Янул А.Д. Определение этапов предварительной обработки текста и выбор алгоритма кластеризации сообщений журнальных файлов сервера// Международный журнал информационных технологий и энергоэффективности. – 2022. –
Т. 7 № 3 (25) часть 2 с. 8–15

Международный журнал информационных технологий и энергоэффективности

Сайт журнала:
http://www.openaccessscience.ru/index.php/ijcse/

УДК 004.9

ОПРЕДЕЛЕНИЕ ЭТАПОВ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ТЕКСТА И ВЫБОР АЛГОРИТМА КЛАСТЕРИЗАЦИИ СООБЩЕНИЙ ЖУРНАЛЬНЫХ ФАЙЛОВ СЕРВЕРА

Янул А.Д.
Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А.Бонч-Бруевича, Россия (193232, г. Санкт-Петербург, ул. пр. Большевиков, 22, к. 1), e-mail: adyanul@mail.ru

В статье приводится обоснование разработки собственного российского ПО, представлено приложение для кластеризации сообщений журнальных файлов сервера и описан эксперимент по определению этапов предварительной обработки текста для процесса кластеризации. Также приводится обоснование основного инструментария для выполнения данной задачи (библиотеки scikit-learn (Python)) и выбор алгоритмов кластеризации (DBSCAN и BIRCH). По ходу статьи на графиках приведены зависимости качества кластеризации (в виде коэффициента Силуэта) и времени выполнения процесса от выбранных этапов предварительной обработки. В заключении дается анализ полученных в ходе эксперимента зависимостей.

Ключевые слова: scikit-learn, кластеризация текста, анализ журнальных файлов, DBSCAN, BIRCH.

DEFINING THE STAGES OF PREPROCESSING TEXT AND SELECTING AN ALGORITHM FOR CLUSTERING THE MESSAGES OF THE SERVER LOG FILES

Yanul A.D.
The Bonch-Bruevich Saint Petersburg State University of Telecommunications, Russia (2193232, St. Petersburg, Bolshevikov Ave., 22, building 1), e-mail: adyanul@mail.ru

The paper provides a rationale for the development of proprietary Russian software, presents an application for clustering messages of server log files, and describes an experiment to determine the stages of pre-processing text for the clustering process. The rationale for the main toolkit for this task (scikit-learn (Python) library) and the choice of clustering algorithms (DBSCAN and BIRCH) are also given. In the course of the paper, the graphs show the dependencies of clustering quality (in the form of the Silhouette coefficient) and process execution time on the selected preprocessing steps.In the conclusion, the analysis of the dependencies obtained during the experiment is given.

Keywords: scikit-learn, text clustering, log file analysis, DBSCAN, BIRCH.

Ограничение для России доступа к иностранным информационным продуктам и технологиям, наблюдаемое в настоящее время, служит дополнительным фактором, обуславливающим необходимость разработки собственного программного обеспечения (ПО) [1].

Проблемы в области информационных технологий (ИТ) в России наблюдались и ранее, что подтверждают некоторые нормативные документы [2,3]. Это такие проблемы, как сильная зависимость от ПО из-за рубежа, недостаточный уровень исследований, низкий уровень кадрового обеспечения и др. Данная работа призвана внести некоторый вклад в решение подобных проблем.

Данная работа посвящена системам диагностирования с одной стороны, и интеллектуальным системам – с другой. Актуальность разработок в любой из этих областей знаний также подтверждается рядом нормативных документов [2,4].

Целью данной работы является выбор алгоритмов и определение необходимых стадий технологического процесса кластеризации журнальных файлов серверов, обеспечивающего выполнение поставленных требований и разработка приложения на основе этого технологического процесса.

Приложение должно выполнять кластеризацию журнальных файлов с коэффициентом Силуэта не ниже 0,8 и временем кластеризации журнального файла размером 25 тыс. строк на бытовом компьютере MS Windows 10 Professional; Intel(R) Core(TM) i5-4200U CPU @ 1.60GHz 2.30 GHz; 12 ГБ ОЗУ; SSD не более 30 с.

При этом кластеризация должна выполняться без предварительного указания числа кластеров, и требуемое ПО и библиотеки должны устанавливаться только на удаленное рабочее место администратора; на целевой сервер никакого ПО, кроме необходимого для обеспечения удаленного доступа, устанавливаться не должно.

