Sergey N. Lysenko[1], Yuri A.Khalin[2] DOI: 10.25045/jpit.v11.i1.06
Federal state budgetary educational institution of higher professional education «South West Statement University», Kursk, Russia
[1]dj3gmix@mail.ru, [2]yur-khalin@yandex.ru

IMPLEMENTATION AND MODIFICATION OF THE SCHEME BM 25 USING GENETIC ALGORITHM

Received: 26.01.2019 Revised: 29.03.2019 Accepted: 24.06.2019

The article describes the technologies of genetic algorithms for searching information on the Internet. The amount of electronic information is growing at a tremendous pace. In such situation, the need for data retrieval and analysis systems has sharply increased, and the demand for the intellectualization of information retrieval systems has risen. Currently, there are many types of models and algorithms designed to solve different searching and processing information tasks. Each algorithm has its advantages and disadvantages. Therefore, it is important to choose the algorithm which is the most appropriate for achieving a specific purpose. This article focuses on the information concerning the modification of BM25 schemes by means of genetic algorithms. The experiment are conducted and reveal that the modification based on the genetic algorithm represents a significant improvement in an original model by obtaining more proper solutions. The results of this work allow to broaden the application field of the system and improve the accuracy and quality of information retrieval on the Internet.

Keywords: *genetic algorithms, mutation, adaptation, relevance, crossing.*

1. Introduction

With a huge development pace, information space of computers creates new necessities in processing, introducing, storage and especially in search of information. The criterion of relevance is put on the first place which allows to improve and increase the efficiency of information searching. Actually, there is a rather huge number of schemes and models for solving purpose, and the BM25 is one of them.

BM25 and its update modifications (for example, BM25F) represent TF-IDF-like ranking functions. TF-IDF (from eng. TF — term frequency, IDF — inverse document frequency) — is a static argument which is used to estimate the word importance at the text context (being the part of the document collection). The weight of some words is proportional to the number of times it appears in the document and inversely proportional to the usage frequency of this word in other documents collection [1].

2. Scheme bm25f

For example, if there are one hundred words in the document and the word “tree” is found 5 times, then the frequency of the word “tree” in the document will be 0,05 (5/100). The document frequency is determined as the number of documents, which includes the word “tree”, is divided by the number of all documents, that is if the word “tree” is found in 1000 documents out of 10 000 000 documents, then DF will be equal 0,0001 ($1000/10000000$). For the final calculation of word weight, TF must be divided by DF. In our example, TF-IDF weight for the word “tree” will be equal to 500 ($0{,}05/0{,}0001$).

The scheme of weighting Okapi BM25 is developed as a method of building a probabilistic model, which was sensitive to the term frequency and the document length, but it did not use a huge number of additional parameters [1]. According to it, every document d gets an estimate on the request q, which is defined as the next formula:

$$Score_{d,q} = \sum_{t \in q} w_{q,t} \times w_{d,t} \quad (1)$$

where,

$$w_{q,t} = \ln\left(\frac{N - f_t + 0{,}5}{tf_{q,t} + 0{,}5}\right) \times tf_{q,t}$$

$$w_{q,t} = \frac{(k_1 + 1) f_{d,t}}{k_d + tf_{d,t}}$$

$$k_d = k_1\left((1-b) + b\frac{w_d}{w_a}\right)$$

Argument — k_1 is a positive parameter setting, which helps with the setting of term frequency. Argument b — is the second setting parameter ($0 \le b \le 1$), this parameter determines the numbering based on the document length.

Recommended values for k_1 and b are 1.2 and 0.75 accordingly; w_d and w_a – are the length of the document and the average document length.

For the selection of superstructure parameters, we will use next genetic algorithm, which gets number of coefficients (n) on enter, which are used at the model, and returns selected coefficients. The general algorithm will be as follows:

1) The initial population creation. The coefficients are chosen randomly ranging from $C_{\min}$ to $C_{\max}$ (individual range installs for every algorithm), then it is necessary to select k_n numbers of coefficients and to transfer them to binary view.

2) After it, we calculate the chromosome adaptation. Then we need to measure the mistake for every number of coefficients.

