UDC 004.415.2(045)
DOI:10.18372/1990-5548.58.13513

[1]I. V. Kolomiets,
[2]L. M. Oleshchenko

THE METHOD OF DYNAMIC DESIGN OF A SEARCH ENGINE BASED ON AUTOMATED ANALYSIS OF USER REQUESTS

Department of Computer Systems Software, National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnical Institute," Kyiv, Ukraine
E-mails: [1]klolo966@gmail.com, [2]oleshchenkoliubov@gmail.com

Abstract**—***The article describes the method of dynamically designing a search engine based on the analysis of user search queries and automated user's document generation. The advanced method of dynamic search with the account of individual features of users is offered, which allows to increase the cost of search. The service for designing own search engine is created. It replaces the algorithm of ranking on the algorithm for sorting search engines by category using the pseudo relevance feedback method. The information is filled in with a system of successive objects, which is a list. Objects contain key information, these keys allow to set up work with information and increase the relevance of the search. The service contains graphical tools for updating, storing and deleting information in the search engine. An analysis of the configuration of the search service is performed to increase the end-user search cost. There is a conclusion of the results, which indicates the shortcomings of the search engine developer of this system and generated documentation on its use based on the settings for the end user. The sorting system analyzes the end-user request and provides the information it finds, sorted by relevancy. The developed method provides acceleration of the algorithm of the search engine and the quality of the information found for the user.*

Index Terms—Search engine; relevance information; automation; ranking algorithm; search cost.

I. INTRODUCTION

Nowadays an exponential increase in the number of sources of information in the world is observed, which is due to an increase in the number of its consumers, the volume of created and accessible information. This causes more and more difficulty in the efficient search for information, which is, on the one hand, the features of human-machine interaction, and on the other hand, the semantic heterogeneity of the sources of information [1].

The solution to this problem is the individualization of information search tools based on the use of automation of the process as an integral part of computer-aided design (CAD), the adaptation of the search process to individual user characteristics, that allows to quickly find relevant information. To perform a user request analysis process, the application software must implement the mathematical support for the direct execution of the design procedures.

II. PROBLEM STATEMENT

The problem of modern search systems is consists in that the systems work the to algorithm of ranking in the region (Fig. 1) and do not sufficiently take into account the relevance of the information.

A large number of incorrect queries can affect the ranking algorithm with great accuracy and reduce the cost of search in the region [1].

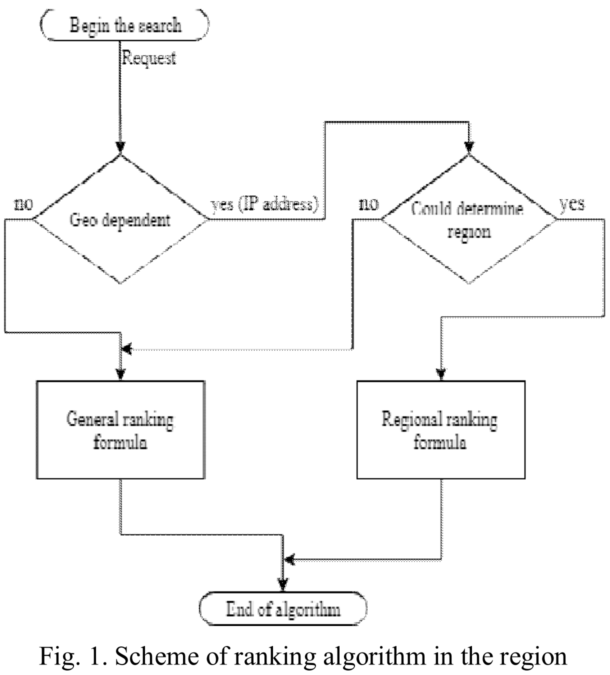

Fig. 1. Scheme of ranking algorithm in the region

Ranking of sites in general is expressed by dependence [2]:

$$R = V(2T + U + 2S), \tag{1}$$

where V is variety of information, T is response time of information resource, U is fullness of information, S is relevance of information.

Search systems can change algorithms of ranking. It can be as a small change of current weights of the factors as submission of fundamental changes in the ranking formula (1).

How it is that the new ranking algorithm affects search engine optimization?

The most significant changes are observed only for phrases that do not have a sufficient number of relevant responses, from a classical perspective. This implies that ranking of the frequency queues, for which the majority of commercial projects are developed, is evolving minimum changes with input from a new group of factors of commercial projects. Therefore, usually it is not possible to get a relevant information.

