Применение методов кластеризации для автоматизации формирования пользовательских ролей

И.В. Аникин[1], А.В. Агапятова[1], А.С. Катасёв[1,2]

[1]Казанский национальный исследовательский технический университет им. А.Н. Туполева-КАИ, Казань
[2]Казанский государственный энергетический университет, Казань

Аннотация: В статье решается задача автоматизированного формирования пользовательских ролей с применением методов машинного обучения. Для решения задачи используются методы кластерного анализа данных, реализованные на языке Python в среде разработки Google Colab. На основе полученных результатов разработана и апробирована методика формирования пользовательских ролей, позволяющая сократить время формирования ролевой модели управления доступом.
Ключевые слова: машинное обучение, ролевая модель управления доступом, кластеризация, метод k-средних, иерархическая кластеризация, метод DBSCAN.

Введение

Бурное развитие информационных технологий ведет к росту количества новых угроз информационной безопасности и реализуемых кибератак на информационные системы компаний [1, 2]. Все более важным для организаций становится повышение уровня защищенности хранимой и обрабатываемой информации [3], при этом одной из эффективных мер защиты информации является реализация механизмов корректного управления доступом в соответствии с ролевой моделью [4].

До внедрения такой системы необходимо определиться с требуемым количеством пользовательских ролей и соответствующими правами доступа, исходя из функциональных задач, выполняемых ролью. Слишком большое количество ролей может усложнить систему, а слишком малое – привести к излишним привилегиям некоторых пользователей. Формирование ролевой модели является сложной задачей для организации, в связи с чем появляется необходимость применения средств автоматизации. В данной статье для формирования пользовательских ролей и их профилей доступа применен кластерный анализ [5].

Базовые сведения

Для решения поставленной задачи автоматизации формирования пользовательских ролей были проанализированы существующие модели управления доступом [6], системы управления доступом [7], методы машинного обучения [8 – 11] и кластерного анализа [12]. Модель управления доступом – важная часть защищенной компьютерной системы. Она позволяет разграничить доступ пользователей к ресурсам, исходя из их должностных обязанностей, и предотвратить угрозы информационной безопасности, в частности, несанкционированный доступ [13].

Ролевая модель управления доступом более приближена к реальной организационной структуре организации и реальным бизнес-процессам, а роли позволяют отобразить разделение функций различных должностей. При большом количестве пользователей данная модель позволяет уменьшить вероятность некорректного назначения прав доступа.

Для формирования ролей пользователей в рамках выбранной модели управления доступом может быть применен метод кластерного анализа [14]. Полученные кластеры будут являться ролями пользователей.

Исследование методов кластеризации

Для исследования методов кластеризации применительно к решению поставленной задачи выбран язык Python. С помощью пакетов SciPy, Pandas, Matplotlib, Sklearn реализованы методы иерархической кластеризации, k-средних и DBSCAN. Для разработки и тестирования программ кластеризации создан небольшой файл тестовых данных формата .csv, в котором права доступа закодированы в восьмеричном виде в соответствии с таблицей 1.

Таблица № 1

Кодирование прав доступа

Маска	Бинарное представление	Восьмеричное представление	Описание
- - -	000	0	Отсутствие прав
- - x	001	1	Права на выполнение
- w -	010	2	Права на запись
- w x	011	3	Права на запись и выполнение
r - -	100	4	Права на чтение
r - x	101	5	Права на чтение и выполнение
r w -	110	6	Права на чтение и запись
r w x	111	7	Полные права

В файл данных включены следующие должности (post): администратор (admin), инженер1 (engineer1), инженер2 (engineer2), главный инженер (main_eng), сотрудник удаленного доступа1 (external_user1), сотрудник удаленного доступа2 (external_user2), руководитель (head), заместитель руководителя (deputy_head). Должностям предоставлены права доступа, приведенные в таблице 2 в формате маски.

Таблица № 2

Права доступа, присвоенные каждой должности

post	file1	file2	file3	file4	file5
admin	rwx	rwx	rwx	rwx	rwx
engineer1	r	r	rwx	r	-
engineer2	r	r	rw	rwx	-
main_eng	r	r	rwx	rwx	-
external_user1	rw	rw	-	-	-
external_user2	r	r	-	-	-
head	rwx	rwx	rwx	rwx	r
deputy_head	rw	rw	rw	rw	r

Результат иерархической кластеризации представлен на рис. 1.

