УДК 004.853
DOI: 10.25559/SITITO.16.202001.81-89

Разработка банка алгоритмов и метода поиска программ в соответствии с требованиями пользователей

А. А. Куликовская[1,2], Е. А. Доренская[1*], Ю. А. Семенов[1,3]

[1] Институт теоретической и экспериментальной физики имени А.И. Алиханова Национального исследовательского центра «Курчатовский институт», г. Москва, Россия
117218, Россия, г. Москва, ул. Большая Черёмушкинская, д. 25
* dorenskaya@ itep.ru

[2] Национальный исследовательский центр «Курчатовский институт», г. Москва, Россия
123182, Россия, г. Москва, пл. Академика Курчатова, д. 1

[3] Московский физико-технический институт (национальный исследовательский университет), г. Долгопрудный, Россия
141701, Россия, Московская обл., г. Долгопрудный, Институтский пер., д. 9

Аннотация

Создан банк программ на языке Perl. Каждый программный модуль снабжен описанием и списком ключевых слов. По данным ключевым словам, пользователь может указать область, к которой относится интересующая его программа. В данной статье рассматривается алгоритм, выполняющий поиск программы в созданной базе данных. Поиск основан на выявлении совпадений между ключевыми словами, которые были указаны авторами программ и выбраны пользователем из предлагаемого списка диалоговым методом. Пользователь последовательно добавляет ключевые слова и в интерактивном режиме может посмотреть, нашлась ли интересующая его программа. Исследования выполнены в ФГБУ «Институт теоретической и экспериментальной физики имени А.И. Алиханова Национального исследовательского центра «Курчатовский институт». Предлагаемый алгоритм гарантирует однозначность поиска. Многократное использование программ разными людьми с использованием различных исходных данных приводит к уменьшению количества ошибок. При обнаружении неправильной работы программ или при нахождении ошибок в коде пользователи имеют возможность сообщить об этом администратору базы данных, либо автору программы. Предполагается, что пользователи, при отсутствии желаемой программы в базе данных, могут написать ее и предложить для добавления в существующий банк. Таким образом, количество программ будет расти. Статистический анализ показывает, что данный метод можно использовать и для значительно большего числа программ в базе (порядка 10000 и более), чем существует на текущий момент.

Ключевые слова: Database, SQL, Perl, MySQL, СУБД, программа, поиск наилучшего соответствия, ключевые слова.

Для цитирования: Куликовская, А. А. Разработка банка алгоритмов и метода поиска программ в соответствии с требованиями пользователей / А. А. Куликовская, Е. А. Доренская, Ю. А. Семенов. – DOI 10.25559/SITITO.16.202001.81-89 // Современные информационные технологии и ИТ-образование. – 2020. – Т. 16, № 1. – С. 81-89.

Development of an Algorithm's Bank and Method for Searching Programs in Accordance with User Requirements

A. A. Kulikovskaya[a,b], E. A. Dorenskaya[a*], Yu. A. Semenov[a,c]

[a] Alikhanov Institute for Theoretical and Experimental Physics of National Research Center "Kurchatov Institute", Moscow, Russia
25 Bolshaya Cheremushkinskaya Str., Moscow 117218, Russia
[*] dorenskaya@ itep.ru
[b] National Research Center "Kurchatov Institute", Moscow, Russia
1 Akademika Kurchatova pl., Moscow 123182, Russia
[c] Moscow Institute of Physics and Technology (National Research University), Dolgoprudny, Russia
9 Institutskij per., Dolgoprudny 141701, Moscow region, Russia

Abstract

A database containing Perl programs, descriptions and a list of keywords for each program was developed. According to these keywords, the user can determine the area of interest which includes the program, he is looking for. This article discusses an algorithm of searching that program in the created database. The search is based on finding matches between keywords, which the authors of the programs indicated and keywords, which the user selected from the proposed list. The user sequentially adds keywords and can interactively see, whether the required program has been found or not. The work was carried out at Institute for Theoretical and Experimental Physics named after A.I. Alikhanov of National Research Centre «Kurchatov Institute». One of the objectives of the proposed algorithm is to guarantee unambiguity of the search. The side result we expect is a bugs fixing in existing programs, through the repeated use of them by many users. This goal can be achieved through numerous tests with wide range of input data by different users and viewing the codes of existing programs by different people. In case of program malfunction or if errors are found in the code, users will be able to inform the database administrator or the author of the program about this. It is assumed, that users, in the absence of the desired program in the database, will be able to write it and add to the existing database, thereby increasing the number of programs. Statistical analysis shows that this method can also be used for a significantly larger quantity of programs in the database than currently exists.

Keywords: Database, SQL, Perl, MySQL, DBMS, program, best match, keywords, many eyes method.

For citation: Kulikovskaya A.A., Dorenskaya E.A., Semenov Yu.A. Development of an Algorithm's Bank and Method for Searching Programs in Accordance with User Requirements. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2020; 16(1):81-89. DOI: https://doi.org/10.25559/SITITO.16.202001.81-89

Введение

Поиск нужных программных модулей в банке описаний алгоритмов по ключевым словам с помощью традиционных алгоритмов не представляется разумным. Если бы мы знали имя искомого модуля, задача была бы тривиальной, но имя программы чаще всего неизвестно пользователю. Очевидно, что объем банка будет расти со временем и нам нужно знать минимальное число ключевых слов, которые должны характеризовать программный модуль, чтобы отсутствовало совпадение набора выбранных ключевых слов у двух и более программ. Наборы ключевых слов для программ формируют их разработчики. Понятно, что наборы ключевых слов разных модулей должны отличаться друг от друга как можно больше [1,2]. Но так как программы разрабатываются и описываются разными людьми, некоторое число ключевых слов для разных программ могут совпадать.