3) Moreover, it is important to choose two parents with the least mistake for crossing operation and to make the chromosome selection for mutation operation.

4) We calculate the adaptation of the new coefficient number.

5) if the mistake of 1n – number is larger than assigned mistake E_{enter}, then go to point 3, if not, then to point 7.

6) The resulting set, which minimizes the mistake, returns to the current searching model [2].

Review deeper basic aspects:

- all coefficients are generated originally randomly based on uniform distribution law with upper and lower restrictions.

- then these coefficients are transformed to binary view to apply following crossing and mutation operations.

The mistake is estimated with the help of the formula:

$$\varepsilon = \sum_{i=0}^{n} (r(d_j, q_i) - score(d_j, q_i)^2 \qquad (2)$$

where, $r(d_j, q_i)$ is the average document estimation d_j of experts, for request $score(d_j, q_i)$ returned document relevance d_j for request q_i .

Optimal crossing and mutation operations are obtained in the results of experiments. After carrying out the amount of experiments, it turned out that for the fastest and the best result of target function the chromosome selection must be applied by the following method. It is necessary to choose two optimum chromosomes and to select randomly chromosomes N_{KR} [1–2].

For successful mutation operation we need to find two chromosomes with the lowest adaptation and N_{MUT} chromosomes.

For optimum crossing operation, several experiments are perfomed using different methods. Two optimum adaptive methods (figure 1) are determined in the results. A random selection from

1 to 100 is carried out to check the efficiency. As the parameter, the average relevance estimation from the request answers which determines the optimality taken [3–4].

In the results, the function reaches its maximum point with the help of crossing comb method with very close half method (figure 2). Based on different requests, Comb method reaches its maximum point by one number of requests, half method reaches this result by 2 request numbers and both methods reach this point by 4 numbers of requests.

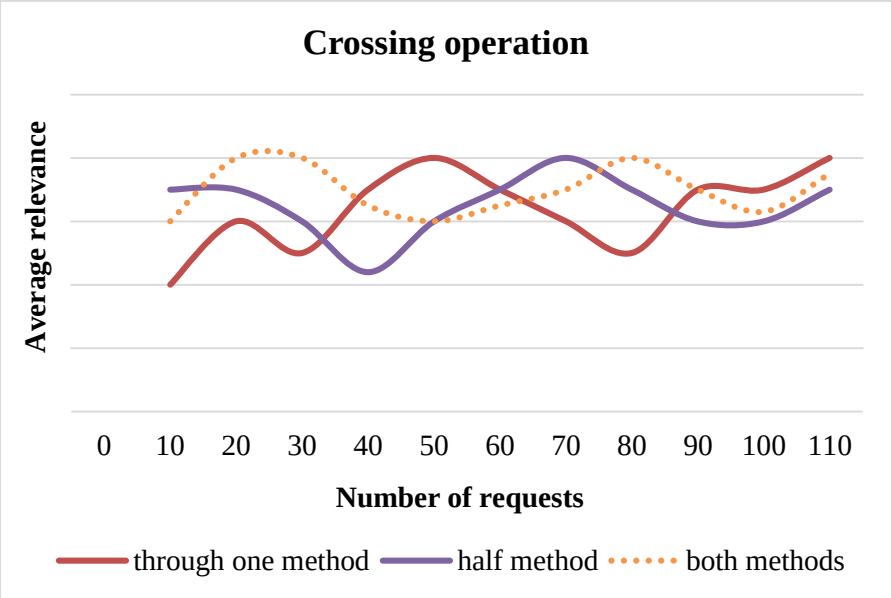

Figure 1. Crossing operation

When crossing «through one» bits from 2 coefficients change through one. When crossing with the help of «half method», it is selected half number of bits from the 1st coefficient and the second part from the 2nd coefficient.

Mutation operation. A number of experiments are performed, where the average document relevance is determined, which is given when all other mechanisms are disabled, to determine the optimum mutation. After concluding the experiment, it is noted, that mutation reaches its maximum point, if the probability is 40%. Dependence plot from mutation probability to searching results is shown on figure 2 [3].