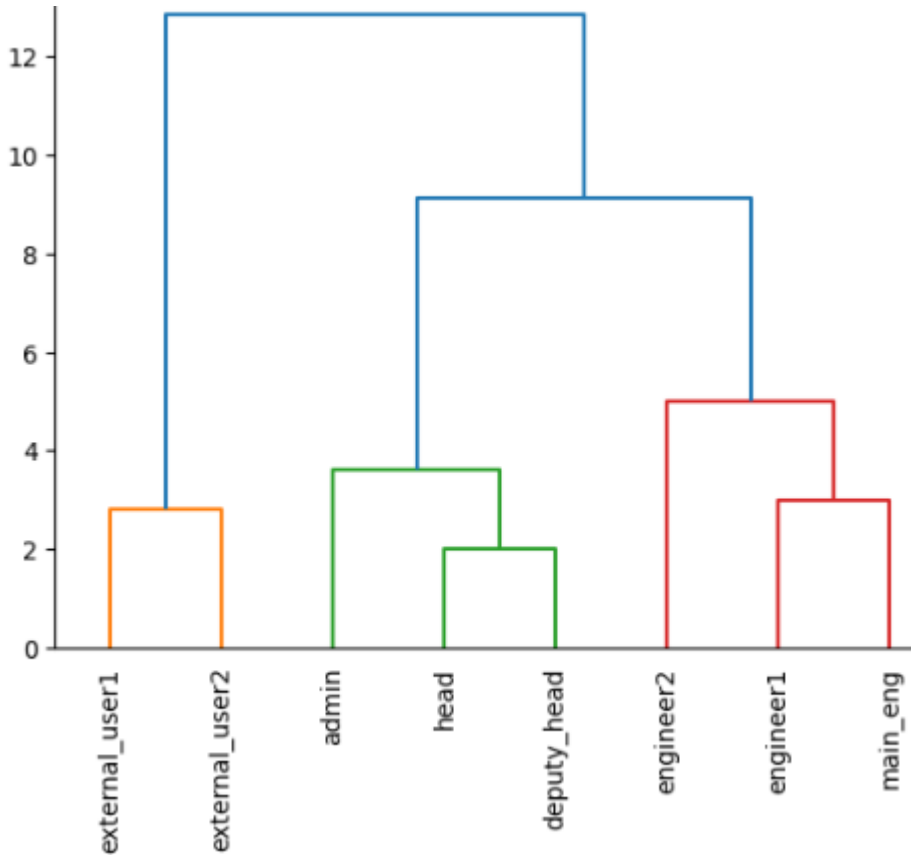

Рисунок 1 – Результат иерархической кластеризации

Выделены следующие кластеры:

1) сотрудник удаленного доступа1, сотрудник удаленного доступа2;

2) администратор, руководитель, заместитель руководителя;

3) инженер2, инженер1, главный инженер.

Для кластеризации методом k-средних проведено масштабирование каждой переменной фрейма данных до среднего значения «0» и стандартного отклонения «1», чтобы каждая переменная имела одинаковую важность при подборе алгоритма k-средних. В противном случае переменные с самыми широкими диапазонами оказывали бы слишком большое влияние. Для выполнения кластеризации k-means в Python использована функция KMeans из модуля sklearn. Количество кластеров не было известно заранее. Для его определения использован «метод колена», результаты работы которого представлены на рис. 2.

