УДК 004.93
DOI 10.17513/snt.40044

МЕТОДЫ АНАЛИЗА КОМПЬЮТЕРНОГО ПОЧЕРКА ДЛЯ БИОМЕТРИЧЕСКОЙ ИДЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЕЙ В ПРОЦЕДУРЕ ПРОКТОРИНГА

Родионов А.В., Шафаревич А.Д.

ФГБОУ ВО «Байкальский государственный университет», Иркутск, e-mail: avr-v@yandex.ru, dmitrievaanastasiad@yandex.ru

В данной статье исследуется применимость нескольких алгоритмов для цифровой идентификации пользователя на основе компьютерного почерка. Компьютерный почерк представляет собой уникальный набор динамических и статических характеристик ввода информации посредством какого-либо устройства ввода/вывода. Целью данного исследования является сравнение качества классификации различных методов анализа компьютерного почерка для биометрической идентификации пользователей в контексте процедуры прокторинга. Для анализа взяты три различных алгоритма: метод k-ближайших соседей, длинная цепь элементов краткосрочной памяти и сверточная нейронная сеть. За основу принят анализ уникальных характеристик динамики нажатия клавиш, таких как временные интервалы между нажатиями, длительность нажатий. Описаны основные метрики оценки и характеристики настройки алгоритмов. В исследовании использовались следующие метрики оценки качества моделей: Accuracy, F1-мера и ROC-AUC, также представлены графики ROC-кривых. Обучение моделей проведено, результаты их работы представлены в сводной таблице. Сделан вывод о том, что все подходы применимы к идентификации пользователей, но наиболее высокие результаты показала сверточная нейронная сеть по всем примененным в работе метрикам. Предложенный инструмент идентификации пользователя по его клавиатурному почерку может быть использован для проведения процедуры аттестации путем интеграции с существующими системами дистанционного обучения.

Ключевые слова: дистанционное обучение, прокторинг, компьютерный почерк, нейронные сети, идентификация

METHODS OF COMPUTER HANDWRITING ANALYSIS FOR BIOMETRIC IDENTIFICATION OF USERS IN PROCTORING PROCEDURE

Rodionov A.V., Shafarevich A.D.

Baikal State University, Irkutsk, e-mail:avr-v@yandex.ru, dmitrievaanastasiad@yandex.ru

This article examines the applicability of several algorithms for digital user identification based on computer handwriting. Computer handwriting is a unique set of dynamic and static characteristics of information input through any input/output device. The purpose of this study is to compare the classification quality of various computer handwriting analysis methods for biometric user identification in the context of a proctoring procedure. Three different algorithms were used for analysis: k-nearest neighbors, long short-term memory, and convolutional neural network. The basis is an analysis of the unique characteristics of the dynamics of keystrokes, such as time intervals between keystrokes and the duration of keypresses. The main evaluation metrics and characteristics of tuning algorithms are described. The following metrics for assessing the quality of models were used in the study: Accuracy, F1-measure and ROC-AUC, graphs of ROC curves are also presented. The models have been trained, and the results of their work are presented in the summary table. It is concluded that all approaches are applicable to user identification, but the convolutional neural network showed the best results for all compared metrics. The proposed tool for identifying a user by his keyboard handwriting can be used to carry out the certification procedure by integrating with existing distance learning systems.

Keywords: distance learning, proctoring, keyboarding, neural networks, identification

В эпоху цифровизации образовательного процесса дистанционное обучение и проведение аттестации в режиме онлайн приобретают все большую популярность и значимость. Важным аспектом обеспечения качества и надежности этих процессов является применение эффективных методов контроля, в том числе прокторинга. Прокторинг – это комплекс мероприятий по контролю за дистанционными экзаменами с целью предотвращения академических нарушений. В рамках прокторинга используются различные технологии и методы, включая видеонаблюдение, мониторинг активности на экране, а также анализ поведения пользователя. К последнему направлению может быть отнесены и методы биометрической идентификации, которые включают в себя процессы распознавания индивидуумов на основе одного или нескольких уникальных физиологических или поведенческих признаков. К традиционным методам относятся отпечатки пальцев, распознавание лица, голоса, сетчатки и радужки глаза. Определенного внимания заслуживает и анализ так называемого компьютерного почерка, который может предложить дополнительные возможности для

