DOI: 10.14529/cmse220404

DEVELOPING INTELLIGENT ASSISTANTS TO SEARCH FOR CONTENT ON WEBSITES OF A CERTAIN GENRE*

© 2022 V.D. Rublev, E.A. Sidorova

A.P. Ershov Institute of Informatics Systems,
Siberian Branch of the Russian Academy of Sciences
(Lavrentieva Avenue 6, Novosibirsk, 630090 Russia)
E-mail: rubleffvlad@gmail.com, lsidorova@iis.nsk.su

Received: 06.11.2022

This paper discusses an approach to automatic generation of intelligent assistants, which provide information search on the content of a website. A feature of the approach is to use genre models, developed for a given type of resource (educational, informational, etc.), on the basis of which the genre structuring and subsequent thematic clustering of the content of the target website is performed. The resulting genre structures allow us to define more precisely the boundaries of thematic clusters related to the topic of the user's search query. The search quality evaluation for the Russian-language websites showed an F-score of 87.8% and originality of 80.9%, which exceeds the Yandex search engine results by 1.1% and 9.1%, respectively. In order to predict user information needs, a method for refining the resulting sample is proposed. It allows a user to get information implicitly, based on current and previous queries, about what the user was not satisfied with in the previous search results. A model of user's search intentions has been developed and its computational component includes a method for evaluating query closeness based on the FRiS function. Based on the proposed methods, a chatbot was created on the Telegram messenger platform to search the websites of educational institutions. The experiments showed that the user needs the average of 1.75 qualifying questions to find the necessary information.

Keywords: information retrieval, intelligent assistant, website genre model, thematic analysis, information retrieval system, user search intent model.

FOR CITATION

Rublev V.D., Sidorova E.A. Developing Intelligent Assistants to Search for Content on Websites of a Certain Genre. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2022. Vol. 11, no. 4. P. 51–66. DOI: 10.14529/cmse220404.

Introduction

Rapid development of information technologies, in particular, the development of the Internet has generated a large amount of electronic information. The simplicity and accessibility of creating and distributing data has led to a huge flow of necessary and unnecessary information. Currently, information search is developing, since from a large amount of data it is necessary to find only the information in which the user is interested. One of the ways to improve the quality of search is to use methods of analyzing the content of Internet sources based on knowledge.

This paper proposes an approach that integrates different search methods based on cluster analysis, uses genre models, a model of user search intentions and relies on the Internet genre of the site. We consider genre characteristics for two levels of resource representation: the site as a whole (macro level) and the site page (micro level) [1]. When analyzing web genres, functional, formal (compositional and lexico-grammatical) and content aspects are usually considered, which correlate with different levels of physical representation (the site as a whole, the page, individual

*This article has been recommended for publication by the Program Committee of the International Conference on Data Analytics and Management in Data Intensive Domains – 2022.

components of pages) [2]. And to test the proposed approach, a chatbot is being implemented to search through the websites of educational institutions in Novosibirsk. To analyze and build a genre model of the site, and conduct experiments, a corpus of websites of educational institutions with a size of 9052 pages was compiled, which contains the content of 137 sites.

The article is organized as follows. Section 1 of this work provides an overview of related works, Section 2 offers a way to describe the genre model of educational institutions' websites, Section 3 describes cluster analysis for thematic clustering of genre fragments of the site, Section 4 presents a model of the user's search intentions, Section 5 presents the proposed architecture of the search engine, in Section 6, the implementation of a chatbot for information search is described, and in Section 7, experimental study of the quality of search and user information support on the corpus of educational organizations websites is presented. In Conclusion, a brief summary of the results obtained in the work is given, and directions for further research are indicated.

1. Related works

There are special search engines for searching, taking into account the deep analysis of the query. The functionality of these products varies, ranging from rearranging the search results of other search engines, and to full-text search of information in indexed texts for key user queries, taking into account morphological features, syntax and semantics of words. The systems can also perform searches taking into account synonyms and related words, which are grouped according to their semantic meaning. Thus, the Nigma system [3] is a meta-search engine that provides a search for textual information, taking into account the meaning of the query given in natural language. This service allows you to improve the quality of the search due to the fact that based on the user's entered query, the system generates a list of documents divided into several clusters so that the user can choose in which cluster to continue the search. To do this, a forming phrase is formed for each cluster and the number of documents in it is indicated. There are also search engines based on thematic cluster analysis, for example, Carrot2 [4]. This system offers two specialized clustering algorithms: Lingo, an algorithm based on singular value decomposition, and STC, the suffix tree method, a classic search results clustering algorithm that very quickly creates a flat cluster with an adequate description. The search in the Yippy system [5] is based on IBM Watson, a supercomputer that has natural language processing technologies and is able to analyze complex, unstructured data and even understand professional slang. Yippy has expanded the capabilities of IBM Watson and added features such as trend tracking, concept clustering, entity extraction, relevance monitoring and sentiment analysis. AskNet [6] consists of two subsystems that allow both the search for information on the Internet and the search for information on users' computers in the corporate network. The system differs from other search engines in that it provides not only links to documents and resources to the user's request, but also text information that is the answer to the user's question. Hakia [7] contains its own linguistic database, in which words are divided into various "meanings" that they convey. It extracts all possible queries related to the content (using its database), and they become paths to the source document. And then independently ranks the content based on additional analysis of offers. It also uses the authenticity and age of the content to determine relevance. Modern search engines such as Yandex and Google in recent years have done much to improve the Internet search, for example, using knowledge graphs [8] or neural network representation of queries and lyrics [9],

but in this work, we study local search sites of one functional genre, and not a global search across the Internet which focused on these systems.

