DOI 10.37882/2223-2966.2024.6-2.35

РАЗРАБОТКА СИСТЕМЫ ОБНАРУЖЕНИЯ СЕТЕВЫХ АТАК

DEVELOPMENT OF A NETWORK ATTACK DETECTION SYSTEM

A. Selin
V. Blinov
E. Amelyutin

Summary. The article is devoted to the development of a network attack detection system that uses a method combining a multilayer perceptron, gradient boosting and a KNN algorithm. The dataset used in the work includes various types of attacks, such as DoS (denial of service), R2L (unauthorized access to the local system), U2R (access enhancement) and Probe (system scan). The proposed solutions can be effectively used in financial institutions, government agencies, medical organizations and corporate networks, where it is critically important to maintain a high level of cybersecurity.

Keywords: intrusion detection system, information security, network attacks, traffic monitoring.

Селин Андрей Александрович
к. т. н., доцент,
МИРЭА — Российский технологический университет
chuknor@yandex.ru
Блинов Владимир Владимирович
МИРЭА — Российский технологический университет
vovah750@mail.ru
Амелютин Евгений Вячеславович
к. т. н., доцент,
МИРЭА — Российский технологический университет
amelyutin9@yandex.ru

Аннотация. Статья посвящена разработке системы обнаружения сетевых атак, которая использует метод, объединяющий многослойный персептрон, градиентный бустинг и KNN алгоритм. Набор данных, используемый в работе, включает в себя различные типы атак, такие как DoS (отказ в обслуживании), R2L (несанкционированный доступ к локальной системе), U2R (повышение уровня доступа) и Probe (сканирование системы). Предложенные решения могут быть эффективно использованы в финансовых учреждениях, государственных структурах, медицинских организациях и корпоративных сетях, где критически важно поддерживать высокий уровень кибербезопасности.

Ключевые слова: система обнаружения вторжений, информационная безопасность, сетевые атаки, мониторинг трафика.

В современном мире информационные технологии проникают во все сферы человеческой жизни, поэтому вопросы информационной безопасности становятся всё более актуальными и значимыми. С каждым днём количество и сложность сетевых атак увеличивается, что требует постоянного развития и совершенствования средств их обнаружения и предотвращения.

Системы обнаружения сетевых атак являются неотъемлемой частью комплексной стратегии защиты информационных систем, обеспечивая дополнительный уровень безопасности путем анализа трафика и поведенческих паттернов в сети для выявления подозрительных действий, которые могут свидетельствовать о попытках несанкционированного доступа, распространения вредоносного программного обеспечения, атак на отказ системы и других форм агрессии.

Сетевая атака — это действия с применением программных и технических средств с использованием сетевого протокола, направленные на реализацию угроз несанкционированного доступа к информации, воздействия на нее или на ресурсы автоматизированной информационной системы [1].

Данная работа сосредоточена только на классах и видах, которые описаны в наборе данных соревнования KDDCup'99 по Data Mining, проведенного в 1999 году. Качественная структурированность данных этого набора облегчает процесс их обработки. Также, преимуществом является отсутствие аномалий и коллизий, так как набор тщательно обновлялся в течение двадцати лет.

Каждое сетевое соединение помечается как нормальное или ненормальное (атака), при этом аномалии подразделяются на 4 категории, общее число видов атак составляет 39 [2]. Классификация наглядно представлена в таблице 1.

Методы обнаружения сетевых атак

Обнаружение сетевых атак — процесс распознавания и реагирования на вредоносную активность, направленную на сетевые инфраструктуры и информационные системы. Этот процесс включает мониторинг и анализ сетевого трафика, системных журналов и других источников данных для выявления признаков необычного или подозрительного поведения. Сегодня применяют три основных метода обнаружения сетевых атак: анализ по сигнатурам, выявление аномального поведения объекта и использование методов машинного обучения.

1. *Сигнатурный метод*. Данный метод использует функции в полезной нагрузке пакета, такие как ключевые слова, специфичные для приложения,

Таблица 1.