проведения процедуры прокторинга в тех задачах, которые могут быть реализованы посредством выполнения заданий (предполагающий ответ в виде «печатного» текста) и тестирования (при наличии «открытых» вопросов, также предполагающих набор текста в виде ответа).

Компьютерный почерк представляет собой уникальный набор динамических и статических характеристик ввода информации посредством какого-либо устройства ввода/вывода. Например, к динамическим характеристикам ввода текста пользователем через клавиатуру (клавиатурный почерк) можно отнести скорость набора, ритм и временные интервалы между последовательными нажатиями клавиш, в то время как статические характеристики включают в себя общую точность и стабильность набора. Эти параметры формируют индивидуальный «цифровой след», который может быть использован для идентификации личности. Эффективность клавиатурного почерка как инструмента аутентификации обусловлена его способностью к выявлению и верификации уникальных поведенческих паттернов пользователя, что делает его значимым инструментом в обеспечении безопасности, особенно в условиях дистанционного обучения и проведения онлайн-тестирования.

Целями данного исследования являются сравнение моделей и методов классификации, которые могут быть использованы для анализа компьютерного почерка (на примере датасета клавиатурного почерка) с целью биометрической идентификации пользователей в контексте процедуры прокторинга, а также оценка их эффективности и точности при идентификации индивидуальных поведенческих характеристик, проявляемых в процессе ввода текста на клавиатуре.

Материалы и методы исследования

В рамках исследования были рассмотрены и сравнены три различных алгоритма классификации для анализа компьютерного почерка: метод k-ближайших соседей (k Nearest Neighbors – kNN), длинная цепь элементов краткосрочной памяти (Long short-term memory – LSTM) и сверточная нейронная сеть (Convolutional neural network – CNN).

Метод kNN – один из самых известных метрических алгоритмов классификации [1, 2] (метрический алгоритм – это алгоритм классификации, основанный на вычислении оценок сходства между объектами с помощью функции расстояния между объектами). В процессе работы алгоритма k-ближайших соседей выполняются следующие ключевые шаги [3]:

1) определение расстояния от объекта, который нужно классифицировать, до каждого элемента в обучающем наборе данных, уже помеченного определенным классом;

2) выбор k элементов из обучающего набора, для которых расстояние до целевого объекта является минимальным (выбор значения k на начальном этапе производится случайным образом, после чего оптимальное значение k подбирается итеративно на основе анализа точности предсказаний для каждого из рассмотренных значений k);

3) классификация объекта на основе наиболее часто встречаемого класса среди k ближайших соседей, где итоговая принадлежность к классу может быть выражена как в виде числового значения, так и в форме названия класса, исходя из первоначальной маркировки классов в обучающем наборе данных.

Алгоритм может использовать разные функции расстояний: евклидово расстояние, манхэттенское расстояние, расстояние Махаланобиса и т.п. В данной работе представлена реализация алгоритма на основе евклидова расстояния. Сущность данной метрики заключается в определении кратчайшего расстояния между указанными точками, выраженного в виде длины прямой, соединяющей их, вычисляемого по теореме Пифагора [4]. В общем случае для n-мерного пространства:

$$p(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Подробнее с архитектурой алгоритма kNN и особенностями параметров можно ознакомиться в работе [5].