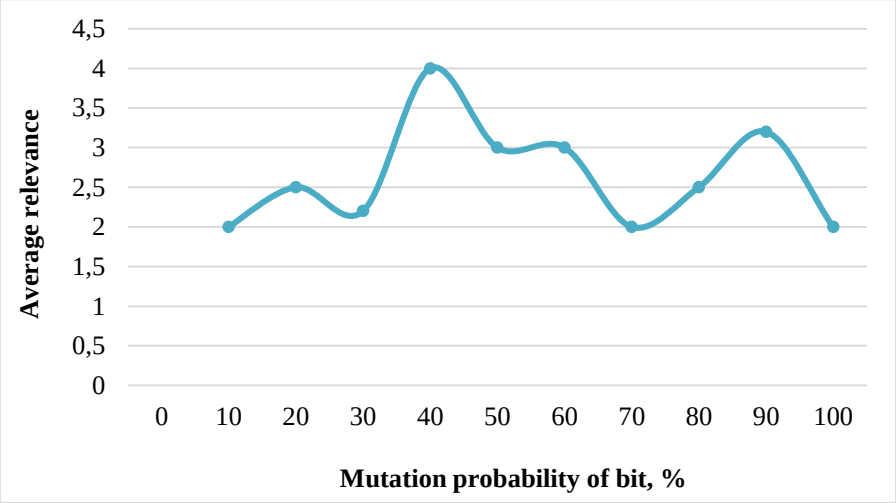

Figure 2. Dependency of mutation probability on search results

2 document bases – requests are created. The 1st basis is used to teach the algorithm, the 2nd one is used to estimate it, 2 collections are taken:

- pseudorandom sites selection from ucoz.ru domain with the document volume by 600 000,
- collection which includes news announcements from 20 resources and contains 3 time laps (about 33 500 documents).

3 kinds of requests are formalized:

- information requests,
- navigation requests,
- transaction requests.

Nearly 4 100 requests are created in equal ratio.

3. Experiment

The experiment consists of realizing the model OkapiBM25 and its modification where selected values by genetic algorithm are used as the superstructure parameter. Resulted metrics for 2 systems by 30 requests are compared.

Fullness is calculated as the ratio of searched relevance documents to general number of relevance documents.

Fullness characterizes the system's ability to find the user's needed documents, but not to count the number of irrelevant documents, which are given to the user. Fullness is shown on figure 3.

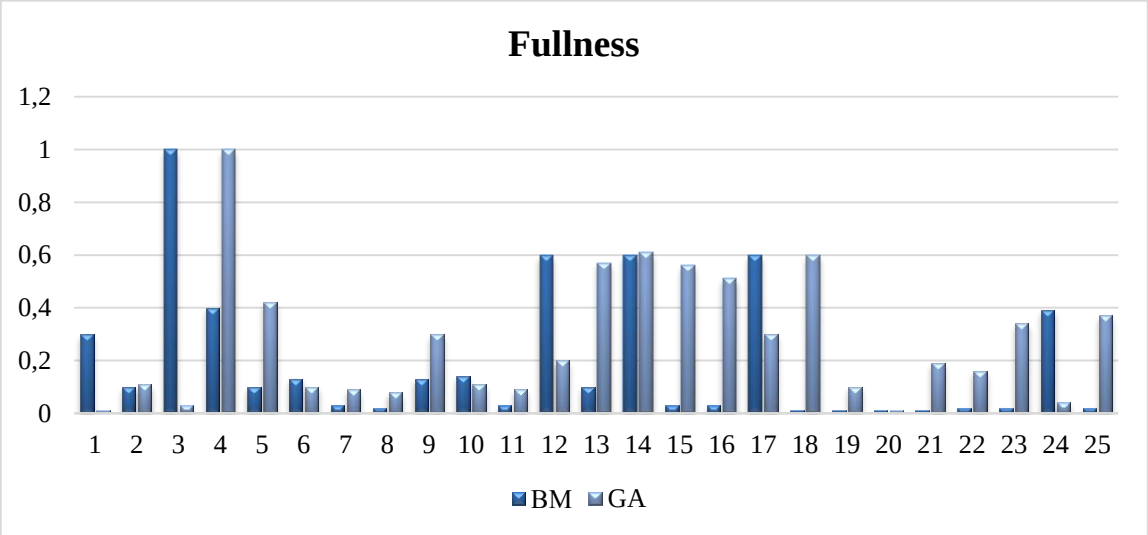

Figure 3. Fullness

Average fullness values are: BM=0,173, GA =0,241. GA shows better fullness, in the average by 38%, it means the user will get more relevant documents by 38%.

