В.Ю. Каганов, А.К. Королёв, М.Н. Крылов,
И.В. Машечкин, М.И. Петровский

МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ В ЗАДАЧАХ АУТЕНТИФИКАЦИИ С ИСПОЛЬЗОВАНИЕМ ИНФОРМАЦИИ О ДИНАМИКЕ НАБОРА ПАРОЛЯ НА КЛАВИАТУРЕ *

1 Актуальность

В большинстве современных информационных систем одной из важнейших задач, помимо сохранения и обработки данных, является задача разграничения доступа к ресурсам. Это используется как для предотвращения несанкционированного доступа к системам извне, так и для разграничения прав сотрудников, работающих с информационной системой внутри организации. Поэтому задача аутентификации, то есть проверки подлинности пользователя, желающего получить доступ к системе, является одной из ключевых.

В настоящее время эта задача может быть решена множеством способов. Самый популярный способ из тех, что используются в современных информационных системах – это аутентификация по паролю — специальной последовательности символов, не известной никому, кроме пользователя, которому разрешен доступ к системе. Помимо очевидных преимуществ, таких как простота реализации и использования, а также распространенность, у нее есть и существенные недостатки: сама кодовая фраза может быть забыта, передана другим лицам, а при недостаточной длине или сложности — подобрана простым перебором или перебором по словарю, что ставит под сомнение возможность их использования в системах, требующих высшего уровня безопасности.

От последнего недостатка свободны системы электронно-цифровых подписей (ЭЦП), также применяемых для аутентификации [1]. Однако проблему безопасного хранения закрытых ключей таким образом решить невозможно, поскольку они хранятся с использованием других средств контроля доступа.

*Работы выполнены при финансовой поддержке Минобрнауки России (Номер соглашения о предоставлении субсидии: 14.604.21.0056).

В связи с вышесказанным, значительная часть систем, обеспечивающих эффективную безопасность, использует биометрию для определения личности пользователя. Биометрические комплексы могут быть разделены на две категории. К первой отнесем системы, которые используют различные в силу естественных причин особенности человека, такие как отпечатки пальцев, сетчатка глаза, голос, тепловая карта тела и т. п. Они эффективны, но довольно дороги, т. к. требуют установки специального оборудования. Другую категорию составляют системы, которые анализируют поведение пользователя и основаны на опыте или особых навыках. Они не требуют какого-либо специального оборудования и просты для внедрения. Один из таких подходов – анализ динамики нажатий клавиш.

В этой статье будут рассмотрены подходы к аутентификации пользователя по различным моделям, построенным на информации о нажатиях клавиш при наборе пароля, предложена модель представления данных о пользователе, а так же проведен анализ эффективности алгоритмов интеллектуального анализа данных, работающих с этой моделью.

2 Постановка задачи

В рамках этой статьи рассматривается задача статической аутентификации.

Метод работает на известном шаблоне, слове или другом заранее предопределенном тексте. Набираемые пользователем при входе данные собираются (например, при вводе пароля) и сравниваются с предшествующими удачными попытками входа. Данный подход рассматривается как расширение стандартного метода аутентификации при входе с использованием логина/пароля (т. е. при входе в систему проверяется не только *что* набрал пользователь, но и *как* он это сделал). Стоит отметить некоторые особенности статической аутентификации. Во-первых, это небольшое количество входных данных. Как правило, статическая аутентификация работает в паре с аутентификацией по паролю, а использование чрезвычайно длинных паролей, которые пользователь набирал бы вручную (более 100 символов), практически исключено.

Другой особенностью является статичность данных. Один и тот же пароль, как правило, используется для входа в систему множество раз, и из набора этого пароля можно извлечь лишь небольшое количество признаков. Кроме того, чаще всего пароль меняется очень редко, что позволяет накопить большой объем выборки. Таким образом, метод должен быть оптимизирован для распознавания пользователя по небольшому множеству параметров.