Классификация сетевых атак

Класс:	U2R	DoS	R2L	Probe
Подклассы:	— Buffer_overflow — Loadmodule — Perl — Rootkit	— Back — Land — Neptune — Pod — Smurf — Teardrop	— Ftp_write — Guess_passwd — Imap — Multihop — Phf — Spy — Warezclient — Warezmaster	— Ipsweep — Nmap — Portsweep — Satan

или шаблоны связи, которые повышают точность классификации пакетов.

Измерения в исследовании [3] показывают, что предлагаемый метод улучшает точность распознавания трафика, поскольку снижает количество неопределенного трафика на 11 % по сравнению с методом мониторинга портов. Данный метод также позволяет выявлять несколько типов трафика, которые были некорректно классифицированы или не были определены.

2. *Анализ аномалий*. Обнаружение аномалий направлено на выявление наличия аномальных паттернов в сетевом трафике. Однако, на сегодняшний день общепринятой процедуры для определения наличия паттернов в трафике не существует. Исследовательские предложения в области обнаружения аномалий обычно следуют четырехэтапному подходу, в котором первые три этапа определяют метод обнаружения, а последний посвящен проверке подхода. В начале процесса осуществляется сбор данных о трафике в сети. Далее, на следующем этапе, проводится анализ этих данных с целью извлечения наиболее важных характеристик. На третьей стадии трафик классифицируется как нормальный или аномальный. Затем, на заключительном этапе, эффективность метода проверяется путем тестирования на разнообразных аномалиях трафика [4].
3. Машинное обучение — класс методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи, а обучение за счёт применения решений множества сходных задач [5]. Постоянно эволюционирующий характер сетевых атак требует создания адаптивной и гибкой системы безопасности, способной анализировать обширные потоки сетевого трафика и приспосабливаться к изменениям сетевой активности.

Поскольку результаты работы моделей в исследованиях различных авторов представлены с помощью разнообразных показателей, то для проведения сравнительного анализа результатов работы различных моделей, необходимо разработать и обосновать единую методику оценивания.

В исследованиях *F*1-мера часто применяется как показатель качества классификации, в полной или частичной форме. Эта метрика особенно подходит для анализа эффективности классификатора на несбалансированном наборе данных, каким и является KDD '99, где преобладает число нормальных подключений по сравнению с соединениями, являющимися атаками. *F*1-мера вычисляется на основе двух параметров: Recall (полнота) и Precision (точность).

Параметр Recall, отражает долю векторов, корректно классифицированных в определенный класс, среди всех векторов, которые действительно относятся к этому классу. Параметр Precision отображает долю векторов, корректно классифицированных в определенный класс, среди всех векторов, которые были отнесены к этому классу, включая ошибочно классифицированные. Каждый из этих параметров представлен десятичным числом и зависит от статистических показателей классификатора на тестовом наборе данных, а именно:

— *TP* (*class* = *a*), количество векторов тестовой выборки, верно определенных в *a*-й класс;

— *TN* (*class* = *a*), количество векторов тестовой выборки, верно определенных в любой другой класс;

— *FP* (*class* = *a*), количество векторов тестовой выборки, ошибочно определенных в *a*-й класс;

— *FN* (*class* = *a*), количество векторов тестовой выборки, ошибочно определенных в любой другой класс.

Логичным подходом было бы применение показателя точности, основанного на упомянутых выше метках:

$$accuracy = \frac{TN + TP}{TN + TP + FP + FN}$$

*F*1-мера выражается следующим образом:

$$F1Score = \frac{2 * Presicion * Recall}{Precision + Recall}$$

Точность (Precision) и полнота (Recall) для *a*-го класса определяются как:

$$Precision(class = a) = \frac{TP(class = a)}{TP(class = a) + FP(class = a)}$$

$$Recall(class = a) = \frac{TP(class = a)}{TP(class = a) + FN(class = a)}$$

Таким образом, в случае бинарной классификации *F*1-меру необходимо рассчитать один раз, а при наличии множества классов — отдельно для каждого класса.