Accuracy is computed as the ratio of searched relevance documents to general number of documents. Accuracy describes the system ability to give only relevant documents in result list. Algorithm accuracy is shown on figure 4.

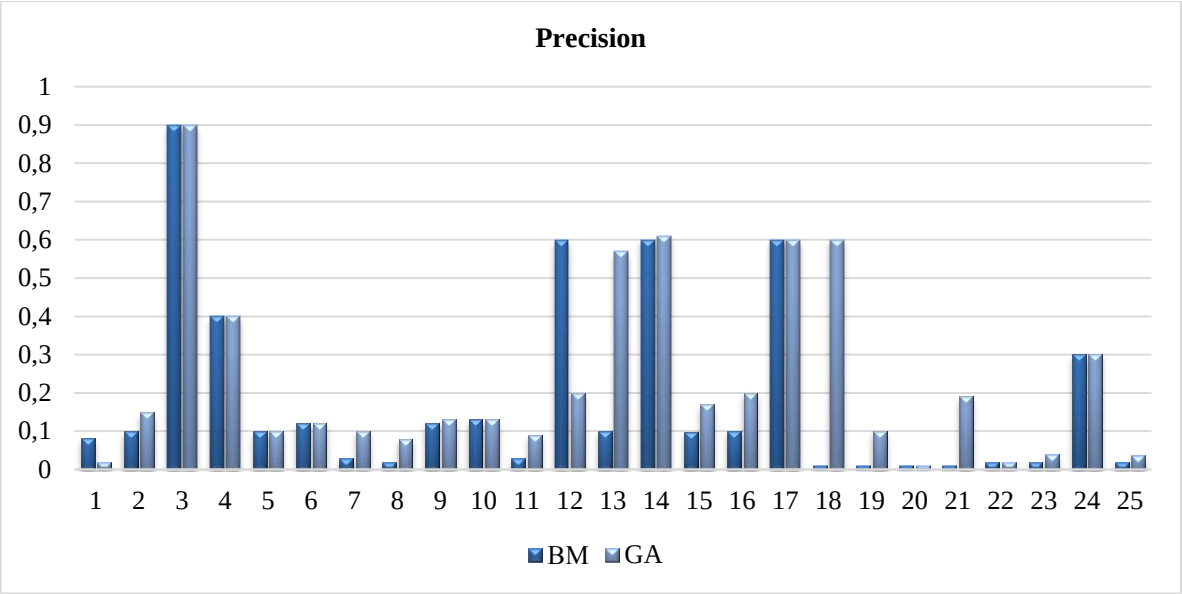

Figure 4. Precision

Average precision values are: BM=0,167, GA=0,217. GA performs higher precision by 30%, it means that there is higher probability that the user will get only relevant documents for his/her request.

The accuracy is computed as ratio of correct decisions to general number of decisions. The algorithm accuracy is providen in figure 5.