Цель исследования

Наша программа предназначена для диалогового метода поиска программных модулей в банке, встроенном в систему автоматизации программирования. Система должна выдавать всегда один или ноль найденных объектов. В традиционных поисковых системах ключевые слова выбирает и вводит сам пользователь. При этом возможны опечатки. Ключевые же слова, содержащиеся в банке и характеризующие конкретный программный модуль, готовит чаще всего программист, написавший программу. Ожидать, что наборы этих ключевых слов будут идентичны или хотя бы близки, друг другу не приходится. Нами исследована зависимость необходимого числа ключевых слов от объема банка алгоритмов.

Если в банке находится M программ и у каждой в среднем 10 ключевых слов, то полное число ключевых слов равно 10*M. Операции с таким числом слов, если M>1000, достаточно проблематичны.

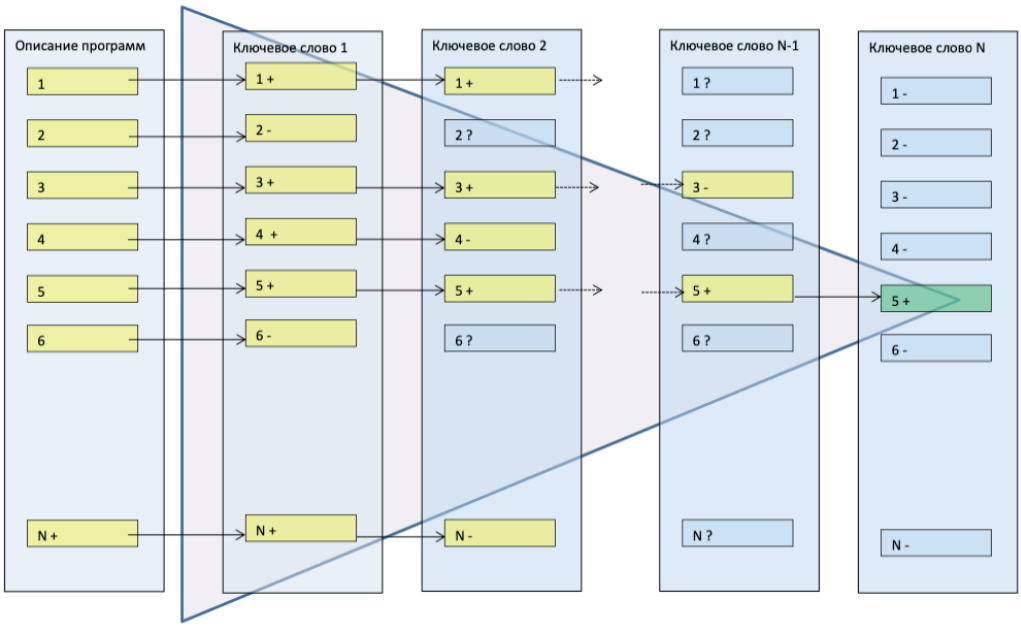
Для того чтобы упростить поиск, программные модули делятся на секции по типу и проблематике (вычислительные, управляющие программы, CGI, html со скриптами на JavaScript и т.д.). Кроме того, модули делятся на несколько категорий по характеру их функции, например: **find, calculate, control, determine, develop, create, form, write, read, load, detect, measure, analyze, convert from, convert to, approximate, interpolate, extrapolate, minimize, maximize, encode, decode, copy, transfer, optimize, send,...[3]**

Программы, написанные человеком, всегда могут содержать ошибки, существующие алгоритмы тестирования программ не позволяют определить, верно ли программа выполняет свои функции во всех возможных случаях. Именно поэтому наиболее часто встречается программное обеспечение с открытым кодом, чтобы любой пользователь, при обнаружении ошибок мог связаться с автором программы, и ошибку ликвидировать. Значительная часть простых вычислительных программ выложена в открытом доступе в Интернет и пользователи могут найти интересующие их модули. Однако нет гарантий, что в таких программах ошибки отсутствуют. Для того чтобы у пользователей была возможность найти готовые программы по ИТ-тематике, создана база данных, содержащая на данных момент 95 программ [4]. Пользователи будут видеть текст программы, смогут ее тестировать и тем самым выполнить проверку программ методом many eyes [5]. Помимо самих программ, их описаний и ключевых слов в базе данных содержатся данные об авторе. Предполагается, что пользователи, при отсутствии желаемой программы в базе данных, смогут написать ее и предложить для добавления в существующий банк, тем самым количество программ будет расти. А при частом использовании программ количество ошибок в них будет снижаться.

Основная часть

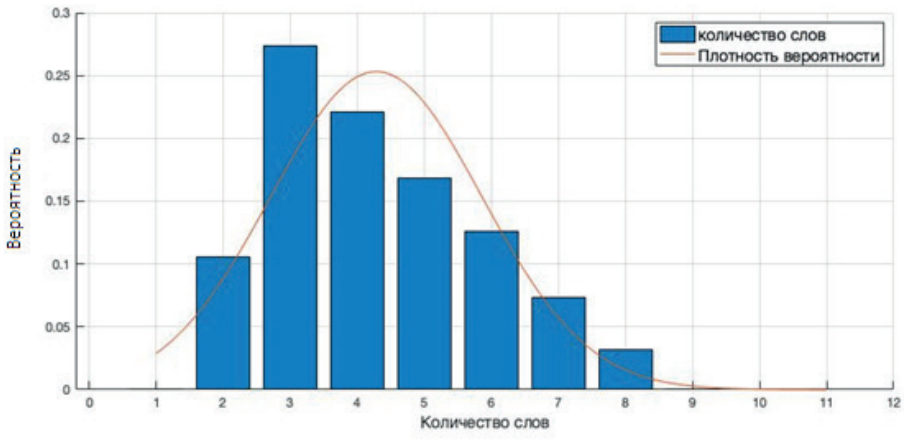
Необходимо создать программное обеспечение, способное хранить и предоставлять пользователям текст программ, которые они смогут просмотреть и сообщить об ошибках, если таковые обнаружатся. Поиск программ реализован по ключевым словам. В программе существует таблица с названиями программ и списком ключевых слов для каждой из них. Пользователю необходимо последовательно выбирать из выпадающих списков ключевые слова, и в интерактивном режиме будет сообщаться количество программ с данными ключевыми словами. Как только будет найдена одна и только одна программа, ее название выводится, и пользователь получает доступ к коду. При последовательном выборе ключевых слов, происходит уменьшение количества программ, содержащих в своем описании ключевые слова, которые указал пользователь [6-10]. При отсутствии программы с выбранными пользователем ключевыми словами в базе данных пользователю будет сообщено.