Выбор модели для обнаружения сетевых атак

В контексте бинарной классификации оценить точность обнаружения отдельных классов атак невозможно. Учитывая, что сравнение возможно только по итоговым показателям эффективности распознавания каждой модели, а также из-за количества исследовательских данных по разным моделям, в этой работе целесообразно сосредоточиться на сравнении наилучших результатов моделей. Результаты сравнения точности моделей представлены в таблице 2.

Таблица 2.

Сравнение результатов точности обнаружения сетевых атак при бинарной классификации

№	Модель	Точность распознавания, %
1	Совмещенная (MLP+SOM+SVM)	99,85
2	MLP	99,8
3	LGBM	99,74
4	CatBoost	99,71
5	ETC	99,7
6	RF	99,69
7	KNN	99,36
8	XGBoost	99,22
9	SOM	99
10	Auto-encoder	94,71
11	MLP, обученная с помощью алгоритма Левенберга-Марквардта	94
12	MLP, обученная с помощью квазиньютоновского метода	92
13	Рекуррентная	91,58

По результатам анализа методов распознавания сетевых вторжений при бинарной классификации можно сделать следующие заключения:

— многослойный персептрон является наиболее эффективным при индивидуальном применении и способен обучаться исключительно на не вредоносном трафике;
— методы, использующие градиентный бустинг и случайный лес, демонстрируют результаты схожие с многослойным персептроном;
— гибридные системы показывают более высокие результаты точности по сравнению с монолитными моделями.

Разработка системы обнаружения сетевых атак

Для первой экспериментальной модели был выбран многослойный персептрон, включающий два скрытых слоя, первый содержит 50 нейронов, а второй — 10. На всех слоях, кроме выходного, применяется функция активации ReLu [7]. На выходном слое нейронной сети используется функция Softmax, формирующая матрицу с размерами $n*m$, где n — количество векторов в выборке, а m — число классов. Для каждого вектора столбцы матрицы заполняются вероятностями принадлежности к каждому классу и выбирается наибольшая из них.

Поскольку при обучении модель стремится к минимизации функции, целесообразно использовать оптимизатор, который снижает потери, изменяя атрибуты нейронной сети, такие как вес и смещение [8]. В этой модели был применен стандартный оптимизатор «adam». Функционал библиотеки keras [9] позволяет в режиме реального времени отслеживать значения отдельных метрик после каждой эпохи обучения, что помогает анализировать процесс обучения и определить момент, после которого скорость обучения модели заметно снижается.

Вторая модель основана на алгоритме градиентного бустинга над решающими деревьями. LGBM-алгоритм эффективно справляется с поиском нелинейных зависимостей в неоднородных данных, что делает его подходящим для данной задачи. Принцип работы алгоритма заключается в последовательном создании решающих деревьев, при этом с каждым новым деревом значение функции потерь снижается. Когда улучшение функции потерь прекращается на протяжении заданного количества эпох, модель завершает обучение.

Третья модель основана алгоритме K-ближайших соседей (KNN), принцип работы заключается в присвоении объекта к классу, метка которого является наиболее распространенной среди ближайших к данному объекту соседей.

После подготовки данных, были созданы функции для обучения и тестирования каждой модели. Для моделей KNN и LGBM были использованы встроенные библиотечные методы, а для многослойного персептрона данные функции были реализованы вручную.

В ходе определения оптимального количества эпох обучения было выявлено, что после пятой эпохи значение функции потерь «loss» больше не снижалось. Из этого следует, что такого количества эпох достаточно для полного обучения модели.

Прогнозы, полученные от каждого из трех алгоритмов, используются в качестве входных данных для модели логистической регрессии. На последнем этапе результаты разделяются на две части: первая содержит индивидуальные результаты каждого алгоритма, а вторая включает в себя результаты, полученные с помощью логистической регрессии (рисунок 1).