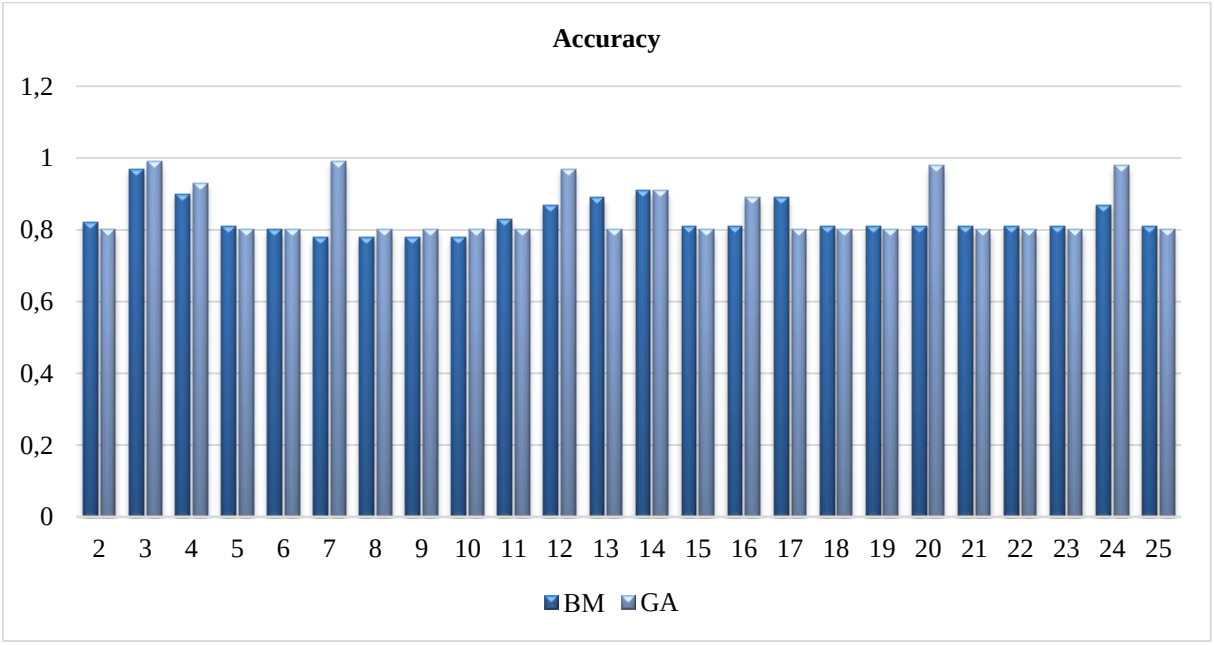

Figure 5. Accuracy

The average accuracy values are: BM=0,842, GA=0,853. GA posseses higher accuracy by 5%, the system makes more correct decisions.

The mistake is estimated as ratio of incorrect decisions to general number of decisions. The mistake is providen in figure 6.

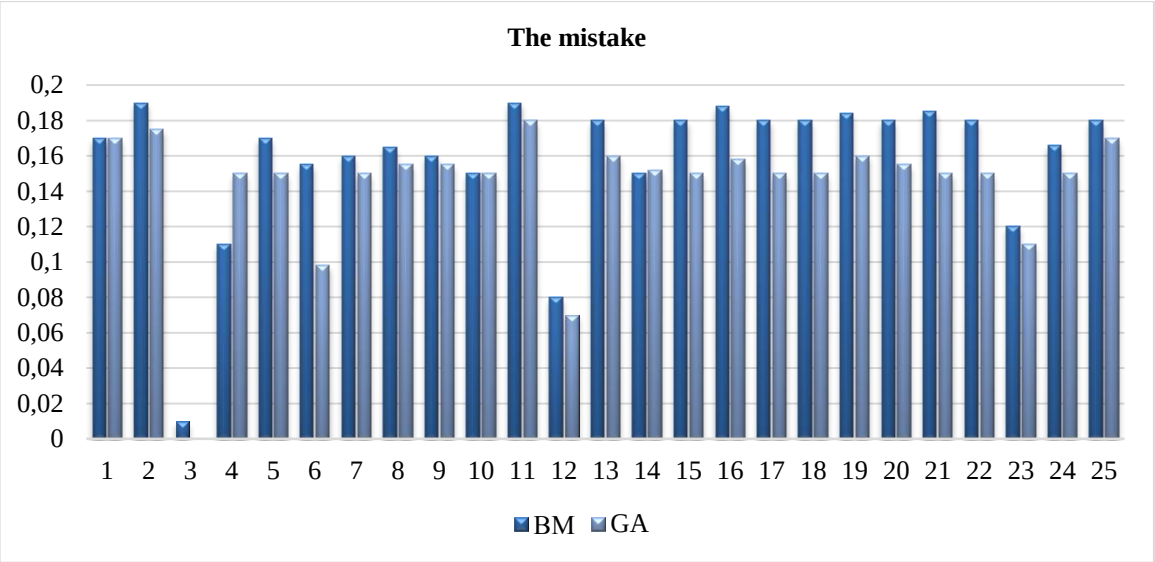

Figure 6. The mistake

The average mistake values are: BM=0,167, GA=0,150. GA possesses lower mistake by 10%, the system chooses less incorrect decisions by 10%.