В таблице, нужно выбирать по очереди слова из выпадающего списка, и в первой строке будут меняться значения, где написано, нужны ли еще уточнения, либо нашлось единственное совпадение. При единственном совпадении для заданной комбинации ключевых слов, название этой программы будет в соответствующем окне. Будет также выведено описание найденной программы. При добавлении новых программ и выборе ключевых слов для них должно выполняться следующее правило: “В базе данных не существует двух программ, выполняющих одни и те же функции, с одинаковым набором ключевых слов”. Пример работы с алгоритмом приведен на рисунке 1.В первом столбце находится описание работы программ и ключевые слова. Во втором и последующих N столбцах приведены ключевые слова. Ключевые слова выбираются из выпадающего списка. Пользователь сам задает набор ключевых слов, последовательно выбирая в выпадающих списках те из них, которые доступны для уточнения. Выбирать пользователь должен до тех пор, пока не будет найдена единственная программа, отвечающая выбранным ключевым словам, либо не будет получено сообщение об её отсутствии. На рисунке 1 после ввода одного ключевого слова найдено 5 программ, в описании которых оно присутствует. Такой результат считается неуспешным (неокончательным), так как необходимо, чтобы совпадение было единственным. Требуется дальнейшее уточнение путем добавления очередного ключевого слова. Комбинация первого и второго ключевых слов есть только в трех программах из списка, комбинация N-1 слова только в двух программах. Комбинация из N выбранных различных ключевых слов, которые указал пользователь, присутствует только в одной программе, эта программа считается результатом работы алгоритма. При этом в списке ключевых слов в описании программы могут быть также и другие ключевые слова.



Р и с. 1. Выбор ключевого слова из выпадающего списка

F i g. 1. Selecting a keyword from the dropdown list



Р и с. 2. Распределение числа ключевых слов, при нахождении нужного программного модуля и аппроксимация этих данных функцией Гаусса (μ=4,2842; σ=1,5755)

F i g. 2. Distribution of the number of keywords when finding the required software module and approximation of these data by the Gaussian function (μ = 4.2842; σ = 1.5755)

Плюсом обозначено наличие выбранного ключевого слова в описании программы, минусом - отсутствие. Вопросительным знаком обозначены строки, которые не рассматриваются, так как эти программы удалены из рассмотрения ранее. Желтым выделены наборы ключевых слов и совпадения, когда слово, выбранное пользователем, есть в наборе ключевых слов программы, которые указал автор. Зеленым выделена программа, соответствующая всем заданным ключевым словам. Так как выбор ключевых слов выполняется последовательно, на каждом шаге сохраняются в списке только те из них, которые есть в списках программ, прошедших предыдущие этапы отбора, за исключением уже выбранных слов.

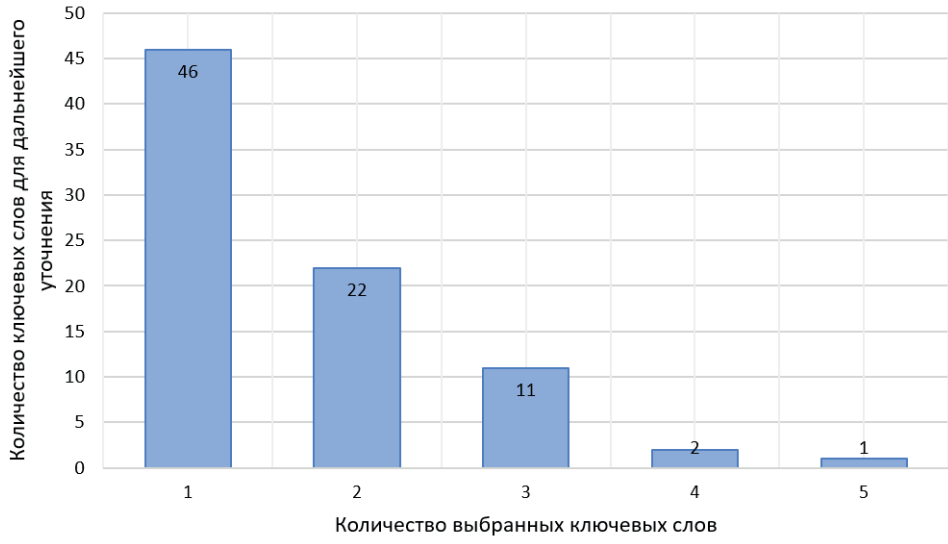
Полученные результаты

На рисунке 2 приведено распределение вероятности нахождения нужной программы при вводе определенного числа ключевых слов. По горизонтальной оси отложено количество ключевых слов, указанных пользователем во время работы алгоритма. По вертикальной оси указана вероятность нахождения программы с таким набором ключевых слов. Из распределения можно сделать вывод, что максимальное количество программ (26%) находится по **3** ключевым словам, что соответствует 0,01 от всех существующих на данный момент ключевых слов, так как всего насчитывается 367 различных

ключевых слов. Распределение на рис 2 аппроксимировано функцией Гаусса, и посчитан интеграл от максимального значения до бесконечности. Значение интеграла равно 0.0002, что составляет около 0.02% от интеграла функции Гаусса от 0 до бесконечности. Следовательно, вероятность того, что ключевых слов, указанных автором, не хватит для нахождения программы составляет менее 0.1%. Данное распределение было сформировано для оценки вероятности того, что ключевых слов в описании при данном алгоритме, может не хватить для нахождения нужной программы.