Кластеризация в общем смысле представляет собой процесс разделения исходных образцов на группы таким образом, чтобы различие между образцами внутри одной группы было минимальным, а различие между образцами из разных групп – максимальным [5,6].

Процесс кластеризации (в отличие от другого процесса – классификации) является неконтролируемым, то есть происходит «без учителя». Размеченные наборы данных требуются только для проведения оценки качества кластеризации некоторыми методиками, если такая оценка требуется.

В настоящее время разработано довольно много алгоритмов кластеризации [6]. Но согласно заданию, в данном случае подходят только алгоритмы, не требующие предварительного указания числа кластеров, поэтому для проведения исследования выбраны только DBSCAN и BIRCH [6].

Кроме этого, для целей валидации дополнительно задействован алгоритм k-средних [6].

В качестве основных инструментов для эксперимента и приложения рассматривались язык R, node.js и Python. В силу того, что согласно рейтингам TIOBE Index [7] и IEEE Spectrum [8], Python является самым популярным в мире в 2022 г. языком программирования, в качестве основного инструмента выбран именно он.

В качестве реализации алгоритмов кластеризации и оценки применены библиотека scikit-learn и ряд дополнительных вспомогательных библиотек.

Процесс кластеризации может состоять из нескольких стадий (далее *стадии процесса кластеризации*):

- Чтение лог-файла с диска и разбиение на отдельные сообщения;
- Предварительная обработка текста;
- Отображение текста в векторное пространство;

- Непосредственно кластеризация;
- Преобразование результатов в форму, пригодную для визуализации.

Из этих стадий для оптимизации наиболее интересна стадия предварительной обработки текста, которая тоже в свою состоит из нескольких этапов (далее *этапы предварительной обработки текста*):

- Чистка пробелов;
- Приведение к нижнему регистру;
- Чистка пунктуации;
- Удаление стоп-слов (слов, не несущих смысловой нагрузки);
- Удаление цифр;
- Разбиение текста на отдельные слова;
- Выделение основ слов (стемминг);
- Приведение слов к нормальной форме (лемматизация);
- Отбор/извлечение признаков;
- Исключение редких слов.

Задачей исследования как раз и является оценить влияние наличия или отсутствия каких-либо из этапов предварительной обработки теста на время и качество кластеризации.

Для оценки качества кластеризации применен коэффициент Силуэта в реализации библиотеки scikit-learn. Для оценки времени выполнения написан собственный класс-профилировщик.

За время *tnp* принималось суммарное время всех *стадий процесса кластеризации*, перечисленных выше, за исключением операций оценки, исключенной из производственного приложения.

В ходе эксперимента оценивались журнальные файлы различной структуры и размера. На Рисунках 1 и 2 приведены зависимости коэффициента Силуэта *sil* и времени выполнения кластеризации *tnp* для двух таких файлов, размером 1500 и 25 тыс. строк.

Исключение редких слов выполнялось согласно стратегии, при которой слово считалось редким и исключалось, если встречалось менее чем в трех сообщениях.

Данные зависимости для остальных исследованных журнальных файлов в рамках данной статьи не приводятся, так как сохраняют общие закономерности.

Полученная зависимость показывает, что наиболее эффективным *этапом предварительной обработки текста* является очистка от цифр. Отметим, что данное утверждение справедливо только для журнальных файлов; для других видов текстовых документов очистка от цифр может иметь и негативный для качества кластеризации эффект.

С исходными кодами приложения, используемого для эксперимента, можно ознакомиться по ссылке https://github.com/alborodin85/clustering-logs.