И, наконец, для статической аутентификации важна высокая скорость работы, потому что возможности проводить обработку данных для аутен-

тификации параллельно с работой пользователя нет: до тех пор пока аутентификация не завершится успешно, пользователь не будет допущен к работе с системой. Поэтому необходимо сделать задержку между вводом пароля и входом в систему как можно меньшей.

Более формально задачу аутентификации можно описать так: пусть задано некоторое множество *пользователей*, которые совершают определенные *действия*. Такими действиями могут считаться, например, нажатие одной клавиши или набор пароля. Задача обучения в этом случае заключается в том, чтобы каждому пользователю сопоставить некоторую функцию (модель), которая может служить мерой аномальности действия для пользователя. В таком случае задача аутентификации – это вычисление аномальности нового действия пользователя по его модели и обработка полученного значения, по которому принимается решение об успешности аутентификации.

Для оценки результатов в рамках данного исследования (и в аналогичных исследованиях других авторов) использовалось вычисление величин False Rejection Rate и False Acceptance Rate [2]. False Rejection Rate (FRR) — процент попыток ввода пароля пользователя, на котором система обучена, воспринятые системой как нажатия другого пользователя (ошибка 1 рода). False Acceptance Rate (FAR) — процент попыток ввода пароля другого пользователя, которые система определила как нажатия пользователя, на котором обучена (ошибка 2 рода).

В силу того, что алгоритмы, в том числе и предлагаемый в данной статье, в качестве результата возвращают не бинарное значение (аутентификация прошла успешно / аутентификация прошла неуспешно), а некоторое вещественное значение, показывающее, насколько попытка соответствовала данным, предоставленным в момент обучения, имеет смысл ввести некоторое пороговое значение, разграничивающее успешные и неуспешные попытки аутентификации. В дальнейшем, варьируя это значение, можно будет найти порог, при котором FRR будет равен FAR. Значение ошибки при использовании этого порога называется Equal Error Rate (EER) и является одним из наиболее важных показателей при оценке качества алгоритма [2].

Так как в данных, используемых для эксперимента, содержится информация о наборе пароля большим числом человек, можно предположить, что результаты работы алгоритма будут различаться в зависимости от испытуемого. Очевидно, что качественный алгоритм должен давать одинаково хорошие результаты вне зависимости от аутентифицируемого. Чем больше будет разброс в значениях EER для разных испытуемых, тем сложнее будет использовать алгоритм на практике. Поэтому другим важным

параметром оценки результата работы алгоритма является среднеквадратичное отклонение, рассчитанное среди значений EER, полученных при аутентификации различных испытуемых.

3 Существующие подходы

Существующие подходы базируются на различных признаках, собираемых при работе пользователя с клавиатурой и создаваемых по этим признакам моделях. Кратко опишем некоторые из них, рассмотренные в статьях [3], [4], [5], [6], [7], более подробный обзор был приведен в статье [8].

Модель, основанная на измерении времени удержания. В этом методе, описанном в статье [4], в качестве модели берется вектор X из n элементов, каждый из которых соответствует одной кнопке на клавиатуре и является парой (M_k, D_k): элемент M_k — среднее время удержания клавиши k, а D_k – величина стандартного отклонения для клавиши k. С помощью этого подхода удалось добиться результатов FAR и FRR менее 0,1 одновременно.

Наблюдение за порядком нажатия и отпускания кнопок [3]. Этот метод основан на предположении, что пользователи при наборе символов на клавиатуре иногда нажимают следующую кнопку до того, как отпустят текущую (в таком случае имеет место так называемый «обмен»). Наблюдая за последовательностью нажатий и отпусканий кнопок и подсчитывая число «обменов», можно составить модель пользователя. Было выяснено, что значения FAR и FRR для результатов применения этого метода сильно зависят от пар пользователей, в экспериментах проявлялись доли ошибок от 0 до 0,7, что не позволяет признать этот метод подходящим для использования иначе, кроме как в связке с другими подходами.