LSTM представляет особую разновидность архитектуры рекуррентных нейронных сетей (Recurrent neural network – RNN) для улучшения процесса обучения сети на основе длительных временных зависимостей. Эта модель была впервые представлена в 1997 году исследователями Сеппом Хохрайтером и Юргеном Шмидхубером [6]. Основная цель разработки LSTM заключалась в преодолении сложностей, связанных с изучением долгосрочных зависимостей, за счет способности сохранять информацию на различные промежутки времени – от краткосрочных до долгосрочных. Отличительной особенностью LSTM является отсутствие применения функции активации внутри рекуррентных блоков, что позволяет избежать размывания значений и исчезновения градиента в процессе обучения с использованием обратного распространения ошибок через время. Благодаря этому LSTM эффективно учится распознавать и

сохранять информацию на длительные периоды, что является естественной характеристикой этой модели, а не результатом специализированного обучения [6, 7].

Первым шагом работы LSTM является определение той информации, которая должна быть исключена из состояния ячейки. Эту функцию выполняет сигмоидальный слой forget gate layer. Этот слой генерирует числа между 0 и 1 для каждого атрибута в состоянии клетки, где 1 означает полное сохранение информации, а 0 – ее полное удаление. Следующий шаг включает в себя решение о том, какая новая информация будет сохранена в клетке. На этом этапе задействованы два процесса: первый, где сигмоидальный слой input layer gate определяет, какие данные необходимо обновить, и второй, где слой tanh-слой формирует вектор кандидатов на добавление в ячейку. Заключительная часть процесса заключается в определении, какая информация будет передана на выход [7].

CNN представляет особый вид нейронных сетей прямого распространения. Под прямым распространением понимается то, что распространение сигналов по нейронам идет по порядку, от первого слоя до последнего. Скрытых слоев в сети может быть достаточно много, все зависит от количества данных и сложности задачи [8]. Функционирование сверточных нейронных сетей часто описывается как процесс перехода от специфических атрибутов данных к более обобщенным характеристикам, а затем к еще более генерализованным понятиям, достигая уровня высокоуровневых концепций. В ходе этого процесса сеть автоматически формирует необходимую иерархию абстрактных признаков, отсеивая менее значимые детали и акцентируя внимание на ключевых аспектах. Однако подобная интерпретация носит скорее метафорический или иллюстративный характер. В действительности характеристики, генерируемые этими сетями, часто оказываются настолько сложными для понимания и трактовки, что при их практическом применении часто не стремятся разобраться в их сущности или корректировать их. Вместо этого, стремясь улучшить результаты распознавания, предпочтение отдают изменению структуры и архитектуры сети. Подробное описание механизма работы сверточных нейронных сетей представлено в работе [9].

Анализ проводился на основе датасета, предоставленного Кевином Киллоури и Ройем Максионом, который содержит данные о динамике нажатия клавиш [10]. Датасет представляет собой таблицу с полями, содержащими информацию о временных метках нажатия клавиш при вводе текста [11], имена столбцов кодируют тип информации о времени: например, столбец «DD. period.t» – описывается время, от нажатия клавиши «.» до нажатия «t» (рис. 1):

− H – время от нажатия до отпускания одной клавиши;

− DD – время от нажатия одной клавиши до нажатия следующей клавиши;

− UD – время от отпускания клавиши до нажатия следующей клавиши.

Для оценки эффективности моделей использовались метрики точности (Accuracy), F1-меры и площади под ROC-кривой (ROC AUC).

Результаты исследования и их обсуждение

Для проведения исследования было использовано следующее программное обеспечение:

1) язык программирования Python, для которого существует множество библиотек работы с данными и нейронными сетями;

2) среда разработки Pycharm IDE и Google Colab.