На рисунке 2 представлена система обнаружения сетевых атак с центральным ядром, которое координиру-

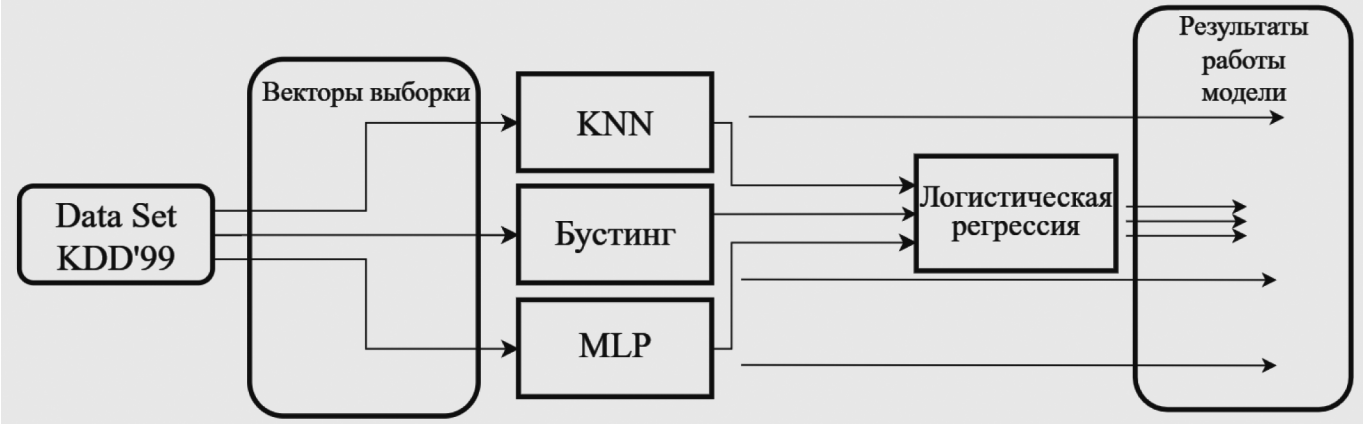

Рис. 1. Блок-схема описания метода работы системы

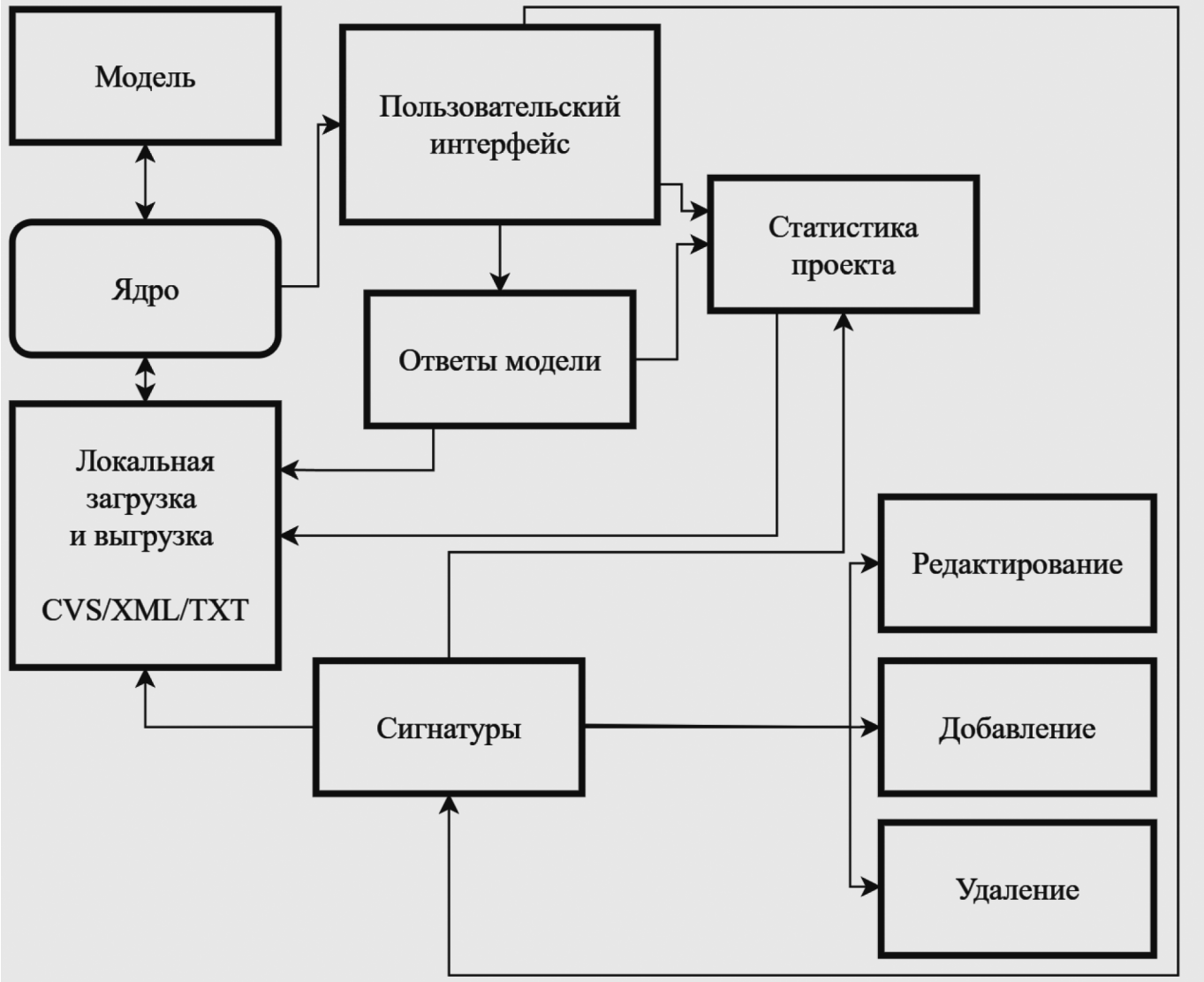

Рис. 2. Структурная схема разработанной системы

ет обработку и передачу данных между компонентами системы. Модель, в свою очередь, анализирует данные для выявления потенциальных атак. Для удобства работы с данными предусмотрена возможность локальной загрузки и выгрузки в форматах CVS, XML и TXT, что позволяет вводить новую информацию в систему. Особое место в системе занимают сигнатуры — ключевые слова, соответствующие известным атакам. Посредством функций редактирования, добавления и удаления сигнатур, пользователем формируется список, который