На рисунке 3 показано как меняется количество ключевых слов. После каждого введенного ключевого слова в последующем столбце (рис. 1) исключаются те ключевые слова, которые отсутствуют в описании программ, прошедших предыдущий этап отбора. Допустим, после указания первого ключевого слова подходит 100 программ, тогда во втором столбце можно выбрать только слова, которые встречаются в описании этих 100 программ. Таким образом, количество ключевых слов будет сокращаться после каждого шага. На рисунке 3 приведено распределение подгрупп ключевых слов. Для чисел в колонках выполнено округление до ближайшего целого.

По горизонтали отмечено количество ключевых слов, по вертикали - среднее число ключевых слов, которые присутствуют в описаниях программам, прошедших отбор. Первоначально для выбора ключевого слова предлагалось 364 варианта. После выбора первого ключевого слова в различных случаях подходит от 3 до 48 программ, которые в своих описаниях в сумме содержат от 11 до 173 различных ключевых слов. Количество ключевых слов для уточнения во втором (и в каждом последующем) выпадающем списке сокращается. В среднем, число различных ключевых слов в описании программ с выбранным первым ключевым словом в рассматриваемым варианте равно 46. Это число записано в верхней части левого столбца диаграммы. Выбор второго ключевого слова в среднем можно сделать уже только из этих 46 терминов, которые присутствуют в описании программ, отобранных ранее. При последующем выборе число доступных ключевых слов уменьшается. По графику можно сделать вывод, что при данном количестве программ нужный программный модуль будет найден после выбора 5 ключевых слов.



Р и с. 3. Средние значения ключевых слов на каждом этапе поиска программы

F i g. 3. Average keyword values at each stage of the program search

Современный словарь IT-терминов содержит более 25000 слов[1], что является слишком большим количеством для перебора. Количество известных программ, написанных за 2016 и 2017 годы превышает 3,6 млн[2]. Однако многие программы имеют достаточно узкую тематику, и если термины разделить на категории, то в каждой категории уже можно вручную ознакомиться со списком терминов и указать подходящие ключевые слова, для поиска интересующей программы[3]. Получается, что для описания работы «бесконечного» числа программ достаточно 25000 слов, что позволяет сделать вывод о том, что зависимость числа различных ключевых слов, указанных в документации к программам, и количеством самих программ достаточно слабая. Действительно, используя набор из N ключевых слов одной категории, можно создать 2^N различных их комбинаций [11]. Таким образом, можно сделать вывод, что при увеличении количества программ в базе данных в 100 раз, необходимое количество ключевых слов увеличится незначительно (см. рис.2), и алгоритм, описанный в данной статье, будет работать корректно.

[1] Computer Language.com. Longest-Running Tech Encyclopedia [Электронный ресурс]. URL: https://www.computerlanguage.com/ (дата обращения: 17.01.2020).

[2] IEEE Top Programming Languages: 2018 Design, Methods, and Data Sources [Электронный ресурс]. URL: https://spectrum.ieee.org/static/ieee-top-programming-languages-2018-methods (дата обращения: 17.01.2020).

[3] Куликовская, А. А. Разработка метода анализа тенденций развития Интернет [Электронный ресурс]. URL: http://saturn.itep.ru/conf_pub.pdf (дата обращения: 17.01.2020).

Заключение

Был создан алгоритм, позволяющий найти программу, соответствующую запросу пользователя, в существующем банке готовых программ. Если такой программы в банке нет, об этом также сообщается. На данный момент в банке содержится 95 описаний программ, для каждой из которых имеется 10-30 ключевых слов). В описаниях всех программ в банке на данный момент содержатся 364 разных ключевых слов, которые были подобраны разработчиками. По этим ключевым словам пользователь может осуществлять поиск в банке алгоритмов. Время для вывода результата работы программы составляет менее секунды. Статистика показывает, что в среднем программа находит нужный программный модуль, использовав 4,5 ключевых слов, но данное число может меняться, в зависимости от тематики задачи. В более чем 99% случаев искомая программа была найдена при вводе 8 ключевых слов, что можно заметить из рисунка 2. При увеличении количества программ в базе данных в 100 раз, алгоритм, описанный выше, будет по-прежнему работать верно. Это говорит о том, что алгоритм не придется модифицировать при увеличении объема банка на 1-2 порядка. Скорость работы алгоритма при этом останется в пределах нескольких секунд.

Список использованных источников

[1] Kim, J. A Statistical Ontology-Based Approach to Ranking for Multiword Search: dis. ... Ph.D. (Computer Science). Ann Arbor: University of Southern California, ProQuest LLC, 2013.

[2] Beall, J. The Weaknesses of Full-Text Searching / J. Beall. – DOI: 10.1016/j.acalib.2008.06.007 // Journal of Academic Librarianship. – 2008. – Vol. 34, issue 5. – Pp. 438-444. – URL: https://www.sciencedirect.com/science/article/pii/S0099133308001067?via%3Dihub (дата обращения: 17.01.2020).

[3] Shannon, G. Cognition-Based Approaches for High-Precision Text Mining: dis. ... Ph.D. (Engineering). Ann Arbor: Missouri University of Science and Technology, ProQuest LLC, 2017.

[4] Гурин, Н. И. Генератор семантической сети информационной системы в таблицу реляционной базы данных / Н. И. Гурин, Я. А. Жук // Труды Белорусского государственного технологического университета. – 2015. – № 6(179). – С. 181-185. – URL: https://elib.belstu.by/bitstream/123456789/15302/1/generator-semanticheskoi-seti-informacionnoi-sistemy-v-tablicu-relyacionnoi-bazy-dannyx-n.-i.-gurin-ya.-a.-zhuk.pdf (дата обращения: 17.01.2020). – Рез. англ.