The analysis the algorithms for constructing modern search engines shows that search engines usually provide search results without taking into account the interests of users [3].

The purpose of the article is to describe the proposed method of dynamically designing a search engine based on the analysis of user search queries and automated user's document generation in order to increase the efficiency and speed of the search process.

III. Review of Existing Solutions

The significant contribution to the development of theoretical and applied issues of increasing the effectiveness of information search was made by G. Selton, D. Solton, Y. F. Skorokhodko, L. Y. Pshenichna, V. V. Sidorenko, V. M. Driyanskiy, O. G. Dubinskiy, Y. V. Rogushina. Developed methods of adaptive information retrieval, such as relevance feedback and modalities of user queries, do not sufficiently take into account the specifics of information retrieval on the Internet. Such search is characterized by low cost of communication, decentralization, heterogeneity and diversity of information resources, as well as the reluctance of users to spend time and effort on the use of methods to improve information retrieval.

IV. Features of the Proposed Method

After analyzing the methods of operation of search services, the following problems were found.

A. The accumulation of "shallow" and outdated information on the Internet.

For example, forums where people discuss a problem but no specific solutions propose. You have to scroll through and read all messages in order to capture the essence of the needed information, and when you have found the link to the requested site, you have noticed that the site is already closed or doesn't work at all.

B. Raising the rating of popular subjects.

There is a growing problem on the horizon. With the release of the complicated movie's title, it starts to get on front pages of search. You have to craft a search query to see exactly text articles on this topic.

C. Resources and services with small text information.

These resources include sites for photographers, illustrators, designers, or, for example, specific services that have useful features but do not have good texts on their pages to be found by search engines. For the most part such sites is found by chance.

D. Earnings on copywriting.

The level of SEO optimizers grows with each passing year, as well as their number, and along with it the number of sites for search services increases in order to earn money on contextual advertising. Once in a while, you open up new sites and see very similar articles with transposed words. It becomes clear that the situation with each passing year will only deteriorate. To solve these problems, you first need to figure out how the search engine is running.

E. Analysis of words and expressions.

Large companies use a lot of computing resources to analyze words from context. For example in such queries:

"Which machines are the fastest?"

"How to setup a machine for the spin?"

"Which machines are suitable for a cluster system?"

The same repeating word "machine", but this word carries a different information value, which depends on the context of the search query.

On the one hand, this approach is correct, but to save on computational resources and improve adaptation, the theory recently invented by psycholinguists called the effect of priming is used. The essence of the effect is that the preceding stimulus affects the performance of the subsequent cognitive action.

F. The existence of priming effects has been confirmed experimentally.

Without realizing it, constantly is copied what is heard. According to these data, it is not necessary to have large computational resources to analyze a word in context, it is necessary to understand the tendency of the search query and look for a similar informational meaning.

G. Ranking relevant pages.

The problem with the ranking of search pages is ranking on the web page of the web of resources and not the information itself, so the problems listed above arise.

In order to solve this problem it is necessary to refuse the ranking of sites and transfer to a new model of information ranking.

Proposed method includes five stages such as

Stage 1. Analysis words and expressions.

Stage 2. Selection of relevant search engines.

Stage 3. Ranking search engines.

Stage 4. PRF method (Pseudo relevance feedback).

Stage 5. Showing the most needed search engine.

There is an intermediate stage in the search for relevant information using the method PRF, the essence of which is written down in the features of the method.

Features of the proposed method contain an individual approach to search, which allows to increase the cost of the search. The essence of the method is creating of service for designing own search engine and replacing the algorithm of ranking on the algorithm for sorting search engines by category using the method PRF [4].

The first step is creating service that could design a new search engine for a specific category and fill it with information. The information is filled in with a system of successive objects, which is a list.

Objects contain key information, these keys allows configure the information work and increase the relevance of the search. In order to set up work with information, you need to configure the types of keys. There are several types of keys: a string, a number, a logical expression, and a link to another object in the list. There are two different ways of working with information (Fig. 2).