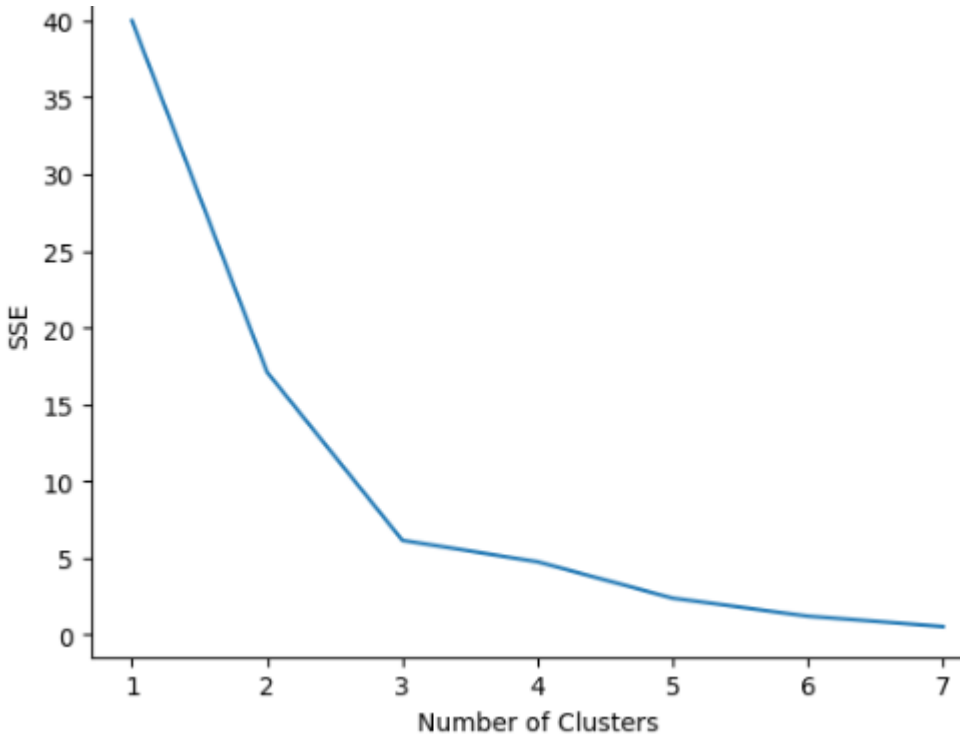

Рисунок 2 – Зависимость суммы квадратов ошибок от числа кластеров

При количестве кластеров, равном трем, на графике присутствует «колено», соответственно, наилучшее количество кластеров равно трем.

Метод k-средних позволил выделить следующие кластеры:

1) администратор, руководитель, заместитель руководителя;

2) сотрудник удаленного доступа1, сотрудник удаленного доступа2;

3) инженер1, инженер2, главный инженер.

Таким образом, результат совпал с иерархической кластеризацией.

Для применения плотностного алгоритма пространственной кластеризации с присутствием шума (DBSCAN) так же, как и в методе k-средних, проведено масштабирование каждой переменной фрейма данных. На рис. 3 представлен результат работы алгоритма DBSCAN.

В данном случае кластеры сформировались следующим образом:

1) администратор, руководитель, заместитель руководителя;

2) сотрудник удаленного доступа1, сотрудник удаленного доступа2;

3) инженер1, инженер2, главный инженер.