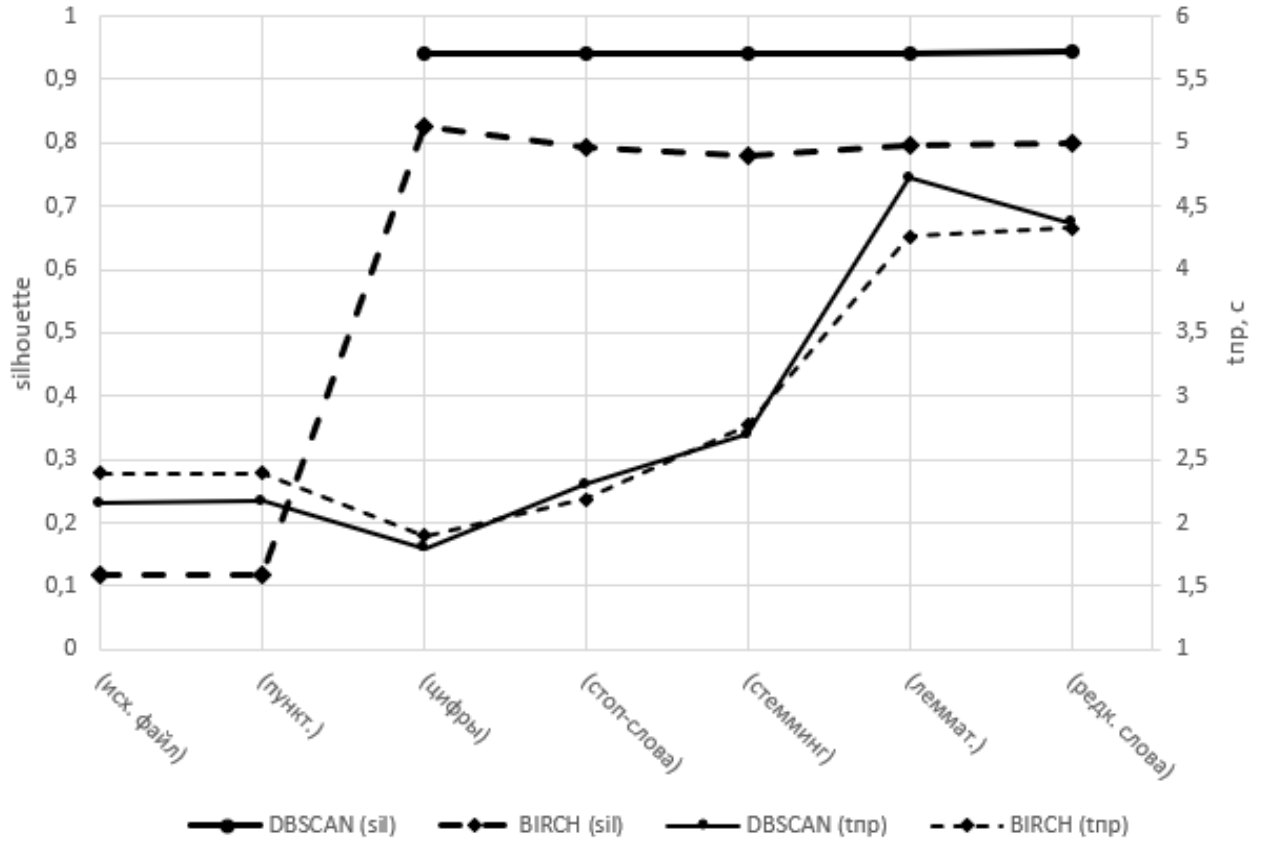

Рисунок 1 – Зависимость коэффициента Силуэта (sil) и времени выполнения процесса кластеризации (tпр) от набора этапов предварительной подготовки текста для журнального файла размером 1500 строк.