Относительная скорость печати. Существует предположение, что для каждой пары кнопок скорость нажатий остается примерно одинаковой вне зависимости от текста. Поэтому высказывается предложение замерять скорости набора пар кнопок и использовать их в качестве модели для пользователя. Для посторения модели использовалось расстояние между упорядоченными по скорости набора векторами пар кнопок, предложенное в [5]. В результате, расстояния между двумя векторами одного и того же пользователя составили в среднем 0,3192, расстояние между векторами разных пользователей – в среднем 0,529. Поэтому предлагается судить об успешности аутентификации пользователя только при различии в векторах меньшем, чем 0,3, а о неуспешности – при различии, большем 0,6.

Использование правой и левой клавиш `Shift`. В этом подходе, описанном в [3], предполагается, что при наборе текста разные люди используют правую и левую клавиши `Shift` по-разному, и это можно исполь-

зовать для аутентификации. По собранным в эксперименте данным люди были разделены на 4 класса: те, кто пользуется только правой или только левой клавишей `Shift`, и те, кто отдает предпочтение левой или правой клавише, но при этом иногда использует и другую. Очевидно, что попадание пользователя в ожидаемый класс не дает права признать попытку аутентификации удачной, так как классов всего 4, и вероятность принять атакующего пользователя за легитимного весьма высока. Однако попадание в чужой класс дает весомое основание отвергнуть попытку аутентификации.

Метод для коротких буквенных или цифровых паролей. В подходе, предлагаемом в статье [6], в качестве модели авторами были взяты замеры продолжительности нажатий клавиш, однако вычислялись они тремя разными способами. В качестве алгоритма, с помощью которого осуществлялось обучение, был взят мультиклассовый линейный SVM [7], так как он демонстрирует высокие результаты на данных несложной структуры. Кроме того, при сборе данных испытуемые были разделены на две группы: одну из них проинформировали о проведении эксперимента, а другую нет. Эксперимент показал, что информированность пользователей влияет на получаемые результаты: полученные значения FAR и FRR оказались в $3 - 5$ раз меньше в группе информированных пользователей (0,01 – 0,03).

4 Предлагаемый подход

4.1 Описание используемой модели и алгоритмов

Пространство признаков для алгоритма было выбрано следующим образом: каждая попытка набора пароля представляет собой вектор из значений задержек между различными событиями нажатия и отпускания клавиш. Этими значениями для каждой клавиши, начиная со второй, являются: промежуток времени удержания клавиши, временной промежуток между двумя нажатиями и временной промежуток между нажатием клавиши и отпусканием предыдущей. Для первой клавиши собирается лишь ее время удержания. Это пространство признаков было предложено в [9], на нем были достигнуты высокие результаты, и для объективного сравнения с другими алгоритмами из [9] оно было выбрано для предлагаемого подхода.

В качестве предлагаемого подхода рассматривается возможность применения для задачи статической аутентификации подхода [10], который зарекомендовал себя в различных областях, связанных с обнаружением вторжений. В данной статье будет проведено адаптирование описываемого метода для поставленной задачи.

В основе этого подхода лежат две основных идеи: использование

функций ядра для вычисления расстояний и теории нечетких множеств для построения модели пользователя. Остановимся на каждой из них подробнее.

4.1.1 Функции ядра

Механизм функций ядра ([11]) широко применяется в различных алгоритмах машинного обучения (SVM и пр.). Как известно, принцип использования функций ядра заключается в следующем.

Пусть задано некоторое пространство X и преобразование φ, которое осуществляет отображение из X в многомерное (бесконечномерное) Гильбертово пространство H:

$$\varphi : X \longrightarrow H \tag{1}$$

Назовем H *пространством признаков* и определим в нем *функцию ядра*, которая будет являться в нем скалярным произведением:

$$K(x, y) = \langle \varphi(x), \varphi(y) \rangle \tag{2}$$

Можно рассматривать функцию $K(x, y)$ как меру близости двух элементов из пространства X, и на основании ее строить функции расстояния.

Заметим, что необходимости вычислять $\varphi(x)$ и $\varphi(y)$ в таком случае нет, а отображение целиком определяется ядром $K(x, y)$. Именно отсутствие необходимости вычисления, а значит временных затрат и затрат по памяти, является одним из главных преимуществ использования функций ядра.