Despite the fact that many of the above systems take into account the semantic features of the text, they do not use preliminary thematic clustering and do not take into account the genre features of indexed web resources. The Nigma and Hakia systems have stopped working. It should also be noted that most of these systems are focused on working with texts in English.

Information search methods are used in various applications today and one of the most popular applications is a messenger with a built-in intelligent assistant or chatbot. There are 6 main methods of building chatbots [10]: rule-based methods, search, generative approach, ensemble methods, grounded learning and interactive learning. Rule-based systems [11] are trained based on a predefined hierarchy of rules that determine how to convert user input into a response or action. Search-based methods [12] are used today in most chatbots. Such systems operate using directed graphs and are trained to provide the best possible answer from their database of predefined answers. Instead of using predefined answers, a conversational chatbot using generative methods [13] receives a large amount of data (real dialogs) and learns to generate a new dialog that is similar to them. Modern conversational chatbots that can talk on any topic were created using ensemble methods [14], which, depending on the context, use some combination of rule-based approaches, search and generative approach. An intelligent assistant using informed learning [15], analyzing a user-entered query, generates a neural network that is configured for this specific query and task. Such an intelligent assistant is better "grounded" due to its ability to learn and use representations of real-world knowledge. Interactive machine learning is algorithms and intelligent user interface structures that simplify machine learning through human interaction. This development allows computers to learn from people by interacting with them in natural language and observing them.

2. The model for presenting a site of a certain genre

In this approach, the search is based on preliminary indexing of the site based on its genre structure. Each site has certain features that are determined by the specifics of the field of activity and are formed due to the similarity of subject matter, composition and style, which corresponds to the classical definition of the speech genre formulated by M.M. Bakhtin [16]. Genre is a typical model of constructing a speech whole. A genre model representing its "typical reproducible genre form" can be defined for each type of site. Each genre model represents a general structure of sites inherent in this model. Thus, the content of each site can be divided into genre fragments representing some aspects of the content. To analyze and build a genre model of the site, it is necessary to have a corpus of texts of a given subject, therefore, a corpus of educational institutions with a size of 9052 pages of 137 sites was collected. Based on the analysis of this corpus, a genre model of the educational institution's website has been developed, the upper level of which is presented in Tab. 1.

Content (content part) of the site is a sequence of text blocks. To determine the genre of these blocks, a set of genre markers and a marker language are used, which allows you to specify terms, their combination or enumeration. To describe the genre model, the language proposed in [17] is used, which allows for the description of aspects of the content of any genre block based on genre markers. The genre model contains sets of markers, each set describes some aspect of the content, and for each genre typical aspects of the content are described and in which part

Table 1. The range of genres of text fragments, depending on the genre of the site

	Site genre: educational institution	
	Higher educational institution	**Secondary educational institution**
Type of the fragment	Description of the scientific institution	Admission to school
	Description of faculties	Parents
	Description of the campus	Teachers
	Incoming	The final essay
	Students	GIA
	Graduates	
	Employees	
	Graduate students	
	News Feed	
	Comment	
	Description of the event	
	Frequently Asked Questions	

of the HTML markup they are located. In Fig. 1, we show some examples of genre markers for describing the genre model using the above-mentioned markup language.

```
# Template for describing content aspects
Content aspect: ["Marker11"]["Marker21"]["Marker31"]
    ["Marker41", "Marker42"]

# Template for genre descriptions
"Genre ID": [<Content aspect 11, tag>]
    [<Content aspect 21, tag> < Content aspect 22, tag>]

# Examples of content aspects
For applicants: ["to enroll"]["admission committee"]["admission"]
    ["admission rules"]["entrant"]["admission campaign"]["rating list"]
       ["tuition fees"]
Education Levels: ["postgraduate"]["master's degree"]
    ["bachelor's degree"]["specialty"]

# Example of a page genre description
"For applicants": [<For applicants, text>]
    [<For applicants, all>< Education Levels, all>]
```

Fig. 1. Examples of genre markers for describing the genre model

The developed genre model is used to highlight genre fragments in the content of the website.

3. Thematic clustering of texts

After segmentation of the text content of the site, its thematic clustering is carried out — the allocation of thematic clusters, which will be searched in the future.

A dictionary is needed to cluster content and analyze a user's query, so a dictionary of terms was created in the KLAN subject vocabulary extraction system [18]. This system supports the

main stages of text analysis: syntactic, semantic and morphological. The dictionary, created on the basis of the corpus of websites of educational institutions, contains one-word and multi-word terms, its size is 17,500 terms for which statistics are collected. K-means algorithm was used [19], due to the fact that it has a high learning rate, and agglomerative hierarchical clustering [20], because it allows you to interpret the result well.

Each document is represented by a vector in the space R_n, where n is the dimension of the dictionary. To obtain such a representation, the statistical measure tf-idf [21] is used, which evaluates the importance of the term in the text, relative to other texts of the corpus. The distance between vectors can be calculated using various metrics, cosine similarity and Euclidean metric have been tested,

$$\rho\left(a,b\right)=\frac{\sum_{k=1}^{n}a_k\cdot b_k}{\sqrt{\sum_{k=1}^{n}a_k^2}\cdot\sqrt{\sum_{k=1}^{n}b_k^2}} \tag{1}$$

and

$$\rho\left(a,b\right)=\sqrt{\sum_{k=1}^{n}\left(a_k-b_k\right)^2} \tag{2}$$

respectively. The Euclidean metric works statistically worse on the collected corpus of texts, therefore, cosine similarity is used in this study.

Centroid (cluster center) is a vector whose elements represent the average values calculated for all proposals (presented in vector form) from the cluster. Thus, the centroid is a vector representation of the thematics of the proposals included in the cluster.