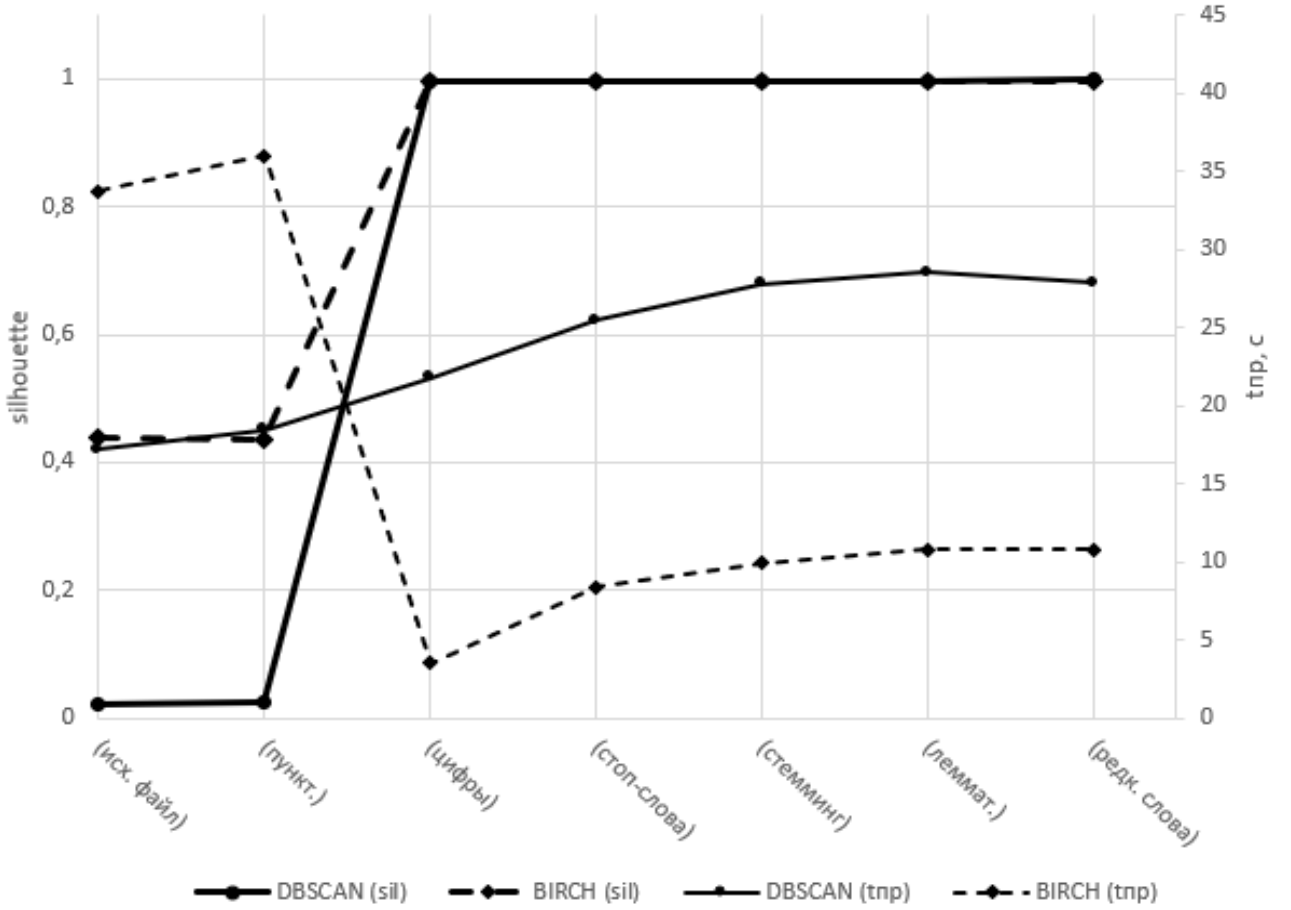

Рисунок 2 – Зависимость коэффициента Силуэта (sil) и времени выполнения процесса кластеризации (tпр) от набора этапов предварительной подготовки текста для журнального файла размером 25 тыс строк.

В результате были выбраны следующие этапы предварительной обработки текста:

- Чистка пробелов (strip);
- Разбиение текста на отдельные слова;
- Приведение к нижнему регистру (lower);
- Удаление цифр (clearDigits).

На основании результатов эксперимента разработано приложение для производственного использования.

Приложение предназначено для работы в ОС Windows10 и интегрируется в известное средство WinSCP для удаленного доступа как пользовательский текстовый редактор.