Необходимо также упомянуть, что использование различных ядер придает рассматриваемому подходу определенную гибкость и расширяет возможности по его конфигурации, может быть подобрана для достижения оптимальных результатов.

В используемом подходе будет рассматриваться следующая функция расстояния, основанная на $K(x, y)$:

$$d(x, y) = \sqrt{K(x, x) - 2K(x, y) + K(y, y)} \tag{3}$$

Особое внимание будет уделено использованию в качестве ядер скалярного произведения:

$$K(x, y) = \langle x, y \rangle \tag{4}$$

и ядра Гаусса:

$$K(x, y) = \frac{e^{-(x-y)^2}}{2\sigma^2} \tag{5}$$

(в последнем случае значение σ может быть подобрано для достижения наилучших результатов).

4.1.2 Нечеткая кластеризация в пространстве признаков

Нечеткое множество ([12]) — это расширение классического понятия множества. Мера принадлежности элемента x на нечетком множестве X может принимать значения на $[0, 1]$.

В предлагаемом подходе происходит поиск нечеткого кластера, содержащего образы элементов исходного пространства X. В данном случае, мера принадлежности образа нечеткому кластеру может служить мерой его соответствия кластеру, т.е. величиной, обратной к аномальности. Образы с малой мерой соответствия (меньше порога, установленного для пользователя) будут считаться нелегитимными попытками аутентификации.

В [10] показано, что поиск нечеткого кластера в гильбертовом пространстве признаков приводит к следующей задачи нечеткой кластеризации:

$$\begin{gathered}\min_{U\in[0,1]^N,\ a\in H} J(U,a),\\ J(U,a)=\sum_{i=1}^{N}(u_i)^m(\varphi(x_i)-a)^2-\eta\sum_{i=1}^{N}(1-u_i)^m\end{gathered} \tag{6}$$

Здесь H – пространство признаков, в котором находятся вектора, представляющие попытки авторизации, a – центр кластера, соответствующего попыткам авторизации легитимного пользователя, N – число легитимных попыток авторизации, использующихся для обучения, $u_i \in [0, 1]$ – степень принадлежности образа $\varphi(x_i)$ кластеру, и, как следствие, мера соответствия кластеру образа x_i, m – степень нечеткости, а η – расстояние от центра кластера, где степень принадлежности считается равной $0,5$.

Для минимизации $J(U, a)$ в [10] предлагается использовать следующий итеративный алгоритм на основе блочного координатного спуска:

Шаг 0. Инициализация U и η в случае его упрощенного подбора

Шаг 1. Найти центр кластера.

$$\langle a,a\rangle=\left(\sum_{j=1}^{N}u_j^m\sum_{i=1}^{N}u_i^mK(x_i,x_j)\right)/\left(\sum_{i=1}^{N}u_i^m\right)^2. \tag{7}$$

Шаг 2. По всем j из $[1, N]$ рассчитать расстояние до нового центра кластера:

$$\langle\varphi(x_j),a\rangle=\left(\sum_{i=1}^{N}u_i^mK(x_i,x_j)\right)/\left(\sum_{i=1}^{N}u_i^m\right). \tag{8}$$

Шаг 3. По всем j из $[1, N]$ рассчитать новые степени принадлежности обучающих векторов:

$$u_j = \frac{1}{1 + \left(\frac{\langle a,a \rangle + K(x_j,x_i) - 2\langle \varphi(x_j),a \rangle}{\eta} \right)^{m-1}}. \tag{9}$$

Шаги 1-3 повторять, пока не станет верным неравенство

$$||U^l - U^{l-1}|| < \varepsilon, \tag{10}$$

где l - номер шага, ε - необходимая точность.