There are many metrics for assessing the quality of clustering and a large number of their comparisons have been carried out [22–24]. One of these metrics is the "Silhouette score" [25], which statistically significantly shows results better than other metrics. The Silhouette coefficient is calculated separately for each object from the cluster using the following formula:

$$S=\frac{b-a}{\max\left\{a,b\right\}}, \tag{3}$$

where a is the average distance from the selected object to the objects of its cluster, b — the average distance from the selected object to the objects of the nearest cluster (not containing the selected object). To assess the quality of clustering, the average value of the "Silhouette" coefficients for all clustered objects is calculated. With the help of this metric, the proposed clustering methods were evaluated and the results were ambiguous, since the outcome strongly depended on the input data.

4. The model of the user's search intentions

In order to improve the quality of user information support, a model of user search intentions has been developed, which allows recognizing intentions through message analysis.

Let us define a formal model of the user's search intentions at the current step of information search as a system of the form

MSI = <Q_{prev}, Q_{next}, Res_{prev}, F_{SS}, Tr, P>, where

Q_{prev} — previous search query,

Q_{next} — new search query,

Res_{prev} — previous search result,

Tr — request proximity threshold,

P = p_{new}, p_{expand}, p_{reduce}, p_{excexp} — multiple search engine states (search parameters): new, expand, narrow, exclude and expand.

F_{SS} — the function of calculating the states of the search engine

$$F_{SS} = Q_{next} \times Q_{prev} \times Res_{prev} \to P. \tag{4}$$

The proximity of queries is calculated using the Function of Rival Similarity (FRiS) in competition with the previous search result. Let p be a metric, then the FRiS function is calculated as follows [26]:

$$FRIS\left(Q_{next}, Q_{prev} \mid Res_{prev}\right) = \frac{\rho\left(Q_{next}, Res_{prev}\right) - \rho\left(Q_{next}, Q_{prev}\right)}{\rho\left(Q_{next}, Res_{prev}\right) + \rho\left(Q_{next}, Q_{prev}\right)}, \tag{5}$$

$$F_{SS}\left(Q_{next}, Q_{prev}, Res_{prev}\right) = \begin{cases} P_{new}, Q_{next} \bigcap Q_{prev} = \emptyset \\ P_{expand}, Q_{next} \bigcap Q_{prev} = Q_{next} \\ P_{reduce}, Q_{next} \bigcap Q_{prev} = Q_{prev} \vee \\ \vee FRiS\left(Q_{next}, Q_{prev} \mid Res_{prev}\right) \geq Tr \\ P_{excexp}, FRiS\left(Q_{next}, Q_{prev} \mid Res_{prev}\right) < Tr. \end{cases} \tag{6}$$

The main idea of the user's search intent model is that with a new search query, we can use information about the previous search results and the previous query, and determine whether this is a new search query or whether the user is not satisfied with something in the results and wants to refine the query. If the request is refined, we can understand the user's intentions (what exactly did not suit him in the request) and perform a search with some parameter that changes the search method. In the case when the model of the user's search intentions came to a state of reduce, then we assume that the user received the necessary information, but along with it he received a large amount of noise and it is necessary to reduce the search selection in order to get rid of the noise and get only the necessary information. In the case of an expand, we assume that the user did not find the necessary information and expand the search selection. Based on the proposed model, Fig. 2 presents an algorithm for analyzing the user's search intentions.

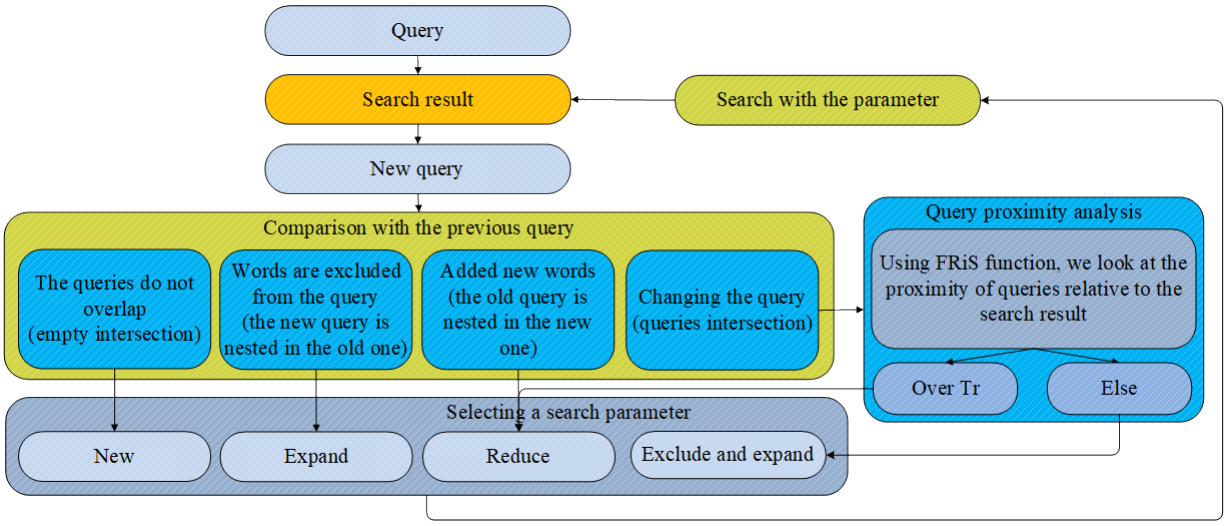

Fig. 2. The scheme of the analysis of the user's search intentions

After receiving the user's request, it is compared with the previous request in order to select one of the four parameters for repeated search. The parameters change the search method: exclusion of previous results, expansion of the sample, narrowing of the sample, and a new search.

