Следующей планируемой модификацией алгоритма идентификации будет поиск наборов зависимых признаков не для всех возможных признаков, а только для тех, которые встречаются в исследуемых информационных объектах, что уменьшит количество возможных наборов и, соответственно, снизит временные затраты. Все это в итоге позволит обеспечить заданный уровень информационной безопасности.

Выводы

Использование предложенного алгоритма позволяет с заданным уровнем доверия решить задачу идентификации объекта в случае большого количества признаков с относительно небольшими вычислительными затратами.

Модификация алгоритма идентификации позволяет снизить уровень ошибок и повысить качество принимаемых решений, в том числе и для обеспечения информационной безопасности.

СПИСОК ЛИТЕРАТУРЫ:

1. *Кулик С. Д., Никонец Д. А.* Примеры использования нейросетевого алгоритма в методиках для эксперта-почерковеда // Нейрокомпьютеры: разработка и применение. 2009. № 9. С. 61–85.
2. *Кулик С. Д., Никонец Д. А., Ткаченко К. И., Жижилев А. В.* Устройство определения поддельных документов // Безопасность информационных технологий. 2009. № 1. С. 114–115.
3. *Кулик С. Д., Никонец Д. А., Ткаченко К. И., Жижилев А. В.* Патент на полезную модель № 73750, Российская Федерация (RU), л. МПК7 G 07 D 7/00. Устройство определения фальшивых рукописных документов на русском языке / С. Д. Кулик, Д. А. Никонец, К. И. Ткаченко, А. В. Жижилев (Россия). Заявка № 2007147832/22; Заяв. 25.12.2007; Зарегистр. 27.05.2008; Приоритет от 25.12.2007. Опубл. Бюл. № 15. Ч. 3. С. 860. (РОСПАТЕНТ).
4. *Webb G. I.* Discovering Significant Patterns // Machine Learning. Vol. 68 (1). Netherlands: Springer, 2007. P. 1–33.

С. Д. Кулик, К. И. Ткаченко

РАЗРАБОТКА ГЕНЕРАТОРОВ ДЛЯ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

На практике при обучении экспертов-криминалистов часто возникает проблема недостатка образцов документов с искаженной информацией из-за малой наполненности учебного фонда необходимыми документами. В рамках решения данной проблемы было предложено на основе имеющихся образцов и правил составления подлинных документов различных типов генерировать образцы искаженных (фальшивых) документов [1–8]. Были разработаны и внедрены генераторы искажения информации для фальшивых векселей и фальшивых инструкций к лекарственным средствам [2–5, 8]. Обобщенная схема функционирования данных генераторов искажения информации [3–6] представлена на рис. 1.

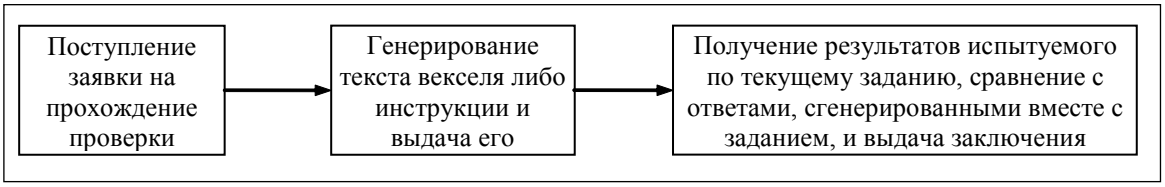

Рис. 1. Обобщенная блок-схема функционирования разработанных генераторов

В криминалистике до сих пор не решена проблема эффективной проверки профессиональных навыков эксперта-почерковеда, способного эффективно выявлять искаженную информацию. На практике проблема усложняется еще и очень большим количеством признаков рукописных букв, распознаванию которых необходимо обучить эксперта (например, для многих букв алфавита существует порядка ста признаков). Проблема усугубляется малым количеством доступных для контроля рукописных текстов, так как данные тексты вначале должны быть обработаны высококвалифицированными экспертами с целью выявления всех признаков, которые должны быть обнаружены испытуемым. Для решения рассматриваемой проблемы предлагается использовать базу данных, как в [7], и генератор рукописных текстов, как в [3, 4], содержащих искажения. Блок-схема функционирования генератора представлена на рис. 2.