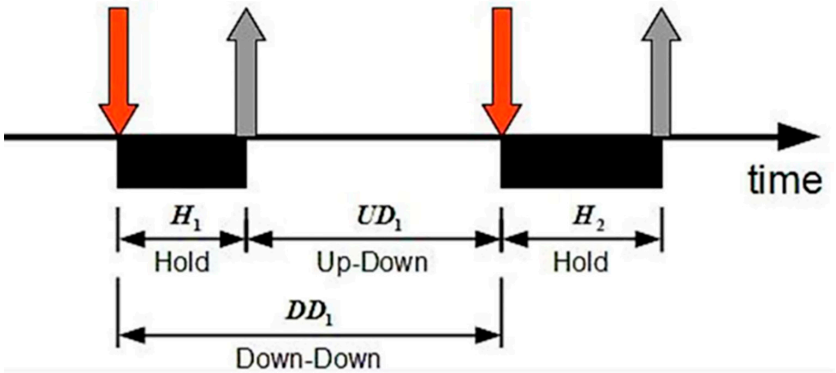

Рис. 1. Схема времени нажатия и опускания клавиши

```
Epoch 18/30
447/447 [==============================] - 7s 15ms/step - loss: 0.0030 - accuracy: 0.9811
Epoch 19/30
447/447 [==============================] - 8s 17ms/step - loss: 0.0029 - accuracy: 0.9835
Epoch 20/30
447/447 [==============================] - 7s 17ms/step - loss: 0.0031 - accuracy: 0.9802
Epoch 21/30
447/447 [==============================] - 7s 17ms/step - loss: 0.0029 - accuracy: 0.9821
Epoch 22/30
447/447 [==============================] - 7s 16ms/step - loss: 0.0030 - accuracy: 0.9809
Epoch 23/30
447/447 [==============================] - 7s 17ms/step - loss: 0.0028 - accuracy: 0.9832
Epoch 24/30
447/447 [==============================] - 7s 16ms/step - loss: 0.0025 - accuracy: 0.9851
Epoch 25/30
447/447 [==============================] - 7s 16ms/step - loss: 0.0023 - accuracy: 0.9857
Epoch 26/30
447/447 [==============================] - 7s 15ms/step - loss: 0.0026 - accuracy: 0.9853
Epoch 27/30
447/447 [==============================] - 7s 16ms/step - loss: 0.0025 - accuracy: 0.9843
Epoch 28/30
447/447 [==============================] - 8s 17ms/step - loss: 0.0025 - accuracy: 0.9851
Epoch 29/30
447/447 [==============================] - 7s 16ms/step - loss: 0.0024 - accuracy: 0.9859
Epoch 30/30
447/447 [==============================] - 7s 16ms/step - loss: 0.0022 - accuracy: 0.9865
```

Рис. 2. Фрагмент обучения CNN модели

Для расчетов в статье алгоритм kNN реализован посредством языка программирования Python с использованием библиотеки Scikit-learn, LSTM реализована на Python с использованием библиотеки Keras, CNN реализована на языке Python с использованием библиотек Keras и TensorFlow. Нейронные сети LSTM и CNN имели близкие настройки с целью лучшей сравнимости результатов, в частности были выбраны следующие параметры:

– количество эпох обучения = 30 (для реального использования это значение, конечно, мало, но для исследования сравнения моделей этого вполне достаточно, поскольку, чем больше эпох, тем дольше будет идти обучение);

– количество слоев модели = 5;

– функция активации = сигмоидная;

– алгоритм оптимизации = Adam;

– batch – количество сэмплов, которые необходимо взять для обновления параметров модели, было установлено в 32.

Фрагмент обучения CNN модели представлен на рисунке 2.

При исследовании весь датасет был разделен на тренировочные и тестовые данные в пропорциях 70:30 соответственно. Оценку качества моделей проводили с использованием следующих метрик [12]:

1. Accuracy – метрика, которая описывает общую точность предсказания модели по всем классам. Она рассчитывается как отношение количества правильных прогнозов к их общему количеству,

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}},$$

где TP (true positive) – классификатор верно отнес объект к рассматриваемому классу, TN (true negative) – классификатор показывает, что объект не принадлежит к рассматриваемому классу, FP (false positive) – классификатор неверно отнес объект к рассматриваемому классу, FN (false negative) – классификатор неверно показывает, что объект не принадлежит к рассматриваемому классу.