To compare queries, lemmatization is performed and service words are excluded. Queries are compared in a set-theoretic way. When the queries do not contain the same lemmas (empty

intersection), a new search is performed without parameters. If the words were removed from the query (the new query is nested in the old one), then a search is performed with the "expand" parameter. If new words were added to the query (the old query is nested in the new one), then "reduce" parameter is used.

For example, when querying "Dorm" after the query "Is there a dorm at the university", the intersection of the queries ("Dorm") is equal to the new query, so a search will be performed with the "expand" parameter. But if the previous request is "Submission of documents", and the new one is "Deadlines for submission of documents", then a search is performed with the "reduce" parameter. In the case when there was a more serious change in the request (intersection of requests), then the proximity of requests is calculated. For example: "How to transfer from one faculty to another" and "How to transfer to another group in your course". The average value of the FRiS function (formula 5) is calculated for all search results and if the average is greater than the preset threshold, then reducing parameter is used, otherwise a search is performed with the exclusion and expansion parameter.

5. Intelligent assistant architecture

Figure 3 shows the architecture of the intelligent assistant, which includes three main modules: a preprocessing subsystem, a search engine subsystem, and a user interaction subsystem.

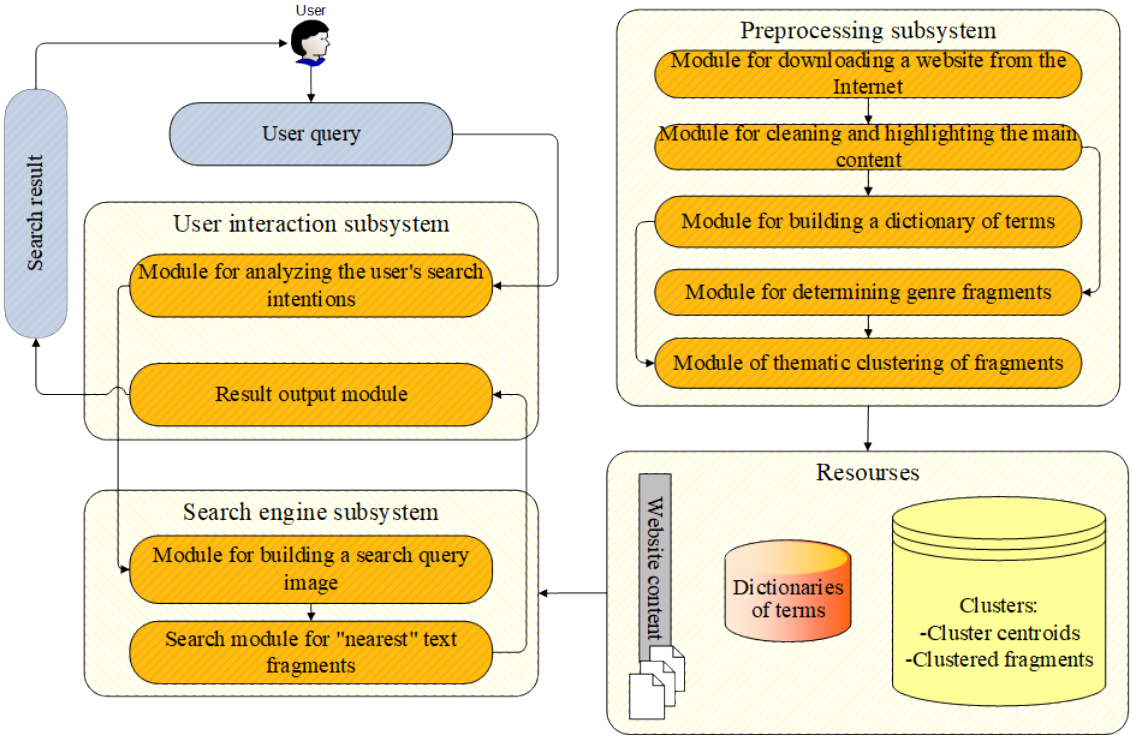

Fig. 3. Architecture of the intelligent assistant

The preliminary processing of the site is done in advance and consists of the following stages:

1. Downloading a website from the Internet — recursive search for all links to the pages of the site and sequential downloading of pages.
2. Cleaning and highlighting of the main content — on all pages of the site, data that is not the main text content is searched and deleted (header, drop-down menu, footer, etc.). All html tags are removed, except for the allowed ones: h1, h2, h3, h4, h5, h6, a, b, ul, ol, li.

3. Site dictionary generation — the received texts with the main content of the site are loaded into the dictionary system, where all terms are automatically located and statistics are collected for them. Lemmatization is performed, as well as the removal of stop words and words that are too rare.
4. Genre segmentation — based on the developed genre model, the presence of genre fragments on the page is determined, the site content is divided into fragments from title to title and the genres of these fragments are determined. Fragments of one genre following each other are combined into one, and fragments whose genre could not be determined are excluded from consideration.
5. Thematic clustering — one of the available clustering algorithms is selected, fragments obtained at the previous stage are taken as text and presented in vector form using the tf-idf measure, the number of clusters into which the text will be divided is determined, depending on the genre and volume of the fragment. Clustering is performed by the selected method for a given number of clusters. Clusters are formed within a single site.

The results of the site preprocessing are saved as the following resources: the clustering result, a text collection of content and a dictionary of site terms.

The search subsystem for each user request produces:

1) building a search query image;
2) search for the "nearest" thematic fragments taking into account the search parameter and their ranking.

The result of the search subsystem will be a set of arranged thematic fragments corresponding to the search query.

The user subsystem receives a user request for input and executes:

1) analysis of the user's search intentions — the parameter of repeated search or its absence (new search) is determined;
2) request to the search subsystem;
3) generating the search result and sending it to the user.