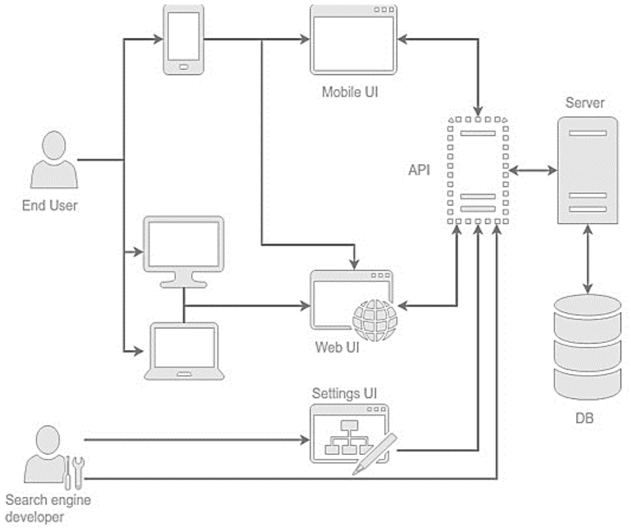

Fig. 2. Interaction of a user with the system

The first way is to configure using the graphics wrappers of the service, namely the choice of the key information, the way to equalize the information and the choice of the main key, which will be searched for information.

The second way is to use the software JS code and developed own library, which will provide methods for working with the information in the created search service.

The obligatory first phase of this development is the security settings of the code written by the developer of search engines using the principle of Kirchhoff [5].

The code is sent to the server, where it undergoes an analysis for optimization, finding the Cross-Site Request Forgery (CSRF) and the complexity of the algorithm. Then afterward this code is used in the search engine.

The service contains graphical tools for updating, storing and deleting information in the search engine. An analysis of the configuration of the search service is performed to increase the end-user search cost. There is a conclusion of the results, which indicates the disadvantages of the search engine to the developer of this system, and it generates documentation for its use based on the settings for the end user.

The next step is an algorithm for sorting search engines by category using the PRF method. The PRF method consisted of conducting a two-step search. In the first step for each end-user query is calculated the value of Score – the index of the relevance of the search engine to the query on which sorting is performed. An adapted model was chosen to calculate the Score. Interest is the question of which components should be added to the formula for calculating the Score.

In the analysis of search engines, the following components were selected: matching words from a query on a search engine W_{single}, matching pairs of words from the query W_{pair}, 100% match the text of the request W_{Phrase}. In addition, there is a component that gives priority to the presence of all the words in the search engine query $W_{AllWords}$. The next step is to hold iterations to calculate the *Score*:

$$Score = \frac{W_{Single} + W_{pair} + \frac{1}{k} W_{AllWords} + kW_{Phrase} + W_{PRF}}{k}. \qquad (2)$$

All these components are taken into account using the rating vectors under the name rating vocabulary, specific search service. The rating vocabulary RV has the form:

$$RV = \left[SW_0, SW_1, ..., SW_{countWords}\right]. \qquad (3)$$

In formula (3), *SW* is a score of a certain word, through which the search service was found, and *countWords* is the number of words in the rating vocabulary.

In turn, the rating vocabulary in the search service is taken into account dynamically and under next criteria.

1) If there is no word in the rating vocabulary, then all the words of the query in which the search service was selected are added to the rating vocabulary with an initial score of 1.

2) If there is no word in the search engine's rating vocabulary, then in order for it to get there you need to fulfill the condition (4).

$$Score_{wordQ} \geq Average_{RV}, \qquad (4)$$

where $Average_{RV}$ is expressed by the following formula:

$$Average_{RV} = \frac{1}{countWords}\sum_{i=0}^{countWords} SW_i, \qquad (5)$$

where, *SW* and *countWords* is expressed by the formula (3).

SW_Q is a vector score of all the words that were in the query to the search service, looks like this:

$$SW_Q, = \left[ScoreQ_0, ScoreQ_1, ..., ScoreQ_{countQWords}\right] \qquad (6)$$

in turn $ScoreQ_i$ is calculated by the formula:

$$ScoreQ = \left[\frac{ScoreQ_{lastmonth}}{Range_{last}}\right] Range_{last} + \frac{1}{2^4} Range_{quarter} + \frac{1}{2^7} Range_{year}, \qquad (7)$$

where *Range* is the amount of word use in requests for a certain period of time, in this case $Range_{last}$ thismonth, $Range_{quarter}$ the last 4 months, $Range_{year}$ the last 12 months, and $ScoreQ_{lastmonth}$ is the rating of the word for the previous month.