2. F_1-мера – среднее гармоническое между precision и recall [13]:

$$F_1 = \frac{2}{\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}} =$$

$$= 2\frac{\text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}} = \frac{\text{TP}}{\text{TP} + \frac{\text{FP} + \text{FN}}{2}},$$

где Recall (полнота) – показывает отношение верно классифицированных объектов класса к общему числу элементов этого класса,

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$

Precision (точность) – показывает долю верно классифицированных объектов среди всех объектов, которые к этому классу отнес классификатор,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

3. Кривая ROC – это график, который иллюстрирует качество работы классификационной модели. Ось X данного графика представляет собой FPR (False positive rate, частота ложноположительных результатов), а ось Y – TPR (True positive rate, частота истинно-положительных результатов). Идеальная модель классификации будет стремиться к точке в верхнем левом углу графика, где TPR равно 1, а FPR равно 0. На основе кривой ROC строится AUC – Area Under the ROC Curve. Данная мера позволяет суммировать производительность модели одним числом – площадью область под кривой ROC. Оценка AUC варьируется от 0 до 1, где 1 – идеальный показатель, а 0,5 означает, что модель выдает ответ случайно [14].

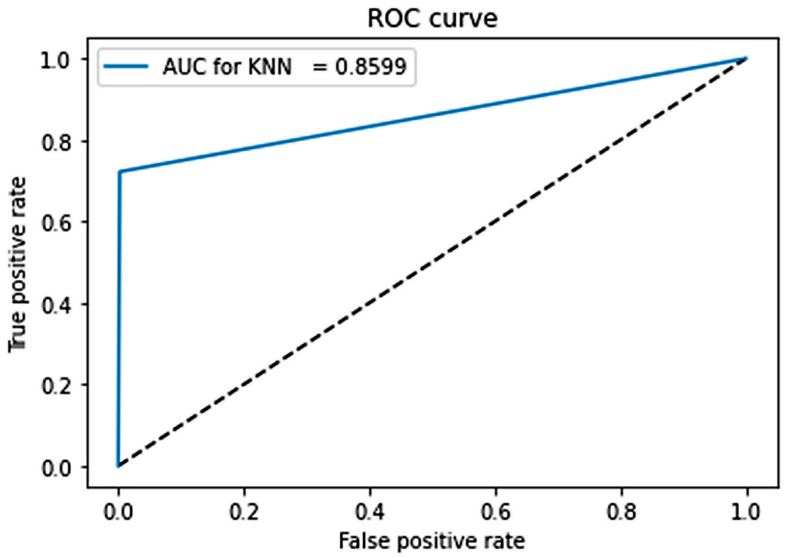

Рис. 3. График характеристики качества бинарного классификатора (ROC-AUC) для архитектуры алгоритма kNN

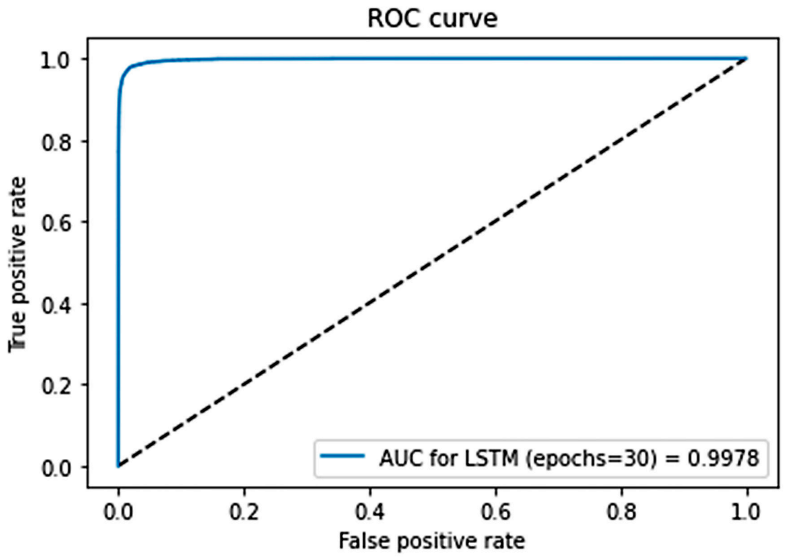

Рис. 4. График характеристики качества бинарного классификатора (ROC-AUC) для архитектуры LSTM