В таком случае функция аномальности $F(x, X)$ принимает вид

$$F(x, X) = \frac{1}{1 + \left(\frac{\langle a,a \rangle + K(x,x) - 2\langle \varphi(x,X),a \rangle}{\eta} \right)^{m-1}}, \tag{11}$$

где

$$\langle \varphi(x, X), a \rangle = \left(\sum_{i=1}^{N} u_i^m K(x_i, x) \right) / \left(\sum_{i=1}^{N} u_i^m \right), \tag{12}$$
$$u_1 \dots u_N \in X, |X| = N.$$

Для использования этого алгоритма необходимо выбрать начальные значения для элементов U и η. В качестве начальных значений элементов U можно использовать одинаковые значения, равные $u_i^0 = 1/N$.

Что касается значения η, его также можно подобрать двумя способами. В случае упрощенного подбора, в качестве η берется квадрат дистанции между двумя наиболее удаленными элементами обучающего множества, и не меняется на протяжении работы всего алгоритма. Альтернативно, для оценки радиуса кластера на каждой итерации используется квадрат расстояния от центра кластера до наиболее удаленного вектора, не являющегося выбросом. Выбросами предлагается считать долю наиболее удаленных от центра кластера векторов, которая является параметром алгоритма. В этом случае значение $F(x, X)$ будет более $0,5$, если вектор x лежит внутри кластера, менее $0,5$, если он лежит вне кластера, и равен $0,5$, если вектор x находится на границе кластера. Поэтому в модельной реализации значение $0,5$ используется в качестве начального значения минимальной степени типичности для признания попытки набора легитимной.

4.1.3 Предобработка данных

Так как в собираемых данных признаки могут быть разнородными, и их значения, соответственно, лежат в разных границах, имеет смысл, учитывая особенности алгоритма, произвести нормализацию, то есть приведение области значений признаков к некоторым одинаковым для всех признаков границам.

Наилучший метод нормализации для данной задачи был подобран экспериментально на стандартном наборе данных. Им является нормализация на значение абсолютного отклонения. Пусть для обучения в N попытках набора пароля встречается признак p. Тогда для этого признака коэффициент нормализации для вектора x будет выглядеть следующим образом:

$$W_p = \sum_{i=1}^{N} \frac{|x_i - \bar{x}|}{N}, x'_p = x_p / W_p, \tag{13}$$

где $\bar{x}$ – среднее арифметическое значение элементов вектора x, а x' – нормализованный вектор признаков p.

Среди других возможных коэффициентов нормализации также можно использовать квадратный корень из приведенного выше значения, межквартильный диапазон (IQR), и некоторые другие, но, как будет показано ниже, из рассмотренных коэффициентов именно значение абсолютного отклонения дает наилучшие результаты. Кроме того, была произведена предобработка данных, заключающаяся в замене каждого значения x во входных данных на $\ln(x + C)$ где параметр C брался достаточно большим, чтобы $\ln(x + C)$ был определен. Это обосновано тем, что случайные величины, описывающие отдельные признаки набора пароля имеют, как было выяснено во время проведения эксперимента, распределение, близкое к логнормальному. В результате этой предобработки и нормализации на значение абсолютного отклонения, были получены предобработанные входные данные.

5 Экспериментальное исследование

Для сравнения метода с уже существующими и подборки оптимальных параметров алгоритма классификации было решено провести серию экспериментов. Для этого предложенный алгоритм был реализован на языке программирования R [13].

5.1 Экспериментальные данные

В качестве экспериментальных данных для оценки алгоритма было решено использовать данные, предоставляемые авторами статьи [9]. Они были

выбраны исходя из того, что соответствуют постановке задачи статической аутентификации, с их использованием были проведены эксперименты, описанные в [9], они имеют большой объем и являются репрезентативными.

5.2 Постановка эксперимента

Для того, чтобы возможно было сравнить результаты предлагаемого алгоритма с результатами других алгоритмов, необходимо провести эксперимент, по условиям полностью совпадающий с тем, на котором были получены результаты другого алгоритма. Для этого необходимы те же данные, те же объемы данных для обучения и те же объемы данных для оценки полученной модели. Так как данные были взяты из статьи [9], было решено взять параметры постановки эксперимента из этого же источника. Эти параметры таковы: для создания модели одного пользователя алгоритму обучения на вход подается 200 **первых** попыток набора пароля этим пользователем, а для оценки полученной модели алгоритму детектирования на вход подается 200 **последних** попыток набора пароля этим пользователем и по 5 **первых** попыток набора пароля 50 остальными пользователями, что в общей сложности составляет 450 попыток набора.