6. Intelligent assistants as messenger chatbots

The developed approach does not depend on messengers and can work with any API, but the Telegram messenger API was chosen for testing the approach because of its popularity, simplicity and free access to the API.

Telegram provides Telegram Bot API for writing chatbots [27]. Figure 4 shows the scheme of the intelligent assistant based on the Telegram messenger.

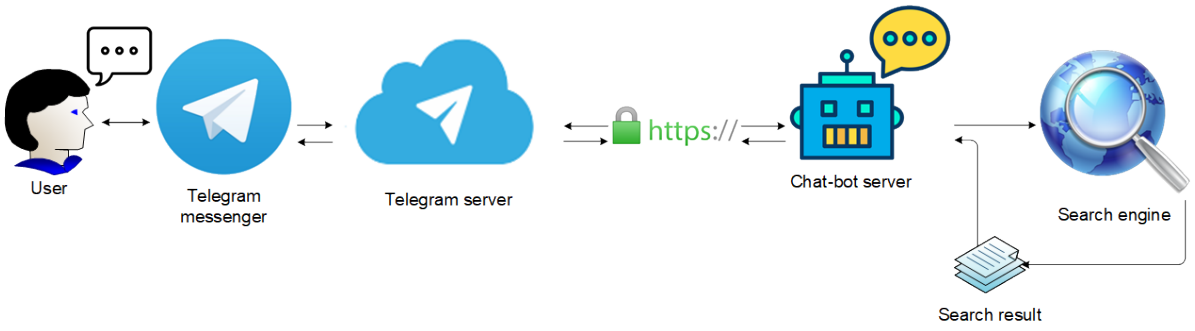

Fig. 4. The scheme of the intelligent assistant based on the Telegram messenger

Messages sent by users are transmitted to the software running on the server. The Telegram intermediate server handles all encryption and communication with the Telegram API independently. Communication with this server takes place using a simple HTTPS interface. All requests are served via HTTPS and should be presented in the following form:

`https://api.telegram.org/bot<token>/METHOD_NAME?Param1=<p1>&ParamN=<pn>`,

where `<token>` — unique chatbot key, `METHOD_NAME` — the method to be called from the chatbot, `Param1, paramN` — parameters of the called method (may be missing), `p1, pn` — parameter values.

The response to requests comes in the form of a JSON object, in which there will always be a Boolean ok field and an optional string description field containing a description of the query result in a format that is easily perceived by a person. If ok:true, the request was successful and the result of its execution can be seen in the result field (Fig. 5).


```
{
   "ok":true,
   "result": {
      "message_id":759,
      "from": {
         "id":457281743,
         "is bot":true,
         "first_name":"SearchBot",
         "username":"SearchBot",
         "language_code":"ru"
      },
      "chat": {
         "id":254188525,
         "first_name":"Vladislav",
         "last name":"Rublev",
         "username":"spac1k",
         "type":"private"
      },
      "date":1648962474,
      "text":"Hello"
   }
}
```


Fig. 5. The response to the successful request

In case of an error (ok:false) the response will have an error_code field with an integer error code and its causes will be described in the description field (Fig. 6).


```
{
   "ok":false,
   "error_code":400,
   "description":"Bad Request: message text is empty"
}
```


Fig. 6. The response to the request in case of an error

7. Experimental research

To assess the quality of the search assistant, a corpus of 90 questions with "reference" answers from the sections of frequently asked questions and answers on educational websites has been collected. Their small number is related to the methodology of preparing data for testing, as to ensure the "profitability" of queries, they were taken in the chapters of frequently asked questions and duplications of queries similar in meaning were removed.

To assess the quality of the search, classical measures were used [28] — cosine similarity, recall, precision, F-measure and originality.

Let rel be the set of all relevant documents, det — be the set of detected documents.

1. Cosine similarity is a measure of the proximity of two texts (in vector form), which is used to measure the cosine of the angle between them. It is calculated by the formula:

$$S_c = \cos(\theta) = \frac{A \cdot B}{\|A\| \cdot \|B\|} = \frac{\sum_{i=1}^{n} A_i \cdot B_i}{\sqrt{\sum_{i=1}^{n} A_i} \cdot \sqrt{\sum_{i=1}^{n} B_i}}. \quad (7)$$

2. Recall determines how well the system finds the documents the user needs, it is the ratio of the relevant documents found to the total number of relevant documents:

$$R = \frac{|rel| \bigcap |det|}{|rel|}. \quad (8)$$

3. Precision determines the ability of the system to issue only relevant documents to the user, it is calculated as the ratio of the relevant documents found to the total number of documents found:

$$P = \frac{|rel| \bigcap |det|}{|det|}. \quad (9)$$

4. F-measure is a weighted harmonic mean of recall and precision, and allows you to give different weight to recall and precision if you need to give priority to one of these metrics:

$$F = \frac{1}{\alpha \frac{1}{P} + (1-\alpha) \frac{1}{R}}, \alpha \in [0,1]. \quad (10)$$

For $\alpha = 1/2$, a balanced F-measure is obtained and calculated using the following formula:

$$F = \frac{2PR}{P+R}. \quad (11)$$

5. Originality determines the number of different search results (documents with different content).

The following experiments were carried out:

1) evaluation of the distribution of genres across the pages of educational sites;
2) evaluation of the average proximity of search results to the "reference" answers;
3) evaluation of the quality of the work of an intelligent assistant in comparison with the Yandex search engine;
4) estimation of the average number of clarifying questions needed to find information.