With the aid of these conditions, word gets to the rating vocabulary of search service, it is necessary to throw away the words from the vocabulary that can get there mistakenly, or which have already lost its relevance to the search of this service. Therefore, the word remains in the vocabulary if the following conditions are met:

$$\begin{cases} Range_{SWMounth} \geq Average_{RV}, \\ Range_{ScoreWord} > 0, \end{cases} \qquad (8)$$

where $Range_{SWMounth}$ is score of the word for the last month and $Range_{ScoreWord}$ is for the last 4 months.

Components of W_{Single} is vector W_{vector} the size of which is equal to the size of the rating vocabulary from the presentation (3). Looks like:

$$W_{vector} = \left[SingleW_0, SingleW_1, ..., SingleW_{countWords}\right]. \qquad (9)$$

Built according to the following rules:

$$\begin{cases} SW, \forall SingleW \in RV : SingleW \in W_{vector}, \\ 0, \quad SingleW \in W_{vector}, \end{cases} \qquad (10)$$

then values W_{Single} are of the form of:

$$W_{Single} = \sum_{i=0}^{countWords} SW_i \cdot SingleW_i. \qquad (11)$$

In the same way, other components are taken into account. In general, the amount of information found in search engines depends on the size of the end-user screen. The sorting system analyzes the end-user request and provides it with the information that is sorted by relevance. If no relevant information was found, the service will suggest creating a new search engine for the selected category. In order to improve the search efficiency of the end-user and reduce the time of creating a search service, the analysis of the settings made by the developer of the search service is committed. Based on this analysis, the developer of the search service can see the disadvantages of his search engine, and correct them in a timely manner.

V. CONCLUSIONS

The proposed method allows to increase the search cost by replacing the ranking algorithm with the search engine sorting algorithm by categories.

The information is filled in with a system of successive objects, which is a list. Objects contain key information, these keys allow to set up work with information and increase the relevance of the search. An analysis of the configuration of the search service is performed to increase the end-user search cost. The sorting system analyzes the end-user request and provides the information it finds, sorted by relevancy. The developed method provides acceleration of the algorithm of the search engine and the quality of the information found for the needs of the user. The process automation occurs at the stage of creating documentation about its use. The system will generate documentation based on the settings made by the developer of the search service automatically. According to the results of the testing, the proposed method can increase the search speed by an average of 12% compared to existing methods.

REFERENCES

[1] B. J. Jansen, A. Spink, J. Bateman, and T. Saracevic, "Real life information retrieval: A study of user queries on the Web," *ACM SIGIR Forum*. 2003, pp. 5–17.

[2] S. Gauch, J. Chaffee, and Pretschner, "Personalized Ranking Algorithm Based on User Interest Modeling," *International Conference on Computer Science and Technology rs*. SBN: 978-1-60595-461-5, 2017, pp. 648–655.

[3] Pant Gautam and R. Olivia, "Interest-Based Personalized Search," *The University of Utah*. pp. 14–35, 2007.

[4] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, "Introduction to Information Retrieval," *Cambridge University Press*. pp. 151–175, 2008.

[5] Kerckhoff's principle [Online] https://artofproblemsolving.com/community/c1671h1005760_kerckhoffs_principle

Received September 26, 2018

Kolomiets Ivan. Student.

Department of Computer Systems Software, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine.

Educaton: National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine (2019).

Research area: programming, designing search engines.

Publications: 2.

E-mail: klolo966@gmail.com, +380 (63) 478-94-22.

Oleshchenko Liubov. Candidate of Science (Engineering). Associate Professor.

Department of Computer Systems Software, National Technical University of Ukraine "Igor Sikorsky Kyiv polytechnic institute," Kyiv, Ukraine.

Education: Taras Shevchenko Chernihiv State Pedagogical University, Chernihiv, Ukraine, (2008).

Research area: information technology in transport systems, mathematical modeling, computer networks, programming.

Publications: 38.

E-mail: oleshchenkoliubov@gmail.com, +380 (97) 140-55-90.

І. В. Коломієць, Л. М. Олещенко. Метод динамічного проектування пошукового сервісу на основі автоматизованого аналізу пошукових запитів користувача