Эта модель проведения эксперимента довольно точно соответствует реальному сценарию использования системы аутентификации, основанной на анализе нажатий клавиш: для обучения используются первые попытки, исходя из того, что обучение происходит в момент смены пароля на новый, когда легитимный пользователь только начинает вырабатывать характерные для него черты набора пароля. Для детектирования используются последние попытки, в которых легитимный пользователь проявляет выработанные особенности набора пароля, а нелегитимные, будучи ранее незнакомыми с паролем, их не проявляют.

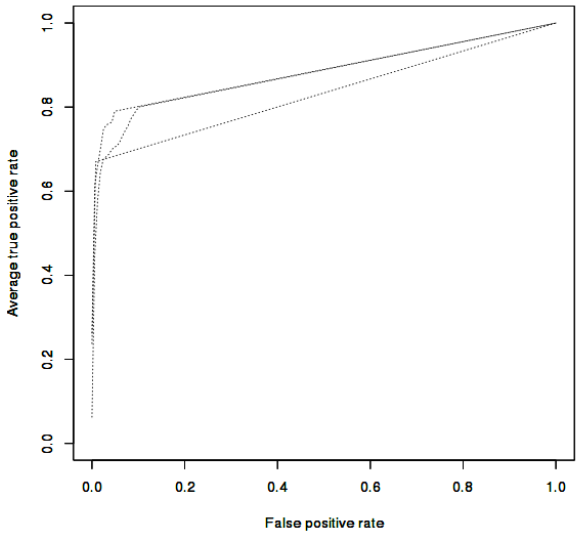
5.3 Подбор параметров и результаты

Для того, чтобы оценить влияние всех параметров и выбрать те их значения, которые наилучшим образом подходят для задачи, было проведено несколько серий экспериментов, в ходе которых значения параметров были проварьированы в заданных отрезках, замерены результаты, и сделаны соответствующие выводы о том, как тот или иной параметр оказывает влияние на результат. После подборки наилучшего значения одного из параметров его значение фиксировалось и начиналась подборка значения другого параметра. Варьируемые параметры в дальнейшем будут описаны в том порядке, в котором они подбирались.

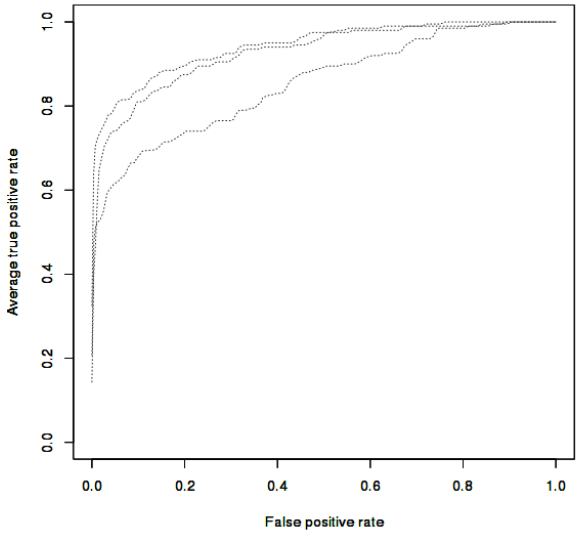
5.3.1 Ядро и его параметры

В качестве ядра в экспериментах использовались две наиболее популярные функции: скалярное произведение (4) и ядро Гаусса (5).

При использовании скалярного произведения в качестве ядра наилучшим полученным результатом оказалось значение EER, равное 0,31, что говорит о весьма низкой точности алгоритма при использовании этой функции. При переходе на ядро Гаусса встала необходимость проварьировать также его параметр — σ.



ROC-кривые при малых значениях σ в ядре Гаусса



ROC-кривые при больших значениях σ в ядре Гаусса.