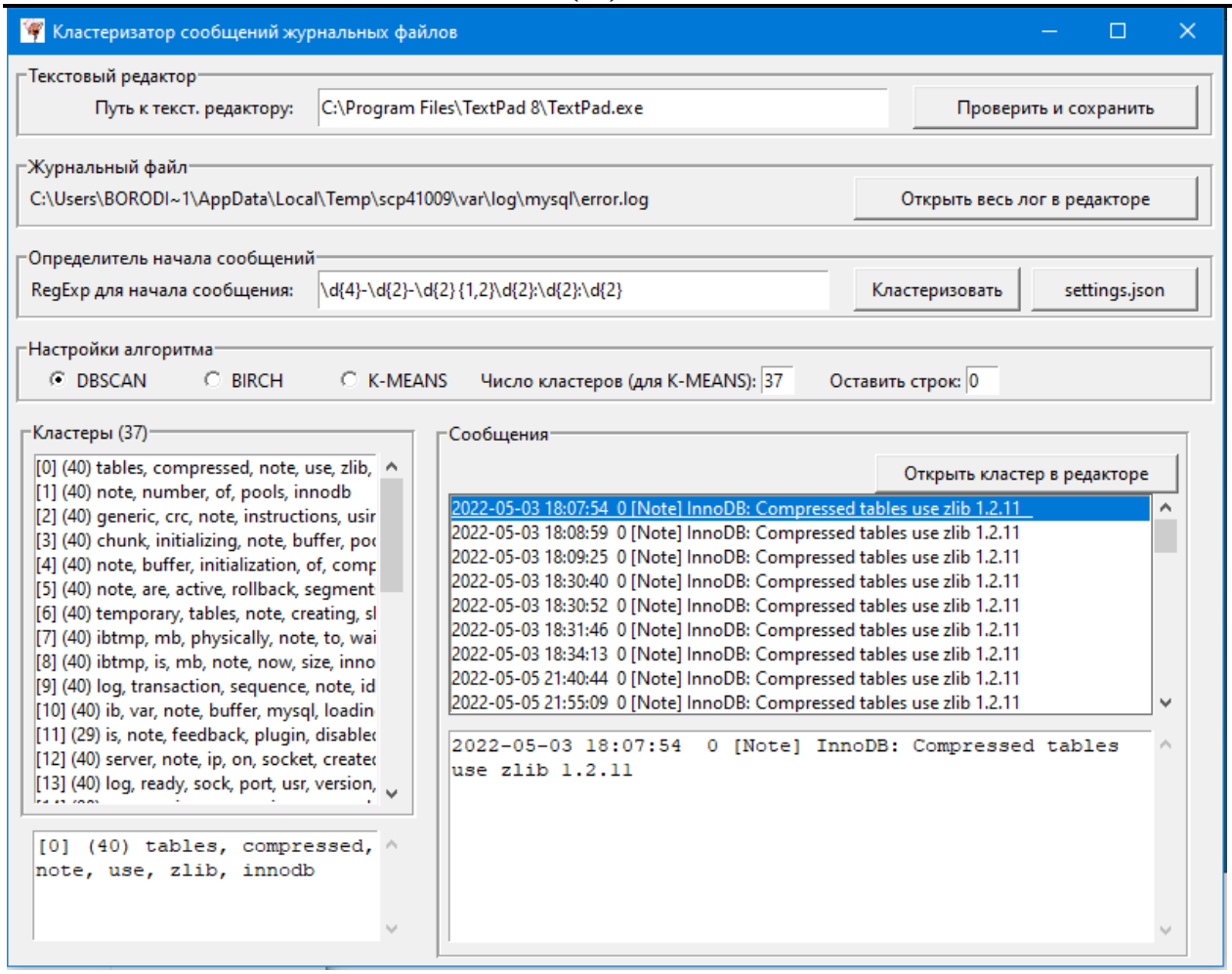

Рисунок 3 – Внешний вид главного окна приложения

Само приложение с подробными описанием и инструкцией размещены в сети Интернет https://github.com/alborodin85/text-clasterisator-winscp-integration, а внешний вид главного окна приложения приведен на рисунке 3.

Таким образом, в ходе работы отечественным производителем разработано с применением интеллектуальных технологий приложение, предназначенное для более эффективного мониторинга и диагностирования.

Актуальность работы подтверждена рядом Российских нормативных документов, ссылки на которые представлены в начале статьи.

Приложение соответствует заданным требованиям, в число которых входят производительность, качество кластеризации, простота использования и отсутствие каких-либо компонент, устанавливаемых на сервер.

Архитектура приложения построена на основе результатов исследования, целью которого являлось определение этапов предварительной обработки текста, обеспечивающей выполнение заданных требований последующей кластеризации.

Также в ходе исследования определена специфика кластеризации сообщений журнальных файлов:

- Самым эффективным этапом предварительной подготовки текстов в этом случае является очистка от цифр;

- Для небольших журнальных файлов (порядка 1500 строк) кластеризация алгоритмами DBSCAN и BIRCH выполняются примерно одинаковое время, но качество кластеризации алгоритмом DBSCAN несколько выше;
- Для журнальных файлов большого размера (порядка 25 тыс. строк) качество кластеризации алгоритмов DBSCAN и BIRCH примерно одинаковое, но скорость алгоритма BIRCH до 5 раз выше.