F-measure (F) is often used as unified metric which unites the accuracy and fullness metrics into one metric [5]. F-measure is calculated with the help of the formula for current request:

$$F = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \tag{3}$$

Necessary flag base properties:

- $0 \le F \le 1$;
- if $recall = 0$ or $precision = 0$, then $F = 0$;
- if $recall = precision$, then $F = recall = precision$;
- $\min(reacall, precision) \le F \le r + \frac{r+p}{2}$.

F-measure of algorithms is shown in figure 7.

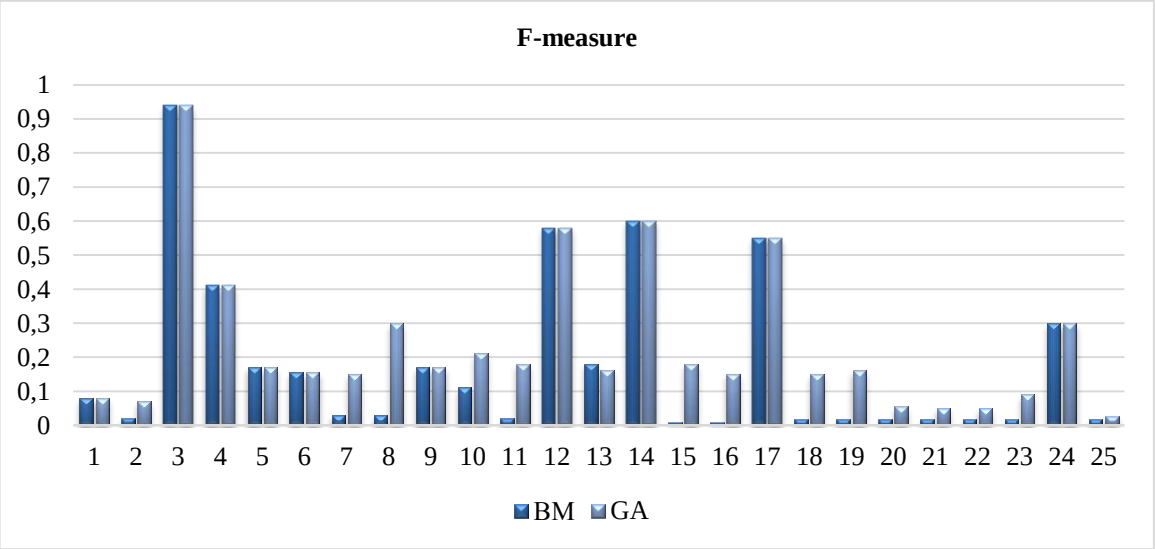

Figure 7. F-measure

The average f-measure value: BM=0,15, GA=0,23. GA allows to improve these metrics by 38%, this GA gives 38% better results in average.

Thus, the modification with genetic algorithm allows to improve the base model in average by 38%, the user will receive a response with more relevant documents by 38%, probability of only relevant responses to request is increasing by 30%, the system chooses more correct decisions by 5% and by 9% as incorrect.

Reported results of the work allow expanding the scope of system application and substantially increasing the accuracy, quality and speed of searching information in the Internet.

4. Conclusion

The article reviewed and proposed the modifications to the methods for calculating the relevance of a document using a genetic algorithm that allowed for improving information retrieval methods. A modification with a genetic algorithm had better metric values compared to the basic algorithm.

As a result, the system made more correct decisions, approximately by 5%, while 9% less received incorrect decisions.