[5] Доренская, Е. А. О технологии программирования, ориентированной на минимизацию программных ошибок / Е. А. Доренская, Ю. А. Семенов. – DOI: 10.25559/SITITO.2017.2.226 // Современные информационные технологии и ИТ-образование. – 2017. – Т. 13, № 2. – С. 50-56. – URL: https://elibrary.ru/item.asp?id=30258630 (дата обращения: 17.01.2020). – Рез. англ.

[6] Johns, B. T. A Large-Scale Analysis of Variance in Written Language / B. T. Johns, R. K. Jamieson. – DOI: 10.1111/cogs.12583 // Cognitive Science. – 2018. – Vol. 42, issue 4. – Pp. 1360-1374. – URL: https://onlinelibrary.wiley.com/doi/full/10.1111/cogs.12583 (дата обращения: 17.01.2020).

[7] Диковицкий, В. В. Семантический анализ текста с применением нейросетевого анализа морфологии и синтаксиса / В. В. Диковицкий // Труды Кольского научного центра РАН. Серия: Управление, вычислительная техника и информатика. – 2017. – Т. 8, № 3-8. – С. 109-115. – URL: https://www.elibrary.ru/item.asp?id=35093836 (дата обращения: 17.01.2020). – Рез. англ.

[8] Dorenskaya, E. A. New Methods of Minimizing the Errors in the Software / E. A. Dorenskaya, Yu. A. Semenov // CEUR Workshop Proceedings: Proceedings of the VIII International Conference "Distributed Computing and Grid-technologies in Science and Education" (GRID 2018). – Dubna, Moscow region, Russia, September 10 -14, 2018. – Vol. 2267. – Pp. 150-154. – URL: http://ceur-ws.org/Vol-2267/150-154-paper-27.pdf (дата обращения: 17.01.2020).

[9] Eidlin, A. A. Analyzing Weak Semantic Map of Word Senses / A. A. Eidlin, M. A. Eidlina, A. V. Samsonovich. – DOI: 10.1016/j.procs.2018.01.023 // Procedia Computer Science. – 2018. – Vol. 123. – Pp. 140-148. – URL: https://www.sciencedirect.com/science/article/pii/S1877050918300243?via%3Dihub (дата обращения: 17.01.2020).

[10] Yan, E. Understanding disciplinary vocabularies using a full-text enabled domain-independent term extraction approach / E. Yan, J. Williams, Z. Chen. – DOI: 10.1371/journal.pone.0187762 // PLoS ONE. – 2017. – Vol. 12, No. 11. – Article number: e0187762. – URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0187762 (дата обращения: 17.01.2020).

[11] Вьюгин В. В. Дискретная математика. Часть 1. Элементы теории множеств. Комбинаторика. Функции алгебры логики / В. В. Вьюгин. – М: ГКА им. Маймонида, 2009. – URL: http://iitp.ru/upload/publications/5948/discret1.pdf (дата обращения: 17.01.2020).

[12] Zhang, H. eRerevise: Using Natural Language Processing to Provide Formative Feedback on Text Evidence Usage in Student Writing / H. Zhang, A. Magooda, D. Litman, R. Correnti, E. Wang, L. C. Matsumura, E. Howe, R. Quintana. – DOI: 10.1609/aaai.v33i01.33019619 // Proceedings of the AAAI Conference on Artificial Intelligence. – 2019. – Vol. 33, No. 01: AAAI-19, IAAI-19, EAAI-20. – Pp. 9619-9625. – URL: https://aaai.org/ojs/index.php/AAAI/article/view/5025 (дата обращения: 17.01.2020).

[13] Kintz, S. Semantic Knowledge Use in Discourse: Influence of Age / S. Kintz, H. H. Wright. – DOI: 10.1080/0163853X.2016.1150652 // Discourse Processes. – 2017. – Vol. 54, issue 8. – Pp. 670-681. – URL: https://www.tandfonline.com/doi/full/10.1080/0163853X.2016.1150652 (дата обращения: 17.01.2020).

[14] Snyder, R. M. Ideas on Learning a New Language Intertwined with the Current State of Natural Language Processing and Computational Linguistics / R. M. Snyder // Proceedings of the 2015 ASCUE Summer Conference / ed. by P. Smith. – 48th, Myrtle Beach, SC, 2015. – Pp. 75-85. – URL: https://files.eric.ed.gov/fulltext/ED571276.pdf (дата обращения: 17.01.2020).

[15] Crossley, S. A. Predicting Text Comprehension, Processing, and Familiarity in Adult Readers: New Approaches to Readability Formulas / S. A. Crossley,

S. Skalicky, M. Dascalu, D. S. McNamara, K. Kyle. – DOI: 10.1080/0163853X.2017.1296264 // Discourse Processes. – 2017. – Vol. 54, issue 5-6: 2016 Society for Text and Discourse Annual Meeting. – Pp. 340-359. – URL: https://www.tandfonline.com/doi/full/10.1080/0163853X.2017.1296264 (дата обращения: 17.01.2020).

[16] Crossley, S. A. Analyzing Discourse Processing Using a Simple Natural Language Processing Tool / S. A. Crossley, L. K. Allen, K. Kyle, D. S. McNamara. – DOI: 10.1080/0163853X.2014.910723 // Discourse Processes. – 2014. – Vol. 51, issue 5-6: 2013 Society for Text and Discourse Annual Meeting. – Pp. 511-534. – URL: https://www.tandfonline.com/doi/full/10.1080/0163853X.2014.910723 (дата обращения: 17.01.2020).

[17] Седова, Я. А. Разработка интеллектуальной поисковой системы по научным социальным сетям / Я. А. Седова // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. – 2015. – № 4. – С. 124-133. – URL: https://www.elibrary.ru/item.asp?id=24891131 (дата обращения: 17.01.2020). – Рез. англ.