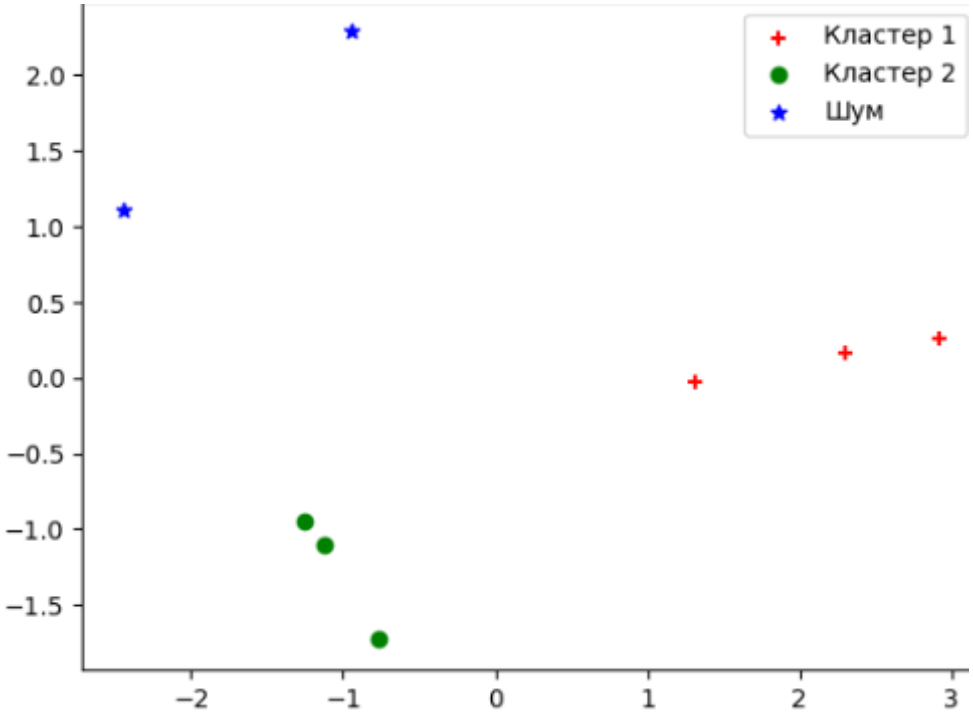

Рисунок 3. – Результат работы алгоритма DBSCAN

Полученный результат совпал с результатом иерархической кластеризации и кластеризации методом k-средних для трех кластеров.

Применение методов кластеризации привело к разделению исходных объектов на три кластера, однако наиболее эффективным оказался метод k-средних, при выполнении которого определено оптимальное количество кластеров.

Автоматизация формирования пользовательских ролей

Рассмотрим этапы разработанной методики автоматизированного формирования пользовательских ролей:

1) ввод исходных данных в программу для кластеризации;

2) анализ результатов кластеризации:

- нахождение оптимального количества кластеров и их ввод в программу с анализом полученного распределения по кластерам (в случае нескольких точек «перегиба» рассматриваются и анализируются оба случая);

- анализ распределения по кластерам методом иерархической кластеризации;

- анализ распределения по кластерам с помощью алгоритма DBSCAN;

3) выбор лучшего варианта кластеризации.

Анализ полученных ролей и корректировка распределения сотрудников состоит из следующих шагов:

1) составление сравнительных таблиц по кластерам;

2) выявление несовпадений в правах доступа для каждой таблицы;

3) корректировка прав доступа в каждой таблице для приведения их к единому виду.

Базовыми задачами современных систем управления доступом являются следующие: автоматизация, упрощение управления учетными записями и доступом к информационным системам внутри организации. В таких системах механизмы внесения изменений (создание, блокирование, удаление учетных записей, изменение прав доступа и т.д.) строятся на основе информации из доверенных источников о пользователях, их назначениях и рабочих статусах. Изменения могут вноситься автоматически или вручную.

Для формирования пользовательских ролей подготовлены исходные данные в виде таблицы прав доступа, содержащей следующие объекты:

- 15 должностей (системный администратор (sysadmin), сотрудник удаленного доступа 1 (external_user1), сотрудник удаленного доступа 2 (external_user2), сотрудник удаленного доступа 3 (external_user3), сотрудник удаленного доступа 4 (external_user4), секретарь (secretary), инженер 1 (engineer1), инженер 2 (engineer2), инженер 3 (engineer3), инженер 4 (engineer4), главный инженер 1 (main_engineer1), главный инженер 2 (main_engineer2), руководитель (head), заместитель руководителя (deputy_head), офицер безопасности (security_officer));

- 15 объектов (каталог проекта 1 на файловом сервере (Katalog1), каталог проекта 2 на файловом сервере (Katalog2), каталог проекта 3 на файловом сервере (Katalog3), каталог проекта 4 на файловом сервере (Katalog4), каталог проекта 5 на файловом сервере (Katalog5), рабочий каталог на файловом сервере (Work_katalog), каталог внутренних документов (Katalog_docs), база данных клиентов (DB_clients), база данных продаж (DB_sales), служба каталогов Active Directory (AD), файловый сервер (FS), почтовый сервер (PS), каталог отчетных документов (Katalog_reports), база данных сотрудников (DB_staff), общие ресурсы (Shared_resources));

- права доступа (чтение (r), запись (w), выполнение (x)) сотрудников соответствующих должностей.

Иерархия должностей представлена на рис. 4.