Using the corpus of sites, a study was conducted on the distribution of genres by pages — on average, there are 1.84 genres per page. The results of this experiment allow us to determine the number of clusters at the stage of thematic clustering. To determine the number of clusters, its genre and volume are looked at for each fragment, and depending on these indicators, the number of clusters is selected.

With the help of a corpus of frequently asked questions and "reference" answers, an experiment was conducted to assess the quality of search based on the proximity measure (formula 7). The average proximity of the first search result of an intelligent assistant to the "reference" answers is calculated. The experiment was conducted for the website of Novosibirsk State University: size of 153 pages, the average number of words per page is 863. The average proximity turned out to be 0.81.

When evaluating the quality of the search for an intelligent assistant in comparison with the Yandex search engine, the search was carried out on the website of Novosibirsk State University. Frequently asked questions from educational websites were taken as search queries, for example, "How can I apply?", "How to get an increased state academic scholarship?", "How many budget places have been allocated this year?", etc. The average values of recall, precision, F-measure and originality for all questions of the corpus and the first three results of search engine results are calculated. The results obtained are shown in Tab. 2.

Table 2. Evaluation of search quality

	Recall	Precision	F-measure	Originality
Yandex search engine	82.16%	91.83%	86.73%	71.84%
Informational chatbot	83.37%	92.75%	87.81%	80.92%

To conduct an experiment to estimate the number of clarifying questions required to find the necessary information, 4 experts were involved, who were given the task to find some information with the help of an intelligent assistant and record the number of clarifying questions required. 40 experiments were conducted and as a result it turned out that the average of 1.75 clarifying questions are required to find the necessary information. For example, after the query "How much does it cost to study?" one of the variants of the query refinement may be "How much does it cost to study at a master's degree?", after the request "How to apply?" there may be a request "Is it possible to send documents by e-mail?". The data on the experiment are presented in Tab. 3.

Table 3. Statistics of the experiment to estimate the number of clarifying questions

	Expert 1	Expert 2	Expert 3	Expert 4
Maximum search time	5	3	3	4
Minimum search time	1	0	1	0
Average search time	2.2	1.1	2.5	1.2

The main errors occur when the search selection is already small, and the user reduces it with a new query. Such situations should be tracked separately and a new search throughout the site should be performed.

Conclusion

The paper proposes an approach to creating intelligent assistants in the form of chatbots that provide site search based on a model of user intentions, genre model and preliminary thematic clustering of text content. A feature of the approach is the use of genre models developed for a given type of resource (educational, informational, etc.), on the basis of which genre structuring of the content of a particular site is carried out. The resulting genre structures allow you to more accurately determine the boundaries of thematic clusters related to the topic of the user's

search query. Further search is carried out by standard methods. In order to improve the quality of information support for the user, a model of the user's search intentions has been developed, which allows you to implicitly get information about what the user was not satisfied with in the search results and refine a new search query. The conducted experimental study showed that the created intelligent assistant provides a good quality of searching for useful information and reduces the search time.

Thus, the scientific contribution is the use of genre models to solve the problems of automatic construction of chatbots and local information search, and in the future this approach can be useful for solving individual text processing tasks (annotation construction, text classification, etc.). The developed system scales well, in particular, the resources created are applicable to arbitrary educational sites, and in order to configure the system for other types of sites, it is enough to write a new genre model and index the specified sites of a new type (for this purpose, an independent indexing module has been developed in the system).

Further direction of work: other methods of text preprocessing (word2vec, FastText), add synonyms dictionaries, apply clustering based on FRiS functions, correction of errors in the request, auto-completion of the query (based on context search).

References

1. Mehler A., Sharoff S., Santini M. Genres on the Web. Computational Models and Empirical Studies. Dordrecht, Springer, 2010. 362 p.
2. Dong L., Watters C., Duffy J., Shepherd M. An Examination of Genre Attributes for Web Page Classification. Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS'08). IEEE, 2008. P. 133–143. DOI: 10.1109/HICSS.2008.53.
3. Kutovenko A. Professional internet search. St. Petersburg, Piter Publishing House, 2011. P. 70–73. (in Russian)
4. Osinski S., Weiss D. Carrot2 Project. Carrot2 – Open Source Search Results Clustering Engine. URL: `http://project.carrot2.org/` (accessed: 30.08.2022).
5. Kutovenko A. Professional internet search. St. Petersburg, Piter Publishing House, 2011. P. 74–77. (in Russian)
6. Official website of the question and answer search engine AskNet. URL: `http://asknet.ru/` (accessed: 30.08.2022). (in Russian)
7. Radhakrishnan A. Hakia's Semantic Search: The Answer to Poor Keyword Based Relevancy. Search Engine Journal. URL: `https://www.searchenginejournal.com/hakias-semantic-search-the-answer-to-poor-keyword-based-relevancy/5246/` (accessed: 30.08.2022).
8. Introducing the Knowledge Graph: things, not strings. URL: `https://blog.google/products/search/introducing-knowledge-graph-things-not` (accessed: 30.08.2022).
9. The Palekh Algorithm: how neural networks help Yandex search. URL: `https://yandex.ru/blog/company/algoritm-palekh-kak-neyronnye-seti-pomogayut-poisku-yandeksa` (accessed: 30.08.2022). (in Russian)

10. Technical Approaches for Building Conversational AI. URL: `https://www.topbots.com/building-conversational-ai/` (accessed: 30.08.2022).

11. Nimavat K., Champaneria T. Chatbots: an overview of types, architecture, tools and future possibilities. International Journal for Scientific Research and Development. 2017. Vol. 5, no. 7. P. 1019–1024.

12. Wu Y., Wu W., Xing C., *et al.* Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-based Chatbots. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017), Vancouver, Canada, July 30 – August 4, 2017. P. 496–505. DOI: 10.18653/v1/P17-1046.