[18] Котов, Э. М. Исследование моделей информационного поиска / Э. М. Котов, А. Н. Целых // Известия Южного федерального университета. Технические науки. – 2009. – № 4 (93). – С. 163-168. – URL: https://www.elibrary.ru/item.asp?id=12834639 (дата обращения: 17.01.2020). – Рез. англ.

[19] Письмак, А. Е. Оценка семантической близости предложений на естественном языке методами математической статистики / А. Е. Письмак, А. Е. Харитонова, Е.А. Цопа, С. В. Клименков. – DOI: 10.17586/2226-1494-2016-16-2-324-330 // Научно-технический вестник информационных технологий, механики и оптики. – 2016. – Т. 16, № 2. – С. 324-330. – URL: https://ntv.ifmo.ru/ru/article/15193/ocenka_semanticheskoy_blizosti_predlozheniy_na_estestvennom_yazyke_metodami_matematicheskoy_statistiki.htm (дата обращения: 17.01.2020). – Рез. англ.

[20] Бермудес, С. Х. Г. О методе определения текстовой близости основанном на семантических классах / С. Х. Г. Бермудес, С. У. Керимова // Инженерный вестник Дона. – 2016. – № 4(43). – С. 70. – URL: https://www.elibrary.ru/item.asp?id=28945370 (дата обращения: 17.01.2020). – Рез. англ.

[21] Усталов, Д. А. Комплекс программ автоматического построения семантической сети слов / Д. А. Усталов, А. В. Созыкин. – DOI: 10.14529/cmse170205 // Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика. – 2017. – Т. 6, № 2. – С. 69-83. – URL: https://www.elibrary.ru/item.asp?id=29410450 (дата обращения: 17.01.2020). – Рез. англ.

[22] Савотченко, С. Е. Показатели оценки качества пертинентности результатов автоматизированного поиска в информационных системах / С. Е. Савотченко // Научные ведомости Белгородского государственного университета. Серия: Экономика. Информатика. – 2016. – 9(230). – С. 135-138. – URL: https://www.elibrary.ru/item.asp?id=27176703 (дата обращения: 17.01.2020). – Рез. англ.

[23] Савотченко, С. Е. Математические методы исследования семантических особенностей подсистемы поиска в автоматизированных информационных системах / С. Е. Савотченко, Е. А. Проскурина // Вестник Сибирского института бизнеса и информационных технологий. – 2015. – № 1(13). – С. 69-76. – URL: https://www.elibrary.ru/item.asp?id=23567314 (дата обращения: 17.01.2020). – Рез. англ.

[24] Федотов, А. М. Проблемы поиска информации: история и технологии / А. М. Федотов, В. Б. Барахнин // Вестник Новосибирского государственного университета. Серия: Информационные технологии. – 2009. – Т. 7, № 2. – С. 3-14. – URL: https://www.elibrary.ru/item.asp?id=12916756 (дата обращения: 17.01.2020). – Рез. англ.

[25] Грекова, Л. В. Информационный поиск в информатике и библиотековедении / Л. В. Грекова // Научные ведомости Белгородского государственного университета. Серия: Экономика. Информатика. – 2013. – №1(144). – С. 95-100. – URL: https://www.elibrary.ru/item.asp?id=20926084 (дата обращения: 17.01.2020). – Рез. англ.

[26] Kenett, Y. N. The semantic distance task: Quantifying semantic distance with semantic network path length / Y. N. Kenett, E. Levi, D. Anaki, M. Faust. – DOI: 10.1037/xlm0000391 // Journal of Experimental Psychology: Learning, Memory, and Cognition. – 2017. – Vol. 43, issue 9. – Pp. 1470-1489. – URL: https://doi.apa.org/doiLanding?doi=10.1037%2Fxlm0000391 (дата обращения: 17.01.2020).

[27] Mimeau, C. The Relation between Syntactic Awareness and Contextual Facilitation in Word Reading: What Is the Role of Semantics? / C. Mimeau, A. Laroche, S. H. Deacon. – DOI: 10.1111/1467-9817.12260 // Journal of Research in Reading. – 2019. – Vol. 42, issue 1. – Pp. 178-192. – URL: https://onlinelibrary.wiley.com/doi/full/10.1111/1467-9817.12260 (дата обращения: 17.01.2020).

[28] Gagné, C. L. Effects of morphology and semantic transparency on typing latencies in english compound and pseudocompound words / C. L. Gagné, T. L. Spalding. – DOI: 10.1037/xlm0000258 // Journal of Experimental Psychology: Learning, Memory, and Cognition. – 2016. – Vol. 42, issue 9. – Pp. 1489-1495. – URL: https://doi.apa.org/doiLanding?doi=10.1037%2Fxlm0000258 (дата обращения: 17.01.2020).

[29] Uba, S. Ya. Semantic Categories of Reporting Verbs across Four Disciplines in Research Articles / S. Ya. Uba. – DOI: 10.5539/elt.v13n1p89 // English Language Teaching. – 2020. – Vol. 13, No. 1. – Pp. 89-98. – URL: http://www.ccsenet.org/journal/index.php/elt/article/view/0/41607 (дата обращения: 17.01.2020).

Поступила 17.01.2020; принята к публикации 10.03.2020; опубликована онлайн 25.05.2020.