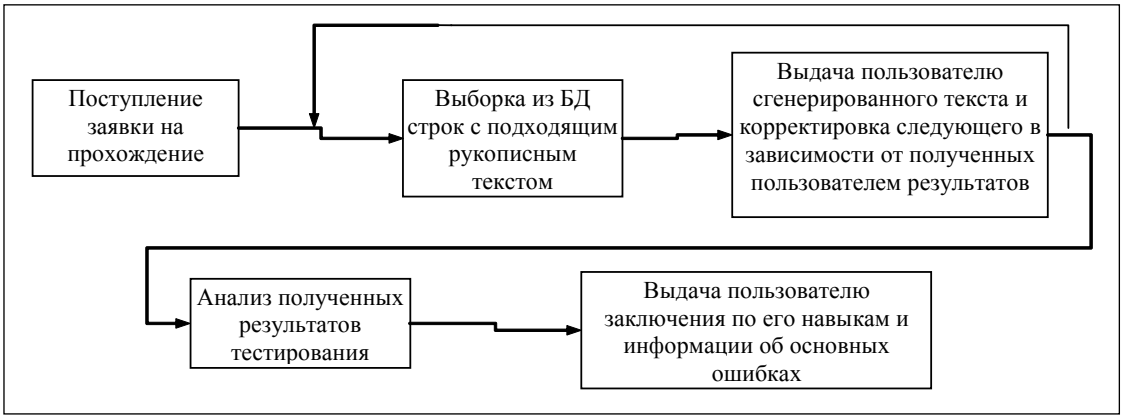

Рис. 2. Блок-схема функционирования генератора рукописных текстов

На первом этапе заполняют базу данных (БД) изображениями образцов рукописных текстов, разбитых на отдельные строки с указанием количества и типов признаков в каждой строке. Далее при необходимости проверки испытуемому генерируется изображение, составленное из различных строк рукописного текста. Затем испытуемый выявляет признаки букв, и его результаты сравниваются с эталонным значением, основанным на данных из БД. После сравнения на основе сделанных испытуемым ошибок и общего числа признаков различных типов в сгенерированном тексте генерируется следующий текст. Так происходит до тех пор, пока не будет принято решение о навыках работы испытуемого или не прекращено само испытание.

Генераторы в совокупности с процессом обучения позволяют повысить эффективность работы эксперта за счет его более полного обучения и проверки навыков. Основными направлениями дальнейшей работы по повышению эффективности деятельности эксперта-криминалиста являются развитие существующих генераторов, разработка новых генераторов текстовых и графических данных, а также создание автоматизированной информационной системы, обобщающей разработанные генераторы для обеспечения информационной безопасности.

СПИСОК ЛИТЕРАТУРЫ:

1. *Кулик С. Д., Ткаченко К. И., Никонец Д. А.* Инструментальные средства выявления искажений информации в документах // Безопасность информационных технологий. 2009. № 3. С. 29–36.

2. *Кулик С. Д., Никонец Д. А., Ткаченко К. И., Жижилев А. В.* Патент на полезную модель № 73750, Российская Федерация (RU), кл. МПК7 G 07 D 7/00. Устройство определения фальшивых рукописных документов на русском языке / С. Д. Кулик, Д. А. Никонец, К. И. Ткаченко, А. В. Жижилев (Россия). Заявка № 2007147832/22; Заяв. 25.12.2007; Зарегистр. 27.05.2008; Приоритет от 25.12.2007. Опубл. Бюл. № 15. Ч. 3. С. 860. (РОСПАТЕНТ).