13. Kapočiūtė-Dzikienė J. A Domain-Specific Generative Chatbot Trained from Little Data. Applied Sciences. 2020. Vol. 10, no. 7. Article no. 2221. DOI: 10.3390/app10072221.

14. Cuayáhuitl H., Lee D., Ryu S., *et al.* Ensemble-based deep reinforcement learning for chatbots. Neurocomputing. 2019. Vol. 366. P. 118–130. DOI: 10.1016/j.neucom.2019.08.007.

15. Kim S., Kwon O.-W., Kim H. Knowledge-Grounded Chatbot Based on Dual Wasserstein Generative Adversarial Networks with Effective Attention Mechanisms. Applied Sciences. 2020. Vol. 10, no. 9. P. 3335. DOI: 10.3390/app10093335.

16. Bahtin M.M. The problem of speech genres. Jestetika slovesnogo tvorchestva (Aesthetics of Verbal Creation). Moscow, Iskusstvo, 1986. P. 250–296. (in Russian)

17. Kononenko I.S., Sidorova E.A. Genre aspects of website classification. Software Engineering. 2015. Vol. 8. P. 32–40. (in Russian)

18. Sidorova E.A. A comprehensive approach to the study of lexical characteristics of the text. Vestnik SibGUTI. 2019. Vol. 3. P. 80–88. (in Russian)

19. MacQueen J.B. Some Methods for classification and Analysis of Multivariate Observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press, 1967. P. 281–297.

20. Guo J., Hartung S., Komusiewicz C., *et al.* Exact algorithms and experiments for hierarchical tree clustering. Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11–15, 2010. AAAI Press, 2010. P. 1–6.

21. Manwar A., Mahalle H., Chinchkhede K., *et al.* A vector space model for information retrieval: a MATLAB approach. Indian Journal of Computer Science and Engineering. 2012. Vol. 3. P. 222–230.

22. Rendon E., Abundez I., Arizmendi A., *et al.* Internal versus external cluster validation indexes. International Journal of computers and communications. 2011. Vol. 5, no. 1. P. 27–34.

23. Liu Y., Li Z., Xiong H., *et al.* Understanding of internal clustering validation measures. IEEE International Conference on Data Mining, Sydney, NSW, Australia, December 13–17, 2010. IEEE, 2010. P. 911–916. DOI: 10.1109/tsmcb.2012.2220543.

24. Arbelaitz O., Gurrutxaga I., Muguerza J., *et al.* An extensive comparative study of cluster validity indices. Pattern Recognition. 2013. Vol. 46. P. 243–256. DOI: 10.1016/j.patcog.2012.07.021.

25. Rousseeuw P.J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics. 1987. Vol. 20. P. 53–65. DOI: 10.1016/0377-0427(87)90125-7.

26. Zagoruiko N.G., Borisova I.A., Kutnenko O.A., Dyubanov V.V. Constructing a compressed description of data using the competitive similarity function. Industry math. 2013. Vol. 16, no. 1. P. 275–286.

27. Telegram Bot API. URL: `https://core.telegram.org/bots/api` (accessed: 30.08.2022).

28. Manning C. D., Raghavan P., Schütze H. Introduction to Information Retrieval. Cambridge University Press, 2008. P. 151–175. DOI: 10.1017/CBO9780511809071.

УДК 004.912 **DOI: 10.14529/cmse220404**

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНЫХ ПОМОЩНИКОВ ДЛЯ ПОИСКА ПО КОНТЕНТУ ВЕБ-САЙТА ОПРЕДЕЛЕННОГО ЖАНРА

© 2022 В.Д. Рублев, Е.А. Сидорова

Институт систем информатики им. А.П. Ершова СО РАН
(630090 Новосибирск, пр. Академика Лаврентьева, д. 6)
E-mail: rubleffvlad@gmail.com, lsidorova@iis.nsk.su

Поступила в редакцию: 06.11.2022

В данной работе предлагается подход к созданию интеллектуальных помощников в виде чат-ботов, поддерживающих информационный поиск на основе модели намерений пользователя, предварительной жанровой и тематической кластеризации контента веб-сайта. Особенностью подхода является использование жанровых моделей, разрабатываемых для заданного типа ресурса (образовательный, информационный и т.п.), на основе которых осуществляется жанровая структуризация контента конкретного сайта. Полученные жанровые структуры позволяют более точно определять границы тематических кластеров, относящиеся к теме поискового запроса пользователя. Оценка качества поиска по сайту НГУ показала F-меру 87.8% и оригинальность 80.9%, что превосходит результаты поисковой системы Яндекс на 1.1% и 9.1% соответственно. С целью повышения качества информационной поддержки пользователя разработана модель поисковых намерений пользователя, которая позволяет неявно получить информацию о том, что пользователя не устроило в поисковой выдаче и уточнить новый поисковый запрос. В практической части работы реализован чат-бот на платформе мессенджера Telegram для информационного поиска по сайтам образовательных организаций. Проведенные эксперименты показали, что пользователю в среднем требуется 1.75 уточняющих вопросов для нахождения необходимой информации.

Ключевые слова: поисковая система, интеллектуальный помощник, жанровая модель веб-сайта, тематический анализ, модель поисковых намерений пользователя.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Rublev V.D., Sidorova E.A. Developing Intelligent Assistants to Search for Content on Websites of a Certain Genre // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2022. Т. 11, № 4. С. 51–66. DOI: 10.14529/cmse220404.

Литература

1. Mehler A., Sharoff S., Santini M. Genres on the Web. Computational Models and Empirical Studies. Dordrecht: Springer, 2010. 362 p.

