, которая была бы схожа с большим многообразием вредоносных программ.
3. ИИС способна распознавать чистые файлы и вредоносные программы и обнаруживать неизвестные вирусы. Вероятность возникновения ошибки очень мала, что делает такую систему привлекательной для использования в системах защиты информации. Использование ИИС совместно с уже существующими антивирусными продуктами значительно увеличит уровень защиты компьютерной системы

Исследования проводятся в рамках научно-исследовательского проекта по теме «Методы искусственного интеллекта для защиты информации» по заказу Министерства образования Республики Беларусь.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Почему не срабатывают антивирусы – http://www.i2r.ru, 2003.
2. L de Castro and J Timmis. Artificial Immune Systems: A New Computational Intelligence Approach. Springer, 2002.
3. С.В. Безобразов. Искусственные иммунные системы для защиты информации: сравнительный анализ методов негативной и позитивной селекций детекторов // Инженерный вестник.-2006.- №1(21)/1.-С.76-82.
4. Иммунитет. Энциклопедия «Кругосвет» – http://krugosvet.ru, 2004.
5. С.В. Безобразов. Применение искусственных иммунных систем для обнаружения вирусов // Вестник БрГТУ. Физика, математика, информатика.-2005.- №5(35).-С.66-70.
6. F. Esponda, S. Forrest, and P. Helman. A formal framework for positive and negative detection. IEEE Transactions on Systems, Man, and Cybernetics 34:1 pp. 357-373, 2004.
7. Kohonen T. Sell-organised formation of topologically correct feature maps// Biological Cybernetics. - 1982. - N43.-P.59-69.
8. В.Медведев, В.Потемкин. Нейронные сети. MATLAB 6. М: Диалог-МИФИ. 2002. 496 с.
9. В.А. Головко. Нейронные сети: обучение, организация и применение. Кн. 10: Учеб. пособие для вузов / Общая ред. А. И. Галушкина. - М.: ИПРЖР, 2000. –С.114-129.

Статья поступила в редакцию 21.12.2006

УДК 681.3

Горбашко Л.А.

АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ СТЕГАНОГРАФИЧЕСКИХ СИСТЕМ С ВЕКТОРНЫМ КВАНТОВАНИЕМ

Введение

Задача защиты информации от несанкционированного доступа приобретает все большую актуальность в современном мире. Развитие информационных технологий дало новый толчок для развития компьютерной стеганографии. Стеганография исследует скрытую передачу данных в маскирующем сигнале.

Для скрытой передачи информации выбирается контейнер, которым чаще всего служит графический файл. В него особым образом встраивается сообщение, которое нужно передать тайно. В результате получается комбинированное изображение – стего, которое и передается по каналу связи.

Современные исследования направлены на встраивание информации в частотной области. При этом перед встраиванием данных производят разложение контейнера на любые частотные составляющие (преобразования Фурье, вейвлет-, дискретное косинусное и т.п.).

При встраивании текстового сообщения необходимо обеспечить его извлечение без искажений, для чего применяются дублирование сообщения, избыточное кодирование. Соответственно, объем встраиваемых данных в этом случае невелик и составляет примерно 1% от размера контейнера [1]. При встраивании цифровых данных, имеющих аналоговую природу - изображения, звуковые файлы - можно допустить некоторое искажение оригинала при встраивании, т.к. человеческие органы чувств, воспринимающие информацию, не являются идеальными. Небольшие отклонения цветов изображения либо амплитуды и частоты звука не определяются человеком. Поэтому появляется возможность значительно увеличить объем встраиваемых данных, используя предварительное сжатие сообщения. Методы сжатия могут быть различны от отбрасывания нулевых коэффициентов после частотного разложения сообщения до предварительного квантования и кодирования сообщения.

Таким образом, современной проблемой стеганографии является увеличение объема встроенных данных при сохранении прозрачности и робастности стего. Для этого применяются различные методы, в частности, сжатие данных перед встраиванием. Одним из способом сжатия является применение различных методов квантования.

Применение квантования в стеганографии

Квантование является одним из способов сокращения объема информации. В стеганографии широко используются все виды квантования. Так, линейное и решетчатое квантование используются авторами в работах [1, 2].

Линейное квантование заключается в замене каждого отсчета сигнала на число, кратное шагу квантования. В результате уменьшается множество возможных значений сигнала. При встраивании сообщения в квантованный контейнер по аддитивному алгоритму отсчеты сигнала изменяются на произвольную величину, равную значениям отсчетов сообщения.

При линейном квантовании сообщения его можно закодировать при помощи алгоритма Кодирование длин серий (RLE – Run Length Encoding).Сжатие сообщения в данном алгоритме происходит за счет того, что в квантованном изображении встречаются последовательности одинаковых чисел. Тогда их можно заменить на пары <количество повторений, значение> [3]. После такого предварительного сжатия сообщения получим увеличение объема внедренных данных при встраивании.

Если предварительно использовать линейное квантование контейнера, то ограничение набора возможных значений отсчетов контейнера дает возможность извлечь сообщение без знания контейнера. Достаточно проквантовать стего с тем же шагом и вычислить разность между принятым стего и квантованным стего- это и будет встроенное сообщение. Реализация

***Горбашко Лариса Ашотовна**, ст. преподаватель кафедры интеллектуальных информационных технологий БрГТУ, lagorbashko@bstu.by.*
Беларусь, Брестский государственный технический университет, 224017, Беларусь, г. Брест, ул. Московская, 267.