Об авторах:

Куликовская Анна Алексеевна, младший научный сотрудник, Институт теоретической и экспериментальной физики имени А.И. Алиханова Национального исследовательского центра «Курчатовский институт» (117218, Россия, г. Москва,

ул. Большая Черёмушкинская, д. 25); аспирант, Национальный исследовательский центр «Курчатовский институт» (123182, Россия, г. Москва, пл. Академика Курчатова, д. 1), ORCID: http://orcid.org/0000-0002-0214-1697, kulikovskaya@phystech.edu

Доренская Елизавета Александровна, инженер-программист, Институт теоретической и экспериментальной физики имени А.И. Алиханова Национального исследовательского центра «Курчатовский институт» (117218, Россия, г. Москва, ул. Большая Черемушкинская, д. 25), ORCID: http://orcid.org/0000-0002-4249-5131, dorenskaya@itep.ru

Семёнов Юрий Алексеевич, ведущий научный сотрудник, Институт теоретической и экспериментальной физики имени А.И. Алиханова Национального исследовательского центра «Курчатовский институт» (117218, Россия, г. Москва, ул. Большая Черемушкинская, д. 25); заместитель заведующего кафедрой информатики и вычислительных сетей, Институт нано-, био-, информационных, когнитивных и социогуманитарных наук и технологий, Московский физико-технический институт (национальный исследовательский университет) (141701, Россия, Московская обл., г. Долгопрудный, Институтский пер., д. 9), кандидат физико-математических наук, ORCID: http://orcid.org/0000-0002-3855-3650, semenov@itep.ru

Все авторы прочитали и одобрили окончательный вариант рукописи.

References

[1] Kim J. A Statistical Ontology-Based Approach to Ranking for Multiword Search: dis. ... Ph.D. (Computer Science). University of Southern California, ProQuest LLC, Ann Arbor; 2013. (In Eng.)

[2] Beall J. The Weaknesses of Full-Text Searching. *Journal of Academic Librarianship*. 2008; 34(5):438-444. (In Eng.) DOI: https://doi.org/10.1016/j.acalib.2008.06.007

[3] Shannon G. Cognition-Based Approaches for High-Precision Text Mining: dis. ... Ph.D. (Engineering). Missouri University of Science and Technology, ProQuest LLC, Ann Arbor; 2017. (In Eng.)

[4] Gurin N.I., Zhuk Ya.A. The Information System Semantic Network to a Relational Database Table Generator. *Proceedings of BSTU*. 2015; (6):181-185. Available at: https://elib.belstu.by/bitstream/123456789/15302/1/generator-semanticheskoi-seti-informacionnoi-sistemy-v-tablicu-relyacionnoi-bazy-dannyx-n.-i.-gurin-ya.-a.-zhuk.pdf (accessed 17.01.2020). (In Russ., abstract in Eng.)

[5] Dorenskaya E.A., Semenov Yu.A. About the Programming Techniques, Oriented to Minimize Errors. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2017; 13(2):50-56. (In Russ., abstract in Eng.) DOI: https://doi.org/10.25559/SITITO.2017.2.226

[6] Johns B.T., Jamieson R.K. A Large-Scale Analysis of Variance in Written Language. *Cognitive Science*. 2018; 42(4):1360-1374. (In Eng.) DOI: https://doi.org/10.1111/cogs.12583

[7] Dikovitskiy V.V. Improving Quality of Visualization by Changing the Convergence Angle. *Proceedings of Kola Science Centre of the Russian Academy of Sciences*. 2017; 8(3-8):109-115. Available at: https://www.elibrary.ru/item.asp?id=35093836 (accessed 17.01.2020). (In Russ., abstract in Eng.)

[8] Dorenskaya E.A., Semenov Yu.A. New Methods of Minimizing the Errors in the Software. In: *CEUR Workshop Proceedings: Proceedings of the VIII International Conference "Distributed Computing and Grid-technologies in Science and Education" (GRID 2018).* Dubna, Moscow region, Russia, September 10 -14, 2018, vol. 2267. p. 150-154. Available at: http://ceur-ws.org/Vol-2267/150-154-paper-27.pdf (accessed 17.01.2020). (In Eng.)

[9] Eidlin A.A., Eidlina M.A., Samsonovich A.V. Analyzing Weak Semantic Map of Word Senses. *Procedia Computer Science*. 2018; 123:140-148. (In Eng.) DOI: https://doi.org/10.1016/j.procs.2018.01.023

[10] Yan E., Williams J., Chen Z. Understanding disciplinary vocab-ularies using a full-text enabled domain-independent term extraction approach. *PLoS ONE*. 2017; 12(11):e0187762. (In Eng.) DOI: https://doi.org/10.1371/journal.pone.0187762

[11] Vyugin V.V. Discrete Mathematics. Part 1. Elements of Set Theory. Combinatorics. Functions of the Algebra of Logic. Moscow; 2009. Available at: http://iitp.ru/upload/publications/5948/discret1.pdf (accessed 17.01.2020). (In Russ.)

[12] Zhang H., Magooda A., Litman D., Correnti R., Wang E., Matsumura L.C., Howe E., Quintana R. eRerevise: Using Natural Language Processing to Provide Formative Feedback on Text Evidence Usage in Student Writing. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2019; 33(01):9619-9625. (In Eng.) DOI: https://doi.org/10.1609/aaai.v33i01.33019619

[13] Kintz S., Wright H.H. Semantic Knowledge Use in Discourse: Influence of Age. *Discourse Processes*. 2017; 54(8):670-681. (In Eng.) DOI: https://doi.org/10.1080/0163853X.2016.1150652

[14] Snyder R.M. Ideas on Learning a New Language Intertwined with the Current State of Natural Language Processing and Computational Linguistics. In: P. Smith (ed.) Proceedings of the 2015 ASCUE Summer Conference. 48th, Myrtle Beach, SC; 2015. p. 75-85. Available at: https://files.eric.ed.gov/fulltext/ED571276.pdf (accessed 17.01.2020). (In Eng.)

[15] Crossley S.A., Skalicky S., Dascalu M., McNamara D.S., Kyle K. Predicting Text Comprehension, Processing, and Familiarity in Adult Readers: New Approaches to Readability Formulas. *Discourse Processes*. 2017; 54(5-6):340-359. (In Eng.) DOI: https://doi.org/10.1080/0163853X.2017.1296264

[16] Crossley S.A., Allen L.K., Kyle K., McNamara D.S. Analyzing Discourse Processing Using a Simple Natural Language Processing Tool. *Discourse Processes*. 2014; 51(5-6):511-534. (In Eng.) DOI: https://doi.org/10.1080/0163853X.2014.910723

[17] Sedova Ya.A. Development of Intellectual Search System for Scientific Social Networks. *Vestnik of Astrakhan State Technical University. Series: Management, Computer Sciences and Informatics*. 2015; (4):124-133. Available at: https://www.elibrary.ru/item.asp?id=24891131 (accessed 17.01.2020). (In Russ., abstract in Eng.)