References

1. Horoshko, M.B. Algoritmy, ispol'zuemye v poiskovyh sistemah: materialy VII Mezhdunar. nauch.-prakt. konf., g. Novocherkassk, 2009, 242–250 s.
2. Manning, Kristofer D. Vvedenie v informacionnyj poisk. //M: Vil'jams, 2011, 528 s.
3. Umbarkar A.J. and Sheth P.D., Crossover operators in genetic algorithms: a review. ICTACT Journalon Soft Computing, 6(1), 2015, pp. 1083–1092.
4. Arora P.K, Haleem A, Singh M.K, Kumar H. "Optimization of Cellular Manufacturing Systems using Genetic Algorithm: A Review", Advanced Material Research Journal, 2013, vol. 622, pp. 60–63.
5. Eremeev A.V. Geneticheskie algoritmy i optimizacija uchebnoe posobie, Omsk: Izd-vo Om. gos. un-ta, 2008, 48 s.
6. Kurejchik V.V., Sorokoletov P.V., Habarova I.V. Dinamicheskie geneticheskie algoritmy v sistemah podderzhki, prinjatija reshenij, Taganrog: Izd-vo TRTU, 2006, 51 c.

UOT 004.021

Lısenko Sergey N.[1], Xalin Yuriy A.[2]
Cənub-Qərb Dövlət Universitetinin Federal Dövlət Büdcəli Ali Təhsil Müəssisəsi, Kursk, Rusiya
[1]dj3gmix@mail.ru, [2]yur-khalin@yandex.ru

Genetik algoritmin istifadəsi ilə BM 25 sxeminin realizasiya və modifikasiyası

Məqalədə İnternet şəbəkəsində informasiya axtarışının həyata keçirilməsi üçün genetik alqoritm texnologiyaları təsvir edilmişdir. Elektron informasiyanın həcmi böyük sürətlə artır. Bu halda axtarış sistemlərinə və verilənlərin analizinə, həmçinin məlumat-axtarış sistemlərinin intellektuallaşdırılmasına olan tələbat kəskin şəkildə artmışdır.

Hazırda informasiya axtarışı və emalı sahəsində məsələlərin həlli üçün çoxsaylı model və alqoritmlər mövcuddur. Hər bir alqoritm isə üstün cəhətlərə və çatışmazlıqlara malikdir. Bu səbəbdən də konkret məsələnin həlli üçün ən çox uyğun gələn alqoritmin seçilməsi böyük əhəmiyyətə malikdir. Bu məqalədə genetik alqoritmlərin köməyi ilə BM25 sxeminin modifikasiyasına baxılmışdır.

Bu məqsədlə sınaqlar keçirilmişdir. Aparılan sınaqlar onu göstərmişdir ki, genetik alqoritmin tətbiqi ilə həyata keçirilən modifikasiya daha düzgün həllərin alınması hesabına ilkin modelin əhəmiyyətli dərəcədə təkmilləşdirilməsinə imkan vermişdir. Aparılan tədqiqatın nəticələri sistemin tətbiq sahəsinin genişləndirilməsinə və İnternet şəbəkəsində informasiya axtarışının dəqiqliyinin və keyfiyyətinin artırılmasına imkan verir.

Açar sözlər: *genetik alqoritmlər, mutasiya, uyğunlaşma, relevantlıq, calaq.*

УДК 004.021

Лысенко Сергей Н.[1], Халин Юрий А.[2]
Федеральное государственное бюджетное образовательное учреждение высшего образования <<Юго-Западный государственный университет >>
[1]dj3gmix@mail.ru, [2]yur-khalin@yandex.ru

Реализация и модификация схемы BM 25 с использованием генетического алгоритма

В статье описаны технологии генетических алгоритмов для поиска информации в сети Интернет. Объем информации в электронном виде растет огромными темпами. В такой ситуации резко выросла потребность в системах поиска и анализа данных, а также возник спрос на интеллектуализацию информационно-поисковых систем. На сегодняшний день имеется огромное количество моделей и алгоритмов для решения задач в области поиска и обработки информации. У каждого алгоритма есть свои преимущества и недостатки. Поэтому важно выбрать тот алгоритм, который лучше всего подходит для решения конкретной задачи. В данной статье рассмотрена модификация схемы BM25 с помощью генетических алгоритмов. Был проведен эксперимент, в ходе которого выяснилось, что модификация с генетическим алгоритмом позволяет существенно улучшить исходную модель за счет получения более правильных решений. Результаты данной работы позволяют расширить область применения системы и повысить точность, качество поиска информации в сети Интернет.

Ключевые слова: *генетические алгоритмы, мутация, адаптация, релевантность, скрещивание.*