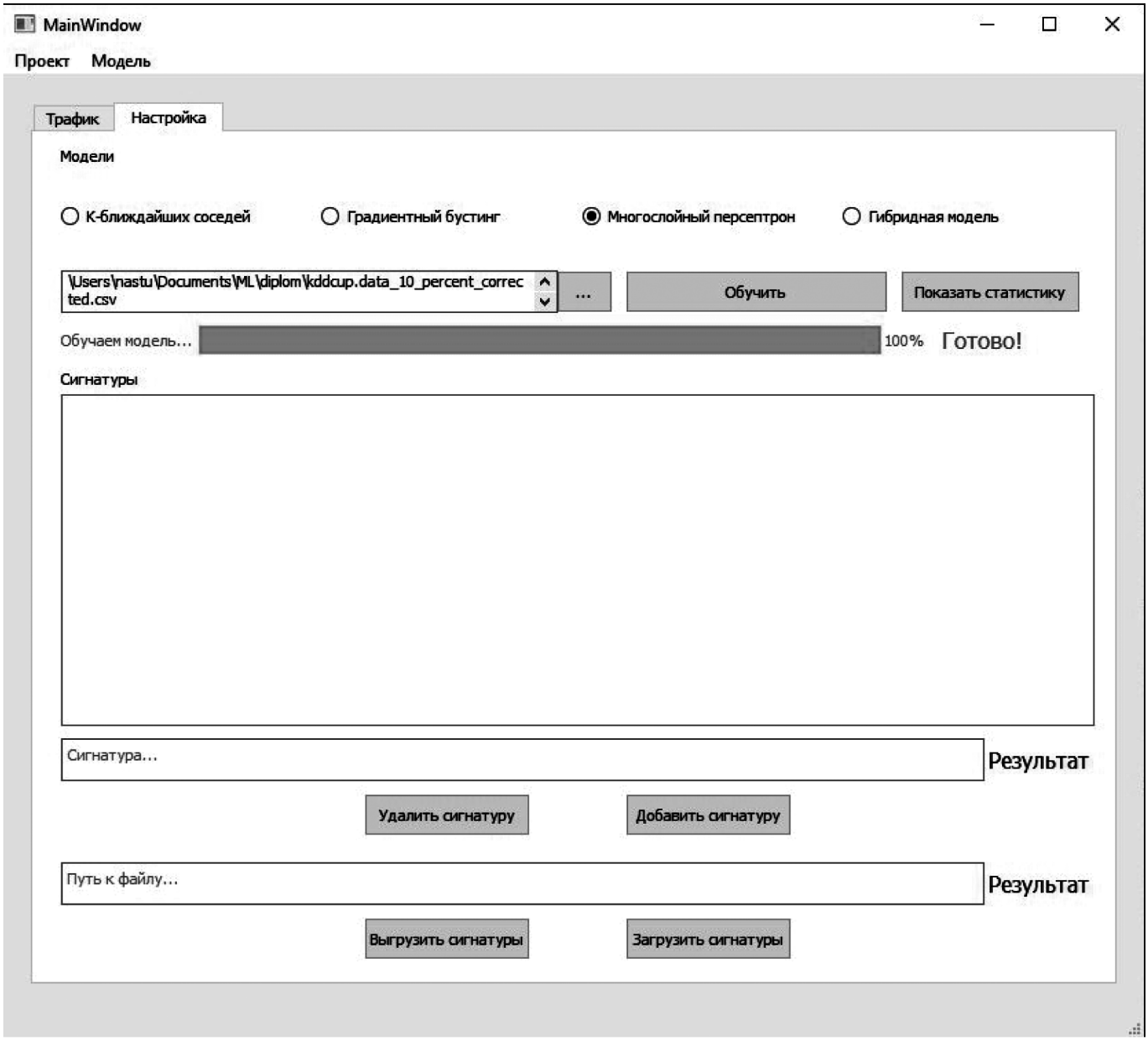

Рис. 3. Пользовательский интерфейс системы

используется системой для выявления атак, что делает систему адаптивной к меняющейся среде угроз. Взаимодействие пользователя с системой происходит через пользовательский интерфейс, который служит платформой для выбора модели и ее отображений. Также через интерфейс предоставляется доступ к статистике проекта с возможностью просмотра метрик, связанных с производительностью системы (рисунок 3).

Работа системы начинается с загрузки набора данных KDD Cup'99 в формате CVS или TXT. После загрузки выполняется проверка на корректность загруженных данных. В случае обнаружения ошибок в данных необходимо повторно вернуться к их загрузке. Затем следует этап добавления сигнатур в систему, который может осуществляться как через командную строку, так и путем загрузки файлов в форматах CVS, XML и TXT. После добавления пользователь проверяет полноту внесенных сигнатур. Если добавлены не все сигнатуры, необходимо вернуться на предыдущий этап. Далее осуществляется проверка на необходимость удаления существующих сигнатур. В случае, если требуется удаление сигнатур, используется функция «удалить сигнатуру». После чего необходимо выбрать модель для обучения: К-ближайших соседей, градиентный бустинг и многослойный персептрон, или гибридная модель, состоящая из двух или трех моделей. После выбора модели следует перейти к непосредственному обучению выбранной модели. По окончании обучения проводится анализ статистических данных. Если анализ данных выполнен некорректно, необходи-