[18] Kotov E.M., Tselykh A.N. Research of Models for Information Retrieval. *Izvestiya SFedU. Engineering Sciences*. 2009; (4):163-168. Available at: https://www.elibrary.ru/item.asp?id=12834639 (accessed 17.01.2020). (In Russ., abstract in Eng.)

[19] Pismak A.E., Kharitonova A.E., Tsopa E.A., Klimenkov S.V. Evaluation of semantic similarity for sentences in natural language by mathematical statistics methods. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*. 2016; 16(2):324-330. (In Russ., abstract in Eng.) DOI: https://doi.org/10.17586/2226-1494-2016-16-2-324-330

[20] Bermudez S.J.G., Kerimova S.U. About Method of Determination of Textual Proximity, Based on the Semantics Classe. *Inženernyj vestnik Dona* = Engineering journal of Don. 2016; (4):70. Available at: https://www.elibrary.ru/item.asp?id=28945370 (accessed 17.01.2020). (In Russ., abstract in Eng.)

[21] Ustalov D.A., Sozykin A.V. A Software System for Automatic Construction of a Semantic Word Network. *Bulletin of the South Ural State University. Series "Computational Mathematics and Software Engineering"*. 2017; 6(2):69-83. (In Russ., abstract in Eng.) DOI: https://doi.org/10.14529/cmse170205

[22] Savotchenko S.E. Indicators of Quality Assessment Results Pertinence Automated Search in Information Systems. *Belgorod State University. Scientific Bulletin. Series: Economics. Information technologies*. 2016; (9):135-138. Available at: https://www.elibrary.ru/item.asp?id=27176703 (accessed 17.01.2020). (In Russ., abstract in Eng.)

[23] Savotchenko S.E., Proskurina E.A. Mathematical Methods of Investigation of Semantic Features of Search Subsystem in Automated Information Systems. *Vestnik Sibirskogo Instituta Biznesa I Informatsionnykh Tekhnologii.* 2015; (1):69-76. Available at: https://www.elibrary.ru/item.asp?id=23567314 (accessed 17.01.2020). (In Russ., abstract in Eng.)

[24] Fedotov A.M., Barakhnin V.B. Problems of Information Retrieval: History and Technologies. *Vestnik NSU. Series: Information Technologies.* 2009; 7(2):3-14. Available at: https://www.elibrary.ru/item.asp?id=12916756 accessed 17.01.2020). (In Russ., abstract in Eng.)

[25] Grekova L.V. Information Search in Computer Science and Library Science. *Belgorod State University. Scientific Bulletin. Series: Economics. Information technologies*. 2013; (1):95-100. Available at: https://www.elibrary.ru/item.asp?id=20926084 (accessed 17.01.2020). (In Russ., abstract in Eng.)

[26] Kenett Y.N., Levi E., Anaki D., Faust M. The semantic distance task: Quantifying semantic distance with semantic network path length. *Journal of Experimental Psychology: Learning, Memory, and Cognition*. 2017; 43(9):1470-1489. (In Eng.) DOI: https://doi.org/10.1037/xlm0000391

[27] Mimeau C., Laroche A., Deacon S.H. The Relation between Syntactic Awareness and Contextual Facilitation in Word Reading: What Is the Role of Semantics? *Journal of Research in Reading*. 2019; 42(1):178-192. (In Eng.) DOI: https://doi.org/10.1111/1467-9817.12260

[28] Gagné C.L., Spalding T.L. Effects of morphology and semantic transparency on typing latencies in english compound and pseudocompound words. *Journal of Experimental Psychology: Learning, Memory, and Cognition.* 2016; 42(9):1489-1495. (In Eng.) DOI: https://doi.org/10.1037/xlm0000258

[29] Uba S.Ya. Semantic Categories of Reporting Verbs across Four Disciplines in Research Articles. *English Language Teaching*. 2020; 13(1):89-98. (In Eng.) DOI: https://doi.org/10.5539/elt.v13n1p89

Submitted 17.01.2020; revised 10.03.2020;
published online 25.05.2020.

About the authors:

Anna A. Kulikovskaya, Researcher, Alikhanov Institute for Theoretical and Experimental Physics of National Research Center "Kurchatov Institute" (25 Bolshaya Cheremushkinskaya Str., Moscow 117218, Russia); Postgraduate Student, National Research Center "Kurchatov Institute" (1 Akademika Kurchatova pl., Moscow 123182, Russia), ORCID: http://orcid.org/0000-0002-0214-1697, kulikovskaya@phystech.edu

Elizaveta A. Dorenskaya, Software Engineer, Alikhanov Institute for Theoretical and Experimental Physics of National Research Center "Kurchatov Institute" (25 Bolshaya Cheremushkinskaya Str., Moscow 117218, Russia), ORCID: http://orcid.org/0000-0002-4249-5131, dorenskaya@ itep.ru

Yuri A. Semenov, Lead Researcher, Alikhanov Institute for Theoretical and Experimental Physics of National Research Center "Kurchatov Institute" (25 Bolshaya Cheremushkinskaya Str., Moscow 117218, Russia); Deputy Head of the Chair for Computer Science, Institute of Nano-, Bio-, Information, Cognitive and Socio-humanistic Sciences and Technologies, Moscow Institute of Physics and Technology (National Research University) (9 Institutskij per., Dolgoprudny 141701, Moscow region, Russia), Ph.D. (Phys.-Math.), ORCID: http://orcid.org/0000-0002- 3855-3650, semenov@itep.ru

All authors have read and approved the final manuscript.