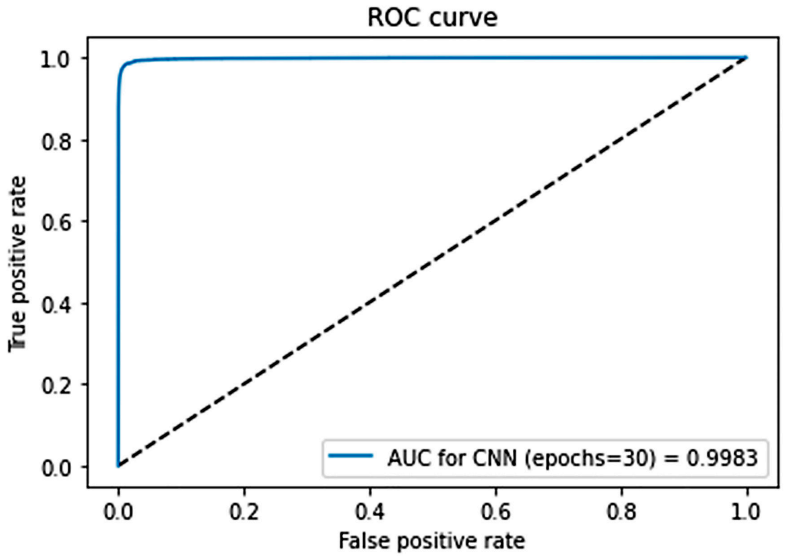

Рис. 5. График характеристики качества бинарного классификатора (ROC-AUC) для архитектуры CNN

Таблица 1

Результаты эксперимента замера метрик Accuracy, F_1-мера, ROC AUC

	Accuracy	F_1-мера (macro)	ROC AUC
kNN	0,7221	0,7749	0,8599
LSTM	0,8993	0,9301	0,9978
CNN	0,9351	0,9779	0,9983

Результаты расчетов (табл. 1) показали, что CNN превосходит остальные алгоритмы по всем трем метрикам: Accuracy составила 0,9351, F1-мера (macro) – 0,9779, ROC-AUC – 0,9983. Метод kNN и LSTM также показали хорошие результаты, но их показатели были ниже, чем у CNN.

Графики характеристик качества бинарного классификатора (ROC-AUC) для рассматриваемых алгоритмов представлены на рисунках 3, 4, 5.

Таким образом, CNN демонстрирует наибольшую эффективность среди рассмотренных методов классификации для идентификации пользователей по компьютерному почерку, опережая метод k-NN и LSTM, обеспечивая высокую точность классификации сложных паттернов поведения в рассматриваемом сценарии. А полученные значения метрик оценки качества классификации подтверждают потенциал использования динамических и статических характеристик компьютерного почерка пользователя как надежного средства идентификации, в том числе при проведении процедуры прокторинга.

Заключение

Технология идентификации по компьютерному почерку может значительно повысить эффективность и надежность процедуры прокторинга при проведении аттестации и проверки авторства работ за счет постоянного, но в то же время «ненавязчивого» процесса идентификации аттестуемого. Сильной стороной данного метода идентификации при использовании CNN является способность достаточно точно идентифицировать пользователей на основе уникальных характеристик ввода информации. Распознавание клавиатурного почерка не требует использования дорогостоящего специализированного оборудования (конечно, для обучения нейронных сетей высокопроизводительный сервер обработки данных желателен, но для обеспечения, например, видеопотока в режиме онлайн по множеству аттестуемых вложений в инфраструктуру потребуется гораздо больше), вследствие чего цена внедрения такой системы относительно невысока. Кроме того, мониторинг клавиатурного почерка можно производить непрерывно

и незаметно для пользователя, не отвлекая его внимание от проведения обучения или аттестации.