При малых (порядка 10^{-1}–10^{0}) значениях этого параметра наблюдалось медленное увеличение верно распознанных попыток при снижении порога (рис. 5.3.1). При сильном увеличении значения σ относительно оптимального наблюдалась общая деградация ROC-кривой (рис 5.3.1), без каких-либо характерных особенностей, при дальнейшем увеличении итерационный алгоритм прекращал сходиться, что, скорее всего, вызвано ошибками в округлении в расчетах с использованием чисел с плавающей точкой. Оптимум значения σ для представленной выборки оказался равен $\sigma = 10^{1}$.

5.3.2 Нахождение значения η

Как было описано выше, для подборки значения η, представляющего собой расстояние от центра кластера, при котором степень принадлежности считается равной 0,5, существует два способа, упрощенный и итеративный.

При использовании упрощенного способа никаких дополнительных параметров не требуется. Наилучшее значение EER, полученное с помощью упрощенного способа, составило 0,188. При использовании итеративного способа необходимо задать долю выбросов. Экспериментальным путем было установлено, что оптимальная ожидаемая доля выбросов составляет 0,1, тогда использование итеративного алгоритма дает значение EER 0,178. Варьирование доли выбросов в границах 0,05-0,2 показывает вариацию значения EER в границах 0,181 – 0,178.

5.3.3 Методы нормализации входных данных

Наиболее существенные улучшения результатов были получены после того, как перед обработкой данных была проведена их нормализация. Это объясняется тем, что параметры по своей природе обладают сильно разнящимися значениями (так, например, время удержания кнопки бывает лишь строго положительным, в то время как время между отпусканием предыдущей кнопки и нажатием текущей зачастую бывает отрицательным.

Поэтому были рассмотрены некоторые методы нормализации параметров входных данных, а именно нормализация на квадратный корень из дисперсии, значение абсолютного отклонения, квадратный корень из абсолютного отклонения, межквартильное расстояние (IQR), медиану абсолютного отклонения.

Наиболее точные результаты были получены при использовании для нормализации абсолютного отклонения и корня из него. Следует упомянуть, что на некоторых пользователях лучше оказался первый тип нормализации, а на других – второй тип. В связи с этим была предпринята попытка выяснения лучшего способа нормализации в момент обучения. Для этого использовалась кросс-валидация: обучающая выборка делилась пополам, первая часть выборки нормализовалась на значение абсолютного отклонения и на корень из него, после чего на ней обучались, соответственно, две модели и тестировались на второй половине обучающей выборки. После получения результатов обучение аналогично производилось на второй половине, а тестирование на первой. Лучшим методом нормализации для пользователя признавался тот, чей средний EER оказывался меньше. Однако подобно введение кросс-валидации не привело к улучшению показателя EER.

Кроме того, была внедрена предобработка данных, заключающаяся в замене каждого значения x во входных данных на $\ln(x + C)$, описанная в п. 4.1.3. В результате этой предобработки и нормализации на значение абсолютного отклонения, был получен итоговый результат.

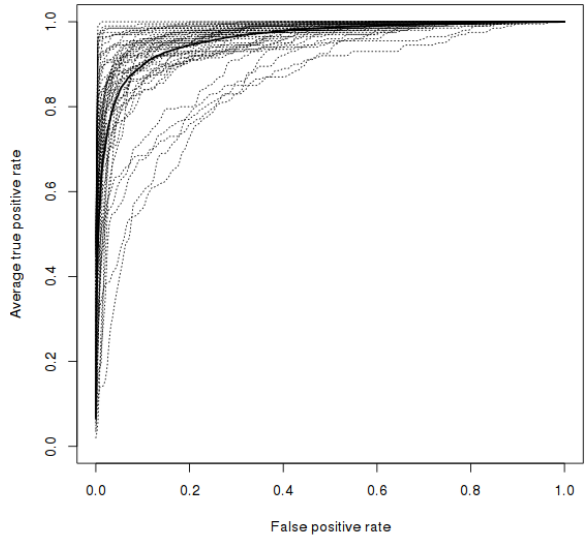
5.4 Результаты экспериментов и выводы

Результаты работы алгоритмов

Алгоритм	Значение EER	Значение σ
Предложенный алгоритм	0,092	0,05
Manhattan Scaled [1]	0,096	0,069
Nearest Neighbor	0,100	0,064
Outlier Count	0,102	0,077
SVM	0,102	0,065
Mahalanobis	0,110	0,065
Manhattan Filter	0,136	0,083
Manhattan	0,153	0,092
Neural Network	0,161	0,080