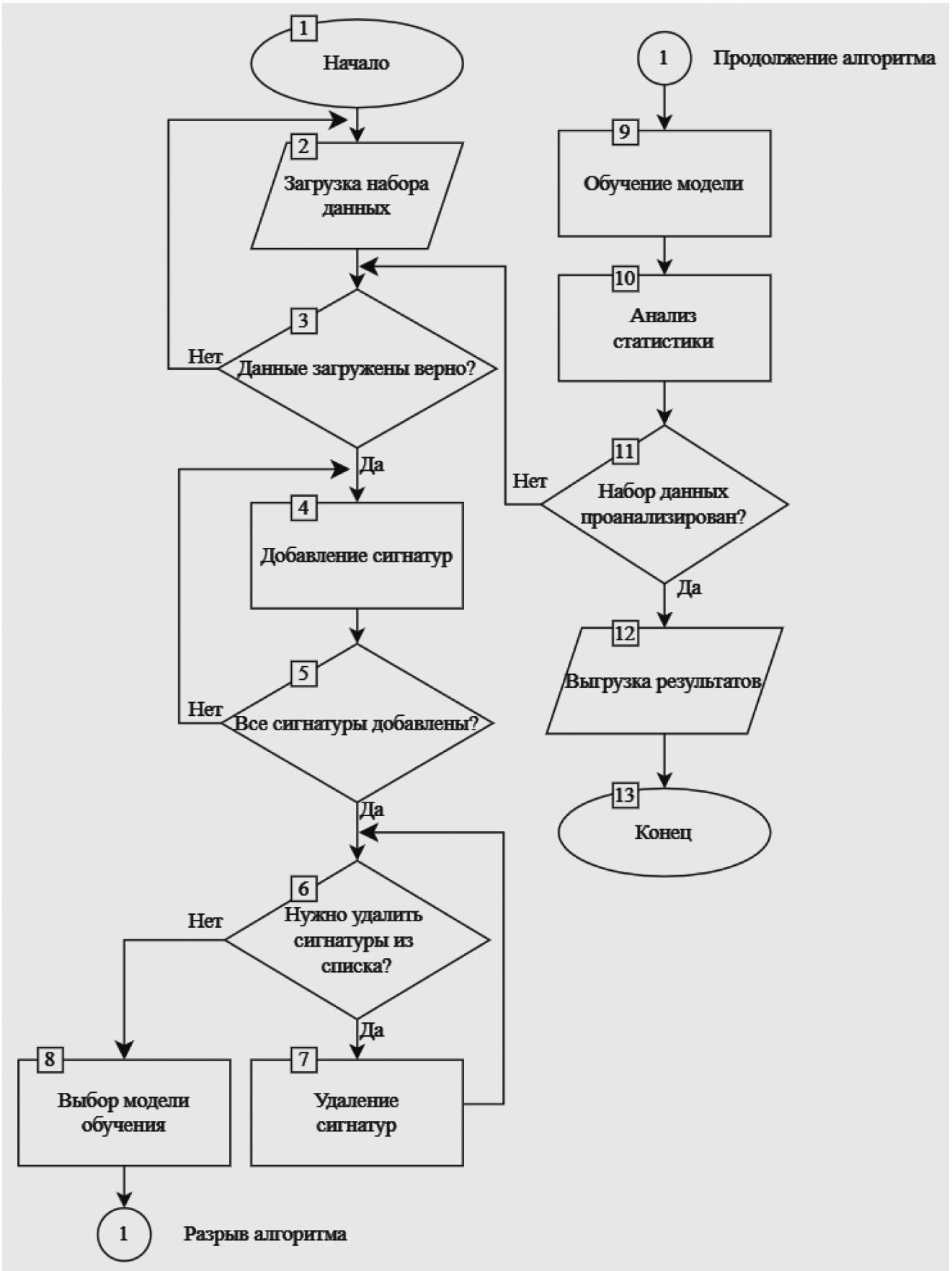

Рис. 4. Блок-схема разработанной системы

мо вернуться к этапу проверки загруженных данных. При завершении анализа данных результаты экспортируются в форматы CVS, XML или TXT, после чего алгоритм завершает свою работу. Общий процесс работы системы представлен на рисунке 4.