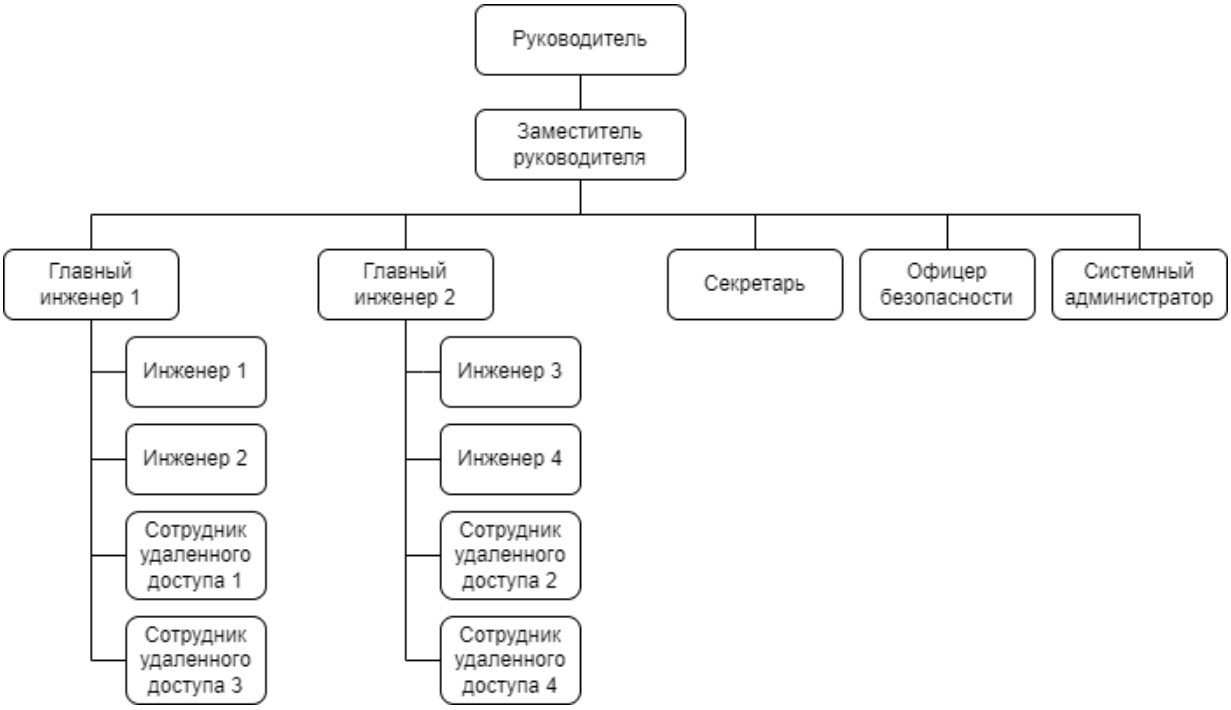

Рисунок 4. – Иерархия должностей

Составлена матрица доступа, в которой права доступа закодированы в восьмеричную систему и переведены в формат .csv.

Метод k-средних распределил должности по трем кластерам следующим образом:

1) сотрудник удаленного доступа 1, сотрудник удаленного доступа 2, сотрудник удаленного доступа 3, сотрудник удаленного доступа 4, секретарь, инженер 3, инженер 4;

2) системный администратор, руководитель, заместитель руководителя, офицер безопасности.

3) инженер 1, инженер 2, главный инженер 1, главный инженер 2.

Однако, поскольку, при трех кластерах значение суммы квадратов ошибок было высоким, были рассмотрены также другие два метода кластеризации. Результат DBSCAN-кластеризации для трех кластеров получился неподходящим для данного случая, поскольку большинство должностей программа распределила как «шум».

По результатам иерархической кластеризации должности разделены на 3 кластера следующим образом:

1) секретарь, сотрудник удаленного доступа 1, сотрудник удаленного доступа 2, сотрудник удаленного доступа 3, сотрудник удаленного доступа 4;

2) руководитель, заместитель руководителя, системный администратор, офицер безопасности;

3) инженер 3, инженер 4, инженер 1, инженер 2, главный инженер 1, главный инженер 2.