Как видно из таблицы 5.4, предложенный алгоритм оказался лучше наиболее успешных из рассмотренных в [9]. Это позволяет говорить о высоком качестве алгоритма и его применимости к поставленной задаче.

На рис 5.4 представлены ROC-кривые, полученные на упорядоченных данных с использованием предложенного алгоритма. Жирная линия – усредненная ROC-кривая.



ROC кривые, полученные на упорядоченных данных с использованием предложенного алгоритма

6 Заключение

В статье исследовалась задача статической аутентификации пользователей по данным об их работе с клавиатурой.

В качестве основного результата работы для решения задачи был применен метод, основанный на нечетких множествах и использовании

[1] Лучший из рассмотренных в [9]

функций ядра, ранее дававший высокие результаты в задачах обнаружения атак. Был реализован соответствующий алгоритм, проведены серии экспериментов на эталонном наборе данных с целью подбора оптимальных параметров, проведено сравнение с уже существующими методами, в ходе которых предложенный метод продемонстрировал высокие результаты, тем самым доказав применимость на практике к задачам подобного рода.

Список литературы

[1] *Goldwasser, Shafi, Silvio Micali, and Ronald L. Rivest.* A digital signature scheme secure against adaptive chosen-message attacks // SIAM Journal on Computing, 1988. Vol. 17. No. 2. P. 281–308.

[2] *Micki Krause, Harold F. Tipton* Handbook of Information Security Management Auerbach Publications, CRC Press LLC

[3] *Lau E., Liu X., Xiao C., Yu X.* Enhanced user authentication through keystroke biometrics. – Massachusetts Institute of Technology, 2004.

[4] *F. W. M. H. Wong, A. S. M. Supian,and A. F. Ismail.* Enhanced User Authentication through Typing Biometrics with Artificial Neural Networks and K-Nearest Neighbor Algorithm.“ IEEE, p. 911-915, 2001.

[5] *F. Bergadano, D. Gunetti, and C. Picardi.* User Authentication through Keystroke Dynamics.” ACM Transactions on Information and System Security, Vol. 5, No. 4, Nov., p. 367-397, 2002.

[6] *Saggio G., Costantini G., Todisco M.* Cumulative and ratio time evaluations in keystroke dynamics to improve the password security mechanism // Journal of Computer and Information Technology, 2011. Vol. 1. No. 2. P. 4–11.

[7] *Sung K. S., Cho S.* GA SVM wrapper ensemble for keystroke dynamics authentication // Proc. of International Conference on Biometrics. – Hong Kong: ICB, 2004. P. 654–660.

[8] *В.Ю. Каганов, А.К. Королёв, М.Н. Крылов, И.В. Машечкин, М.И. Петровский.* Методы активной аутентификации на основе анализа динамики работы пользователей с клавиатурой. – Информатика и ее применения (Том 7, Выпуск 3, 2013)

[9] *Kevin S. Killourhy and Roy A. Maxion.* Comparing Anomaly Detectors for Keystroke Dynamics IEEE Computer Society Press, Los Alamitos, California, 2009.

[10] *Petrovsky M. I.* Outlier Detection Algorithms in Data Mining Systems // Programming and Computer Software, 2003. Vol. 29. No 4. P. 228–237.

[11] *Scholkopf B., Smola A.* Learning with kernels, 2002.

[12] *Zadeh L. A.* Fuzzy sets // Information and Control, 1965. Vol. 8. No 3. P. 338—353.

[13] *Team, R. C.* R: A language and environment for statistical computing // R foundation for Statistical Computing, 2005.