У статті розглянуто спосіб динамічного проектування пошукової системи на основі аналізу пошукових запитів користувачів та автоматизованого формування документації користувача. Запропоновано удосконалений метод динамічного пошуку з урахуванням індивідуальних особливостей користувача, який дозволяє збільшити вартість пошуку. Запропонований сервіс для проектування власної пошукової системи замінює алгоритм ранжування на алгоритм сортування пошукових систем за категоріями з використанням методу PRF. Заповнення інформації відбувається за допомогою системи послідовних об'єктів, що являє собою список. Об'єкти містять інформацію за ключами, які дозволяють налаштувати роботу з інформацією та збільшити релевантність пошуку. Сервіс містить графічні інструменти для оновлення, зберігання та видалення інформації у пошуковій системі. Для збільшення вартості пошуку кінцевого користувача здійснюється аналіз налаштування пошукового сервісу. Відбувається виведення результатів, у якому вказуються недоліки пошукової системи розробнику даної системи та генерується документація щодо її використання на основі налаштувань кінцевому користувачеві. Система сортування аналізує запит кінцевого користувача і надає інформацію, яку він знаходить, відсортовану за релевантністю. Розроблений метод забезпечує прискорення алгоритму пошукової системи та якість знайденої інформації для користувача.

Ключові слова: пошукова система; релевантність інформації; алгоритм ранжування; вартість пошуку.

Коломієць Іван Валерійович. Студент.

Кафедра програмного забезпечення комп'ютерних систем, Національний технічний університет України «КПІ імені Ігоря Сікорського», Київ, Україна.

Освіта: Національний технічний університет України «КПІ імені Ігоря Сікорського», Київ, Україна, (2019).

Напрям наукової діяльності: програмування, розробка пошукових систем.

Кількість публікацій: 2.

E-mail: klolo966@gmail.com, +380 (63) 478-94-22.

Олещенко Любов Михайлівна. Кандидат технічних наук. Доцент.
Кафедра програмного забезпечення комп’ютерних систем, Національний технічний університет України «КПІ імені Ігоря Сікорського», Київ, Україна.
Освіта: Чернігівський державний педагогічний університет ім. Т. Г. Шевченка, Чернігів, Україна, (2008).
Напрям наукової діяльності: інформаційні технології в транспортних системах, математичне моделювання, комп’ютерні мережі, програмування.
Кількість публікацій: 38.
E-mail: oleshchenkoliubov@gmail.com, +380 (97) 140-55-90.

И. В. Коломиец, Л. М. Олещенко. Метод динамического проектирования поискового сервиса на основе на основе автоматизированного анализа поисковых запросов пользователя

В статье рассмотрен способ динамического проектирования поисковой системы на основе анализа поисковых запросов пользователей и автоматизированного генерирования документации пользователя. Предложен усовершенствованный метод динамического поиска с учетом индивидуальных особенностей пользователя, который позволяет увеличить стоимость поиска. Предложенный сервис для проектирования собственной поисковой системы заменяет алгоритм ранжирования на алгоритм сортировки поисковых систем по категориям с использованием метода PRF. Заполнение информации происходит с помощью системы последовательных объектов, представляет собой список. Объекты содержат информацию за ключами, которые позволяют настроить работу с информацией и повысить релевантность поиска. Сервис содержит графические инструменты для обновления, хранения и удаления информации в поисковой системе. Для увеличения стоимости поиска конечного пользователя осуществляется анализ настройки поискового сервиса. Происходит вывод результатов, в котором указываются недостатки поисковой системы разработчику данной системы и генерируется документация по ее использованию на основе настроек конечному пользователю. Система сортировки анализирует запрос конечного пользователя и предоставляет найденную информацию, отсортированную по релевантности. Разработанный метод обеспечивает ускорение алгоритма поисковой системы и качество найденной информации для пользователя.

Ключевые слова: поисковая система; релевантность информации; алгоритм ранжирования; стоимость поиска.

Коломиец Иван Валерьевич. Студент.
Кафедра программного обеспечения компьютерных систем, Национальный технический университет Украины «Киевский политехнический институт им. И. Сикорского», Киев, Украина.
Образование: Национальный технический университет Украины «КПИ им. Игоря Сикорского», (2019).
Направление научной деятельности: программирование, разработка поисковых систем.
Количество публикаций: 2.
E-mail: klolo966@gmail.com, +380 (63) 478-94-22.

Олещенко Любовь Михайловна. Кандидат технических наук. Доцент.
Кафедра программного обеспечения компьютерных систем, Национальный технический университет Украины «КПИ им. Игоря Сикорского», Киев, Украина.
Образование: Черниговский государственный педагогический университет им. Т. Г. Шевченко, Чернигов, Украина, (2008).
Направление научной деятельности: информационные технологии в транспортных системах, математическое моделирование, компьютерные сети, программирование.
Количество публикаций: 38.
E-mail: oleshchenkoliubov@gmail.com, +380 (97) 140-55-90.