Более низкий уровень дендрограммы сформировал восемь кластеров:

1) секретарь;

2) сотрудник удаленного доступа 1, сотрудник удаленного доступа 3;

3) сотрудник удаленного доступа 2, сотрудник удаленного доступа 4;

4) руководитель, заместитель руководителя;

5) системный администратор, офицер безопасности;

6) инженер 3, инженер 4;

7) инженер 1, инженер 2;

8) главный инженер 1, главный инженер 2.

Распределение должностей по восьми кластерам в большей степени соответствует должностным обязанностям и матрице доступа, чем распределение по трем кластерам.

Применение алгоритма DBSCAN для восьми кластеров показало распределение должностей на пять кластеров и шум, который следует рассматривать как отдельный кластер. Распределение должностей методом k-средних выделило те же 8 кластеров.

Далее по методике составлены сравнительные таблицы по кластерам, выявлены и подкорректированы несовпадения в правах доступа. По результатам корректировок получено 10 итоговых кластеров (ролей), указанных в таблице 3.

Таблица № 3

Права доступа, присвоенные каждой должности

№	Кластер (роль)	Должности
1	Сотрудники удаленного доступа группы 1	Сотрудник удаленного доступа 1, сотрудник удаленного доступа 3
2	Сотрудники удаленного доступа группы 2	Сотрудник удаленного доступа 2, сотрудник удаленного доступа 4
3	Руководитель	Руководитель
4	Заместитель руководителя	Заместитель руководителя
5	Системный администратор	Системный администратор
6	Офицер безопасности	Офицер безопасности
7	Инженеры группы 2	Инженер 3, инженер 4
8	Инженеры группы 1	Инженер 1, инженер 2
9	Главные инженеры	Главный инженер 1, главный инженер 2
10	Секретарь	Секретарь

Выводы

Эффективность работы системы управления доступом во многом зависит от качества формирования ролевой модели. Однако, при большом количестве пользователей системы составление ролей вручную и оптимальное распределение пользователей может быть слишком затруднительно. Применение методики на практике показало хороший результат формирования пользовательских ролей: из пятнадцати должностей успешно сформированы десять пользовательских ролей на основе десяти кластеров. Полученные кластеры разделили должности по ролям согласно их должностным обязанностям, что говорит о правильности работы алгоритма.

Проведенные исследования показали, что при кластеризации на первом этапе необходимо выбрать оптимальное количество кластеров методом «колена», затем построить количество кластеров по методу k-средних, далее использовать иерархическую кластеризацию и кластеризацию на основе алгоритма DBSCAN. При необходимости можно рассмотреть детальные уровни дендрограммы, полученной в результате иерархической кластеризации, и выполнить кластеризацию методами k-средних и DBSCAN для получения большего количества кластеров. Алгоритм DBSCAN в реализации на Python является наименее удобным из трех представленных, поскольку для каждого случая необходимо переписывать программу для нужного количества кластеров и подбирать параметры алгоритма.

Из полученных разными методами результатов кластеризации выбирается наиболее подходящий согласно разделению должностных обязанностей. Так как все должности имеют различия в правах доступа, требуется корректировка. Однако она занимает значительно меньшее время, чем разделение большого количества пользователей по ролям вручную. Поэтому данная методика может быть применена для сокращения времени при распределении по ролям большого количества пользователей системы.

Литература

1. Бойко А.А. Боевая эффективность кибератак: практические аспекты // Системы управления, связи и безопасности. 2020. № 4. С. 134-162.

2. Abdullaev E.A.O. Cyber-attacks and their impact on the digital economy // International Journal of Humanities and Natural Sciences. 2024. No. 3-2 (90). P. 146-149.

3. Аникин И.В. Нечеткая оценка факторов риска информационной безопасности // Безопасность информационных технологий. 2016. Т. 23. № 1. С. 78-87.

4. Легеня П.Б. Защита от кибератак на основе контроля доступа к данным // Банковское дело. 2019. № 2. С. 65-71.

5. Будникова И.К., Плетенева Е.В. Кластерный анализ как функция интеллектуального анализа данных // Информационные технологии в строительных, социальных и экономических системах. 2022. № 1 (27). С. 25-28.