Тем не менее, внедрение такой модели в существующие системы прокторинга и онлайн-обучения сопряжено с рядом технических и организационных трудностей: необходимо организовать сбор данных, создать надежное хранилище данных с целью защиты персональных данных и конфиденциальности, собрать согласия пользователей на обработку их персональных данных и пр. Для успешного применения анализа компьютерного почерка потребуется адаптация существующих процедур обучения и аттестации (в том числе и фондов оценочных средств), чтобы обеспечить достаточность объема информации для обучения нейронных сетей.

Список литературы

1. Данилина Е. Ю. Метод k-ближайших соседей в задаче распознавания // Математическое и информационное моделирование. 2019. № 17. С. 80-84.

2. Malkov Y. A., Yashunin D. A. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs // IEEE transactions on pattern analysis and machine intelligence. 2018. Vol. 42. №. 4. P. 824-836.

3. Черный С. Н. Метод K-Nearest Neighbors. Разбор без использования бибилотек и с использованием бибилиотек. // Habr. [Электронный ресурс]. URL: https://habr.com/ru/articles/680004/ (дата обращения: 15.03.2024).

4. Шумская А.О. Оценка эффективности метрик расстояния Евклида и расстояния Махаланобиса в задачах идентификации происхождения текста // Доклады ТУСУР. 2013. № 3(29). С. 141-145.

5. Родионов А.В., Ищенко К.Л. Исследование влияния параметров алгоритма k-ближайших соседей на метрики качества моделей // Современные наукоемкие технологии. 2023. № 1. С. 37-43. DOI 10.17513/snt.39496.

6. Lipton Z. C., Berkowitz J., Elkan C. A Critical Review of Recurrent Neural Networks for Sequence Learning. // Arxiv.org [Электронный ресурс]. URL: https://arxiv.org/pdf/1506.00019 (дата обращения: 15.03.2024).

7. LSTM – сети долгой краткосрочной памяти. // Habr Wunder Fund [Электронный ресурс]. URL: https://habr.com/ru/companies/wunderfund/articles/331310/ (дата обращения: 12.03.2024).

8. Андреева А.Н., Лезин И.А. Использование комбинации сверточной сети и сети долго-краткосрочной памяти для небинарной классификации текстов // Вопросы устойчивого развития общества. 2022. № 6. С. 1081-1087.

9. Воробьев Е. В., Пучков Е. В. Классификация текстов с помощью сверточных нейронных сетей // Молодой исследователь Дона. 2017. № 6(9). URL: https://mid-journal.ru/upload/iblock/586/Vorobev-Puchkov-531.pdf (дата обращения: 11.03.2024).

10. Киллури К.С., Максион Р.А. Сравнение детекторов аномалий по динамике нажатия клавиш // 39-я ежегодная международная конференция по надежным системам и сетям (DSN-2009). 2009. С. 125-134.

11. Killourhy K., Maxion R. Keystroke Dynamics – Benchmark Data Set // Keystroke Dynamics [Электронный ресурс]. URL: https://www.cs.cmu.edu/~keystroke/ (дата обращения: 14.03.2024).

12. Оценка моделей многоклассовой классификации // Qlik Help [Электронный ресурс]. URL: https://help.qlik.com/ru-RU/cloud-services/Subsystems/Hub/Content/Sense_Hub/ AutoML/scoring-multiclass-classification.htm (дата обращения: 12.03.2024).

13. Оценка Micro и Macro F1 // Стивен Олрайт [Электронный ресурс]. URL: https://stephenallwright.com/micro-vs-macro-f1-score/ (дата обращения: 12.03.2024).

14. Показатель AUC // Стивен Олрайт [Электронный ресурс]. URL: https://stephenallwright.com/good-auc-score/ (дата обращения: 15.03.2024).