2. Dong L., Watters C., Duffy J., Shepherd M. An Examination of Genre Attributes for Web Page Classification // Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS'08), 2008. P. 133–143. DOI: 10.1109/HICSS.2008.53

3. Кутовенко А. Профессиональный поиск в интернете. СПб.: Питер, 2011. С. 70–73.

4. Osinski S., Weiss D. Carrot2 Project. Carrot2 – Open Source Search Results Clustering Engine, URL: `http://project.carrot2.org/` (дата обращения: 30.08.2022).

5. Кутовенко А. Профессиональный поиск в интернете. СПб.: Питер, 2011. С. 74–77.

6. Официальный сайт вопросно-ответной поисковой системы AskNet. URL: `http://asknet.ru/` (дата обращения: 30.08.2022).

7. Radhakrishnan A. Hakia's Semantic Search: The Answer to Poor Keyword Based Relevancy. Search Engine Journal. URL: `https://www.searchenginejournal.com/hakias-semantic-search-the-answer-to-poor-keyword-based-relevancy/5246/` (дата обращения: 30.08.2022).

8. Introducing the Knowledge Graph: things, not strings. URL: `https://blog.google/products/search/introducing-knowledge-graph-things-not` (дата обращения: 30.08.2022).

9. Алгоритм «Палех»: как нейронные сети помогают поиску Яндекса. URL: `https://yandex.ru/blog/company/algoritm-palekh-kak-neyronnye-seti-pomogayut-poisku-yandeksa` (дата обращения: 30.08.2022).

10. Technical Approaches for Building Conversational AI. URL: `https://www.topbots.com/building-conversational-ai/` (дата обращения: 30.08.2022).

11. Nimavat K., Champaneria T. Chatbots: an overview of types, architecture, tools and future possibilities // International Journal for Scientific Research and Development. 2017. Vol. 5, no. 7. P. 1019–1024.

12. Wu Y., Wu W., Xing C., *et al.* Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-based Chatbots // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017), Vancouver, Canada, July 30 – August 4, 2017. P. 496–505. DOI: 10.18653/v1/P17-1046.

13. Kapočiūtė-Dzikienė J. A Domain-Specific Generative Chatbot Trained from Little Data. Applied Sciences. 2020. Vol. 10, no. 7. Article no. 2221. DOI: 10.3390/app10072221.

14. Cuayáhuitl H., Lee D., Ryu S., *et al.* Ensemble-based deep reinforcement learning for chatbots // Neurocomputing. 2019. Vol. 366. P. 118–130. DOI: 10.1016/j.neucom.2019.08.007.

15. Kim S., Kwon O.-W., Kim H. Knowledge-Grounded Chatbot Based on Dual Wasserstein Generative Adversarial Networks with Effective Attention Mechanisms // Applied Sciences. 2020. Vol. 10, no. 9. P. 3335. DOI: 10.3390/app10093335.

16. Бахтин М.М. Проблема речевых жанров // Эстетика словесного творчества. Искусство, 1986. С. 250–296.

17. Кононенко И.С., Сидорова Е.А. Жанровые аспекты классификации веб-сайтов // Программная инженерия. 2015. № 8. С. 32–40.

18. Сидорова Е.А. Комплексный подход к исследованию лексических характеристик текста // Вестник СибГУТИ. 2019. № 3. С. 80–88.

19. MacQueen J.B. Some Methods for classification and Analysis of Multivariate Observations // Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press, 1967. P. 281–297.

20. Guo J., Hartung S., Komusiewicz C., *et al.* Exact algorithms and experiments for hierarchical tree clustering // Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11–15, 2010. AAAI Press, 2010. P. 1–6.

21. Manwar A., Mahalle H., Chinchkhede K., *et al.* A vector space model for information retrieval: a MATLAB approach // Indian Journal of Computer Science and Engineering. 2012. Vol. 3. P. 222–230.

22. Rendon E., Abundez I., Arizmendi A., *et al.* Internal versus external cluster validation indexes // International Journal of computers and communications. 2011. Vol. 5, no. 1. P. 27–34.

23. Liu Y., Li Z., Xiong H., *et al.* Understanding of internal clustering validation measures // IEEE International Conference on Data Mining, Sydney, NSW, Australia, December 13-17, 2010. IEEE, 2010. P. 911–916. DOI: 10.1109/tsmcb.2012.2220543.

24. Arbelaitz O., Gurrutxaga I., Muguerza J., *et al.* An extensive comparative study of cluster validity indices // Pattern Recognition. 2013. Vol. 46. P. 243–256. DOI: 10.1016/j.patcog.2012.07.021.

25. Rousseeuw P.J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis // Journal of computational and applied mathematics. 1987. Vol. 20. P. 53–65. DOI: 10.1016/0377-0427(87)90125-7.

26. Загоруйко Н.Г., Борисова И.А., Кутненко О.А., Дюбанов В.В. Построение сжатого описания данных с использованием функции конкурентного сходства // Сибирский журнал индустриальной математики. 2013. № 1. С. 29–41.

27. Telegram Bot API. URL: `https://core.telegram.org/bots/api` (дата обращения: 30.08.2022).

28. Manning C. D., Raghavan P., Schütze H. Introduction to Information Retrieval. Cambridge University Press, 2008. P. 151–175. DOI: 10.1017/CBO9780511809071.

Рублев Владислав Дмитриевич, аспирант, Институт систем информатики им. А.П. Ершова Сибирского отделения Российской академии наук, Лаборатория ИИ (Новосибирск, Россия)

Сидорова Елена Анатольевна, с.н.с., к.ф.-м.н., Институт систем информатики им. А.П. Ершова Сибирского отделения Российской академии наук, Лаборатория ИИ (Новосибирск, Россия)