6. Oleynik P.P., Salibekyan S.M. Model of security for object-oriented and object-attributed applications // Proceedings of the Institute for System Programming of the RAS. 2016. Vol. 28. No. 3. P. 35-50.

7. Козлов А.Е. Система контроля и управления доступом на предприятие: понятие, характеристика и основные требования // Вестник Воронежского государственного технического университета. 2019. Т. 15. № 1. С. 42-47.

8. Сабиров А.И., Катасёв А.С., Дагаева М.В. Нейросетевая модель распознавания знаков дорожного движения в интеллектуальных транспортных системах // Компьютерные исследования и моделирование. 2021. Т. 13. № 2. С. 429-435.

9. Воробьёва Ю.Н., Катасёва Д.В., Катасёв А.С., Кирпичников А.П. Нейросетевая модель выявления DDOS-атак // Вестник Технологического университета. 2018. Т. 21. № 2. С. 94-98.

10. Майорова Е.С., Зарипова Р.С. Разработка алгоритма переноса стиля изображения с использованием предобученной нейросети // Инженерный вестник Дона, 2024, № 2. URL ivdon.ru/ru/magazine/archive/n2y2024/8997

11. Питкевич П.И. Методы объединения, сокращения размеров и обработка больших данных // Инженерный вестник Дона, 2021, № 12. URL ivdon.ru/ru/magazine/archive/n12y2021/7338

12. Аникин И.В., Газимов Р.М. Защищенный протокол кластеризации DBSCAN для вертикально распределённых данных // Информация и безопасность. 2016. Т. 19. № 4. С. 515-518.

13. Врембяк А.А., Шарыпова Т.Н. Методы несанкционированного доступа к информации // Инновации. Наука. Образование. 2021. № 32. С. 366-369.

14. Прохоренков П.А., Регер Т.В., Гудкова Н.В. Методы кластерного анализа в региональных исследованиях // Фундаментальные исследования. 2022. № 3. С. 100-106.

References

1. Bojko A.A. Sistemy upravleniya, svyazi i bezopasnosti. 2020. № 4. pp. 134-162.

2. Abdullaev E.A.O. International Journal of Humanities and Natural Sciences. 2024. № 3-2 (90). pp. 146-149.

3. Anikin I.V. Bezopasnost' informacionnyh tekhnologij. 2016. Vol. 23. №1. pp. 78-87.

4. Legenya P.B. Bankovskoe delo. 2019. №2. pp. 65-71.

5. Budnikova I.K., Pleteneva E.V. Informacionnye tekhnologii v stroitel'nyh, social'nyh i ekonomicheskih sistemah. 2022. №1 (27). pp. 25-28.

6. Oleynik P.P., Salibekyan S.M. Proceedings of the Institute for System Programming of the RAS. 2016. Vol. 28. № 3. pp. 35-50.

7. Kozlov A.E. Vestnik Voronezhskogo gosudarstvennogo tekhnicheskogo universiteta. 2019. Vol. 15. № 1. pp. 42-47.

8. Sabirov A.I., Katasev A.S., Dagaeva M.V. Komp'yuternye issledovaniya i modelirovanie. 2021. Vol. 13. № 2. pp. 429-435.

9. Vorob'eva YU.N., Kataseva D.V., Katasev A.S., Kirpichnikov A.P. Vestnik Tekhnologicheskogo universiteta. 2018. Vol. 21. № 2. pp. 94-98.

10. Majorova E.S., Zaripova R.S. Inzhenernyj vestnik Dona, 2024, № 2. URL ivdon.ru/ru/magazine/archive/n2y2024/8997

11. Pitkevich P.I. Inzhenernyj vestnik Dona, 2021, № 12. URL ivdon.ru/ru/magazine/archive/n12y2021/7338

12. Anikin I.V., Gazimov R.M. Informaciya i bezopasnost'. 2016. Vol. 19. № 4. pp. 515-518.

13. Vrembyak A.A., SHarypova T.N. Innovacii. Nauka. Obrazovanie. 2021. № 32. pp. 366-369.

14. Prohorenkov P.A., Reger T.V., Gudkova N.V. Fundamental'nye issledovaniya. 2022. № 3. pp. 100-106.

Дата поступления: 28.07.2024

Дата публикации: 6.09.2024