Список литературы

1. Указ Президента Российской Федерации от 02.03.2022 г. № 83 [Электронный ресурс]. 2 с. URL:http://pravo.gov.ru/proxy/ips/?savertf=&nd=602894815&rdk=&firstDoc=1&lastDoc=1&page=all (дата обращения 24.02.2022).
2. Доктрина информационной безопасности Российской Федерации: утв. Указом Президента Российской Федерации №646 05.12.2016 [Электронный ресурс] : ввод. в действие с 05.12.2016. 16 с. URL: http://pravo.gov.ru/proxy/ips/?savertf=&firstDoc=1&lastDoc=1&nd=102417017&page=all (дата обращения 24.02.2022).
3. Стратегия развития информационного общества в Российской Федерации на 2017 - 2030 годы: утв. Указом Президента Российской Федерации №203 09.05.2017 [Электронный ресурс]: ввод. в действие с 09.05.2017. 28 с. URL: http://pravo.gov.ru/proxy/ips/?savertf=&firstDoc=1&lastDoc=1&nd=102431687&page=all (дата обращения 24.02.2022).
4. Национальная стратегия развития искусственного интеллекта на период до 2030 года: утв. Указом Президента Российской Федерации №490 10.10.2019 [Электронный ресурс] : ввод. в действие с 10.10.2019. 25 с. URL: http://static.kremlin.ru/media/events/files/ru/AH4x6HgKWANwVtMOfPDhcbRpvd1HCCsv.pdf (дата обращения 24.02.2022).
5. Габдрахманова Н. Т. Кластеризация документов с помощью нейронных сетей // Речевые технологии. 2019. N 1. С. 45–53.
6. Кошкин Д. Е., Багдасарова Н. В. Анализ и сравнение алгоритмов кластеризации данных применительно к кластеризации текстового контента // Информатизация образования и науки. 2018. N 4(40). С. 116–128.
7. TIOBE Index for September 2022 [Электронный ресурс] // TIOBE Software BV. 2022. URL: https://www.tiobe.com/tiobe-index/ (дата обращения 11.09.2022).
8. Top Programming Languages 2022 [Электронный ресурс] // IEEE Spectrum. 2022. URL: https://spectrum.ieee.org/top-programming-languages-2022/ieee-spectrums-top-programming-languages-2022 (дата обращения 11.09.2022).

References

1. Decree of the President of the Russian Federation of 02.03.2022 № 83 [Electronic resource]. 2 c. URL: http://pravo.gov.ru/proxy/ips/?savertf=&nd=602894815&rdk=&firstDoc=1&lastDoc=1&page=all (accessed 24.02.2022).
2. Doctrine of information security of the Russian Federation: approved by the Decree of the President of the Russian Federation №646 on 05.12.2016 [Electronic resource] : enacted from

05.12.2016. 16 с. URL: http://pravo.gov.ru/proxy/ips/?savertf=&firstDoc=1&lastDoc=1&nd=102417017&page=all (accessed 24.02.2022).

3. Strategy for the development of information society in the Russian Federation for 2017 - 2030 years: approved by the Decree of the President of the Russian Federation №203 on 09.05.2017 [Electronic resource] : effective from 09.05.2017. 28 p. URL: http://pravo.gov.ru/proxy/ips/?savertf=&firstDoc=1&lastDoc=1&nd=102431687&page=all (accessed 24.02.2022).
4. National strategy for the development of artificial intelligence for the period up to 2030: approved by the Decree of the President of the Russian Federation №490 10.10.2019 [Electronic resource] : put into effect from 10.10.2019. 25 p. URL: http://static.kremlin.ru/media/events/files/ru/AH4x6HgKWANwVtMOfPDhcbRpvd1HCCsv.pdf (date of reference 24.02.2022).
5. Gabdrakhmanova N. T. Document clustering using neural networks // Speech Technology. 2019. N 1. pp. 45-53.
6. Koshkin D. E., Bagdasarova N. V. Analysis and comparison of data clustering algorithms as applied to clustering of textual content // Informatization of education and science. 2018. N 4(40). pp. 116-128.
7. TIOBE Index for September 2022 [Electronic resource] // TIOBE Software BV. 2022. URL: https://www.tiobe.com/tiobe-index/ (accessed 11.09.2022).
8. Top Programming Languages 2022 [Electronic resource] // IEEE Spectrum. 2022. URL: https://spectrum.ieee.org/top-programming-languages-2022/ieee-spectrums-top-programming-languages-2022 (accessed 11.09.2022).