3. *Кулик С. Д., Ткаченко К. И.* Генератор изменений текста // Научная сессия МИФИ-2008. Сборник научных трудов в 15 т. М.: МИФИ, 2008. Т. 13. С. 83–84.
4. *Кулик С. Д., Ткаченко К. И.* Генератор графических данных // Научная сессия МИФИ-2008. Сборник научных трудов в 15 т. М.: МИФИ, 2008. Т. 13. С. 85–86.
5. *Кулик С. Д., Ткаченко К. И.* Генератор статистических данных // Научная сессия МИФИ-2008. Сборник научных трудов в 15 т. М.: МИФИ, 2008. Т. 13. С. 87–88.
6. *Кулик С. Д., Ткаченко К. И.* Инструментальные средства разработки генератора для принятия решений // Актуальные проблемы управления-2007: Материалы 12-й Международной научно-практической конференции: Вып. 4. М.: ГУУ, 2007. С. 48–51.
7. *Кулик С. Д., Ткаченко К. И.* Свидетельство на базу данных Российской Федерации № 2007620326 «База данных задач v.1.0» (DBP) / С. Д. Кулик, К. И. Ткаченко (Россия). Заявка № 2007620227; Заяв. 24.07.2007; Зарегистр. 21.10.2007; Опубл. Бюл. № 4 (61). Ч. 2. С. 338. (РОСПАТЕНТ).
8. *Кулик С. Д., Ткаченко К. И.* Свидетельство на программу Российской Федерации № 2007614032 «Генератор учебных задач v.1.0» (GTP) / С. Д. Кулик, К. И. Ткаченко (Россия). Заявка № 2007613042; Заяв. 24.07.2007; Зарегистр. 21.10.2007; Опубл. Бюл. № 4 (61). Ч. 2. С. 282. (РОСПАТЕНТ).

И. О. Леошкевич

ПОИСК УТЕЧЕК ИНФОРМАЦИИ В ПРОГРАММНОМ ОБЕСПЕЧЕНИИ С ПОМОЩЬЮ АНАЛИЗА ИСПОЛНЯЕМОГО КОДА

Утечки информации могут привести к компрометации системы, использующей математически надежные на данном временном промежутке криптографические алгоритмы. Рассматриваемой в данной работе разновидностью утечек является сохранение программной реализацией секретного ключа, исходного текста или производной от них информации, отличной от предусмотренной используемым алгоритмом. Для ее получения злоумышленнику необходимо иметь определенные привилегии или физический доступ к компьютеру, однако атаку можно осуществить после окончания сеанса работы пользователя. Для этого можно использовать снимки физической памяти и содержимое жесткого диска. Остаточные данные могут сохраниться в физической памяти, если реализация не предполагает необходимых мер по ее удалению. Отметим, что снимок физической памяти можно сделать даже после выключения компьютера с помощью атаки «холодного старта». На диске остаточные данные могут сохраниться при использовании временных файлов, а также в результате подкачки.

Рассматриваемая в данной работе задача формулируется следующим образом: для заданного фрагмента программы и списка участков памяти, изначально содержащих конфиденциальную информацию, требуется определить, какие участки памяти или внешние устройства после завершения работы этого фрагмента могут получить информацию, производную от конфиденциальной. Анализ исходного кода системы, являющийся очевидным методом решения, имеет следующие недостатки. Во-первых, компилятор может удалить операции языка высокого уровня, стирающие конфиденциальную информацию, как мертвый код. Во-вторых, для улучшения производительности зачастую используются фрагменты на ассемблере. В-третьих, не всегда можно доверять компилятору: его разработчики могут быть злоумышленниками или допустить ошибку, кроме того, компилятор может быть модифицирован вредоносным ПО. В этих условиях видится актуальной разработка инструментального средства для анализа зависимостей в исполняемом коде.

Анализ программного обеспечения делится на два этапа: получение модели программы, отражающей интересующие нас свойства, и последующий анализ этой модели. Получение модели программы по ее исполняемому коду связано с рядом специфических трудностей. В