Демонстрация результатов работы системы

Исследования точности классификации и взвешенной *F*1-меры были проведены в семи случаях: для каждой модели по отдельности, для комбинации KNN+MLP+LGBM, а также для комбинаций KNN+MLP, KNN+LGBM и MLP+LGBM. Результаты более подробно отображены в таблице 3.

Таблица 3.

Результаты классификации моделей

	KNN	MLP	LGBM	KNN + MLP	KNN + LGBM	MLP + LGBM	KNN + MLP + LGBM
Точность, %	99,92	99,39	89,55	99,90	99,90	99,01	99,94
*F*1-мера, %	99,92	98,89	86,95	99,91	99,92	99,37	99,92

Заключение

Проведенный анализ научных работ предметной области позволил сделать выводы о том, какие методы обнаружения сетевых атак наиболее эффективны в различных условиях. Особенно высокую эффективность продемонстрировали классические методы, такие как многослойный персептрон и градиентный бустинг, а также гибридные модели, основанные на этих подходах. Основной трудностью в таком анализе является отсутствие единой системы оценивания эффективности таких систем и стандартов представления результатов, а также ограниченная база исследований из-за относительной новизны данной области.

На основе этих выводов была создана гибридная модель обнаружения сетевых атак, включающая модели MLP, LGBM и алгоритм KNN. Конечный результат формируется путем усреднения выходных данных из каждой модели. Однако, успешность модели в одних условиях не гарантирует аналогичных результатов при изменении условий или тестовых данных. Тем не менее, реализация такой модели подтверждает преимущества гибридных моделей перед монолитными, что помогает в практическом выборе оптимальной модели для конкретных задач.

ЛИТЕРАТУРА

1. ГОСТ Р 53114-2008 Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения: национальный стандарт Российской Федерации: дата введения 2009-10-01/ — Изд. официальное. —11 с.
2. Russianblogs. Обнаружение вторжений и распознавание на основе машинного обучения. [Электронный ресурс] — URL: https://russianblogs.com/article/7850855685/#KDD_CUP_67 (дата обращения 23.12.2023)
3. Y. Choi «On the Accuracy of Signature-based Traffic Identification Technique in IP Networks» Conf. Broad. Converg. Networks, June 2007.
4. F. Simmross-Wattenberg, J. Asensio-Perez, P. Casaseca-Higuera and M. Martin-Fernandez, «Anomaly Detection in Network Traffic Based on Statistical Inference and alpha-Stable Modeling» IEEE Trans. Dependable and Secure Computing, vol. 8, no. 4, pp. 494–509, July 2011.
5. Википедия. Машинное обучение. [Электронный ресурс] — URL: https://ru.wikipedia.org/wiki/Машинное_обучение (дата обращения 25.02.2024)
6. Corpsoft24. IDS и ее особенности применения. [Электронный ресурс] — URL: https://www.corpsoft24.ru/about/blog/chto-takoe-ids-i-osobennosti-eye-primeneniya/ (дата обращения 22.04.2024)
7. Habr. Функция активации ReLu. [Электронный ресурс] — URL: https://habr.com/en/articles/348000/ (дата обращения 28.04.2024)
8. Оптимизация в ML. [Электронный ресурс] — URL: https://academy.yandex.ru/handbook/ml/article/optimizaciya-v-ml (дата обращения 30.04.2024)
9. Keras: Deep Learning for humans. [Электронный ресурс] — URL: https://keras.io/ (дата обращения 30.04.2024)

© Селин Андрей Александрович (chuknor@yandex.ru); Блинов Владимир Владимирович (vovah750@mail.ru);
Амелютин Евгений Вячеславович (amelyutin9@yandex.ru)
Журнал «Современная наука: актуальные проблемы теории и практики»